DEVELOPMENT OF SOFTWARE FOR ESTIMATING THE COST OF A PALM OIL REFINERY AND FRACTIONATION PLANT

BY

POPOOLA AYOBAMI OLANREWAJU

M.ENG/SEET/2005/1142

A THESIS SUBMITTED

TO

DEPARTMENT OF CHEMICAL ENGINEERING, SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGERIA.

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF MASTER OF ENGINEERING (M. ENG) IN CHEMICAL ENGINEERING

SEPTEMBER 2006

DECLARATION

I hereby declare that this work was carried out by me and it is a record of my research work. It has not been presented in the previous application for a higher degree. All the sources of information are duly acknowledged by means of references.

.... 14/07/08 ....

Popoola Ayobami **Date**

CERTIFICATION

The thesis titled: "Development of software for estimating the cost of a palm oil Refinery and Fractionation Plant" by: POPOOLA AYOBAMI OLANREWAJU (M.Eng/SEET/05/1142) meets the regulations governing the award of the degree of master's degree of the Federal University of Technology, Minna and is approved for its contribution to scientific knowledge and literary presentation.

------------------------------	18/01/08
Professor R.K. Onifade Name of Supervisor	Date
------------------------------	21/01/08
Dr. M.O. Edoga Name of Head of Department	Date
------------------------------	21/01/08
Professor J.O. Odigure Name of Dean,SEET	Date
------------------------------	5/8/08
Prof. S.L. Lamai Name of Dean, Postgraduate School	Date

DEDICATION

This project is dedicated to God who is the Supreme Being and to the memory of my loving sister Mrs. Oluokun Omosalewa A.

ACKNOWLEDGMENT

All glories belong to God the omnipotent and omnipresent being for divine grace, wisdom and understanding bestowed onto me during the course of this work.

My profound gratitude goes to my supervisor in person of Professor K.R. Onifade for his fatherly support and advice.

My regard also goes to my love Omokehinde Oluwayemi Ajao who out of her love and kindness stayed with me and contributed one thing or the other to the success of this project. I will not forget the contribution of Ayodele Popoola and other members of my family. I want to express my sincere gratitude to every member of staff of Chemical Engineering Department most especially the Head of Department Dr M.O. Edoga, and my former Head of Department Dr. F. Aberuagba I will also like to thank the following people for their immense contributions to the success of this project, Brother Tayo Afolabi, Mallam Abdulrafiu Abdulwahid, Alhaji Manko A, Baba R.K. Andrews, Samuel Jiya, salawudeen L.O, and all my course mates most especially, Suleiman M. Mallam Buhari M, Mrs. Olugbenga Adeola, Bro. N.O. Ajakaiye and Miss Momon Adija

ACKNOWLEDGMENT

All glories belong to God the omnipotent and omnipresent being for divine grace, wisdom and understanding bestowed onto me during the course of this work.

My profound gratitude goes to my supervisor in person of Professor K.R. Onifade for his fatherly support and advice.

My regard also goes to my love Omokehinde Oluwayemi Ajao who out of her love and kindness stayed with me and contributed one thing or the other to the success of this project. I will not forget the contribution of Ayodele Popoola and other members of my family. I want to express my sincere gratitude to every member of staff of Chemical Engineering Department most especially the Head of Department Dr M.O. Edoga, and my former Head of Department Dr. F. Aberuagba .I will also like to thank the following people for their immense contributions to the success of this project, Brother Tayo Afolabi, Mallam Abdulrafiu Abdulwahid, Alhaji Manko A, Baba R.K. Andrews, Samuel Jiya, salawudeen L.O, and all my course mates most especially, Suleiman M, Mallam Buhari M, Mrs. Olugbenga Adeola, Bro. N.O. Ajakaiye and Miss Momon Adija

Table of content

LIST OF TABLES

LIST OF TABLES

LIST OF APPENDICES

NOMENCLATURE

610FR	Java source code representing six-tenth factor rule
CIS	Java source code representing cost index and scaling method
Crsti	Current cost of crysallizer (₦)
CurIndex	Current chemical engineering's plant cost index (no unit)
Current cost B	Current cost of equipment B (₦)
E_i	The variable exponential which is the slope of correlation curve on the log- log cost (no unit)
f_I	Indirect cost factors used in the fixed capital investment (no unit)
Fcapn	Fixed capital investment (₦)
Hei	Current cost of heat exchanger (₦)
HisIndex	Reference/historical chemical engineering's plant cost index (no unit)
Historic cost A	Historical cost of equipment A (₦)
Historic cost B	Historical cost of equipment B (₦)
Oti	Current cost of olein tank (₦)
Pect	Delivered purchased Equipment cost (₦)
Pui	Current cost of pump (₦)
QA	known capacity of equipment A (either in m, m^2, m^3, m^3/s)
QB	known capacity of equipment B (either in m, m^2, m^3, m^3/s)
Ratio Index(ri)	Ratio of current index to the reference index (no unit)
SQL	Structural querying Language
Sum	Sum of direct factor (no unit)

Sum1	sum of indirect factors (no unit)
K	Sum of the cost of equipments used for drawing the chart (₦)
Tcapn	Total capital investment (₦)
Wcapn	Working capital investment (₦)

ABSRACT

Two computer aided design programs were developed using an object oriented language called java whose computer workstations are Windows based unlike some previously existing computer aided design software's developed in languages such as FORTRAN, PASSCAL.

The two programs are based on cost index and scaling method (CIS) and Six-tenth factor rule (610FR). The Programs were developed for costing and design of Palm Oil Refinery and Fractionation plant. There are thirty one sub modules (subclasses) in each program which computes current cost which is the cost in 2006 for each equipment and the current capital investments of the plant using 1996 cost data (historical cost data) for the two cost estimation methods. The value of capital investment and total capital investment obtained from the two programs differ with 610FR having higher values. The graphical presentation enables one to readily view the changes on the total capital investment when the cost of any of the equipment is changed, thus enabling quick decision to be made when design is presented before a company board.

The impact of cost fluctuation of selected equipments such as the Crystallizer, Pump, Heat exchangers and Olein tank on capital investment of the plant was also studied. The results show that the cost of heat exchanger has the most impact. Hence in cost optimization of the design or modification of the plant, the heat exchanger can be made the independent variable of the objective function.

CHAPTER ONE

1.0 INTRODUCTION

Palm oil in its crude form contains many impurities, fatty acids, phospholipids, trace metals and others which give the oil some objectionable flavour, odour and taste. This then necessitates the need for refining of this product and subsequent separation of strearin from the liquid olein in the fractionation units to obtain quality edible oil of desired purity levels done in most efficient manner. In achieving this, the oil has to undergo several stepwise processes, degumming, neutralization (chemical method) or deodorization (physical method), washing, drying, bleaching, and fractionation process. A typical modern plant comprises both refinery and fractionation units which consist of about twenty six major units. Considering this high number of unit processes, the design or modification of the plant will require enormous data for a single design or modification. This could further be complicated when a number of parameters are to be varied and the effect on the overall design is to be study. The fundamental premise of the cost analysis approach is that one can determine accurate costs by breaking a complex process into tasks (activities) consisting of cost elements that define time-based and per part consumption of resources. Therefore, application of computer aided design will be of great help to ensure speedy calculation and optimum design especially where the technical data are used to obtain the overall total investment cost. The computer aided design software package may produce its results in several formats, but it typically provides a graphically-based result which is then able to be used to create concept sketches for assessment and approval. Previous works have used computer aided design modules based on DOS applications and In addition to these the programs have not studied the impact of the cost of equipment on estimated capital investments of the plant as been done by this present program.

1.1 AIMS AND OBJECTIVES

In presenting the design or modification of plant, the chemical process engineer is continually faced with how to present the results in a way that the board of a company can make a quick appraisal and decision. The most attractive and simplistic approach is by graphical presentation of the fact with an inbuilt procedure. This work therefore, intends to use windows based graphical software to present the cost design of Palm oil Refinery and fractionation plant and show the effects of the variation of some of the equipment on total investment cost.

The objectives of this work are

1. To develop two robust computer aided design programes, one for each of the cost methods used.
2. To develop the computer aided design programes in Java Language that can be adaptable for graphical method.
3. To use the computer aided design programes for cost estimation of the Plant.
4. To show graphically the impact which the cost of some of the equipment has on the total investment cost.

1.2 Scope and Limitations

This study is based on the available technical data of Société Camerounaise de Refinnage – MAYA et Cie (S.C.R.M) a Palm oil Refinery and Fractionation Plant established in 1992 (in the Bonandale Industrial zone of Douala, the economic capital of the economic capital of Cameroon) with a refining capacity of 15,000m^3 per annum. The pertinent design specifications of the plant were use of to achieve the objectives of the study.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 Vegetable Oils

2.1.1 Background

Vegetable oil is fat extracted from plant sources, known as oil plants. Plants contain oils (example, olive oil) or fats (example, cocoa butter) and mainly in their seeds. Accumulation of triglycerides may also occur in certain yeasts and moulds. The analysis of the distribution of the fatty alkyl groups in triglycerides from numerous yeasts and mould indicates that these microbial lipids are similar to their plant counterparts mainly in locating the unsaturated chains. Vegetable oils are frequently classified into two main groups, according to their source pulp oil (palm, olive, and avocado) and seed oil (other sources). The amount of lipids in plant parts varies from as low as 0.1% in potatoes to about 70% in pecan nuts. Some vegetable products are fat poor (1% in lentils, 3% in mushrooms), some seeds have a middle range amount (about 10% in wheat germ, 20% in soybeans) while some are very oily (44% in peanuts, 55% in almonds, 65% in walnuts).

Plants store their energy production first as carbohydrates, but during ripening they transform these oxygen containing compounds in carbon-rich triglycerides. Thus, a minimum of volume is required to store maximum of energy content. Conversely, the triglyceride stock is reconverted into carbohydrates during germination along with incorporation of high amount of water. One gram of oil is converted into 2.7 g of carbohydrates.

2.1.2 Oil deposition in the plant tissues

In most plants storage of lipids are in the form of triglycerides. There are very few examples of alternative forms of stored lipid in higher plants. The most known of these is the desert shrub, jojoba, which stores its seed lipid as a liquid wax. Stored lipids may be accumulated in one or both of the main types of seed tissue, embryo or endosperm. In oilseeds such as sunflower, linseed or rapeseed, the cotyledons of the embryo are the major sites of lipid accumulation. In species such as castor bean, coriander or carrot, the endosperm is the main site of lipid accumulation. Finally, in tobacco, both embryo and endosperm tissues store lipids.

2.1.3 Types of vegetable oils

Saturated fats are unhealthy in excess, but a small amount is essential. Unsaturated fats (monounsaturated, polyunsaturated) are healthier for those living in Western-style homes, but most nutrition authorities recommend that no more than 30% of Western diet be fats. In unheated northern environments, up to 2/3 of the diet can be fats without health problems.

2.2 Palm Oil as a Case Study

The oil palm (*Elaeis guineensis*) originates from the coastal regions of West Africa, where it was mainly grown along rivers. Palm oil is presumed to have formed part of the diet in large parts of Africa well before our written history began. Evidence of palm oil has been found at archaeological digs in Egypt, dating from 3,000 BC, which seems to indicate that it was already traded on the African continent at that time. The Portuguese discovered the crop during their expeditions to West Africa in the 15th century and Palm oil later became a basic part of the food on board of slave ships.

Table 2.1 Types of Oil and Their Use

Type of Oil or Fat	Saturated	Mono unsaturated	Poly unsaturated	Smoke Point	Availability	Uses
Butter	66%	30%	4%		common	cooking, baking, condiment, sauces, flavoring
Coconut oil	92%	6%	2%		common	commercial baked goods, candy and sweets, whipped toppings, nondairy coffee creamers, shortening
Corn oil	13%	25%	62%	236 °C	common	frying, baking, salad dressings, margarine, shortening
Cottonseed oil	24%	26%	50%		common	margarine, shortening, salad dressings, commercially fried products
Grape seed oil	12%	17%	71%		common	cooking, salad dressings, margarine
Lard	41%	47%	12%	138-201 °C	common	baking, frying
Margarine, hard	80%	14%	16%		common	cooking, baking, condiment
Margarine, soft	20%	47%	33%		common	cooking, baking, condiment
Olive oil	14%	77%	9%	190 °C	common	frying, cooking, salad dressings
Palm oil	52%	38%	10%		common	cooking, flavoring, vegetable oil
Peanut oil	18%	49%	33%	231 °C	common	cooking, salad oils, margarine
Sunflower oil	10%	13%	77%	265 °C	common	cooking, salad dressings, margarine
Soybean oil	15%	24%	61%	241 °C	common	cooking, salad dressings, vegetable

Small-scale growers in central and West Africa began to export their products to Liverpool and Marseilles in the late 18th century. The industrial revolution created a much larger demand for palm oil, which was used at the time to make candles and as a lubricant for machines.

The area of wild palm groves, only partly harvested, was estimated as 2,400,000 ha, whereas there were 72,000 ha of estate plantations and another 97,000 ha of smallholder plantations. Estate plantations, which require large consolidated areas, are still difficult to create in Nigeria because the oil palm growing regions are densely populated and the complex traditional land holding system has been carefully preserved. Elsewhere in West Africa, population densities are lower, but the problems of obtaining labour to sustain plantation developments are correspondingly greater.

2.2.1 Palm Oil as a Cost Effective Product

Palm oil is obtained from the flesh of the palm fruit. Each palm tree produces approximately one fruit bunch, containing as many as 3000 fruit lets per annum. In addition, each palm tree continues producing fruit economically for up to 25 years. This ensures a constant stable supply, as compared with other annual crops.

Naturally, palm oil is characterized as stabilized oil due to its chemical composition. As such, it can be used in most food applications without hydrogenation, thus, reducing production cost by as much as 30 %. Palm oil price is also competitive and can represent a saving of up to several naira per kilogram, as compared to the price of other edible oils.

2.2.2 Commercial and domestic consumption of oils

Palm oil is one of the most widely consumed edible oils in the world today. Besides, it contains more monounsaturated fatty acids than many other vegetable oils. Recent scientific studies indicate that consumption of monounsaturated have some beneficial effects in order to maintain a healthy life style. In addition, compared with other vegetable oils, palm oil is a rich source of the anti-oxidant vitamin E containing about 360 – 600 ppm in its refined form. Commercially, palm oil products are used mainly as food additives (80%) and in the cosmetics and soap industries

2.2.3 Chemical composition of palm oils

Palm oil is extracted from the mesocarp of the fruit of the palm. Crude Palm Oil is the richest natural source of Tocotrienols.

Tocotrienols, together with tocopherols compose the vitamin E family. Natural tocotrienols exist in four different forms or isomers, namely, alpha-, beta-, gamma- and delta- tocotrienol, which contain different numbers of methyl groups on the chromanol ring. Although all the isomers are important antioxidants due to the ease of donating a hydrogen atom from the hydroxyl group on the chromanol ring to reduce free radicals, each of them has its own biological activity. Furthermore, many studies on test-tube scale indicate that tocotrienols have an anti-cancer effect, especially against skin and breast cancer for the extra three unsaturated bond. It is also a remedy for cancer, headaches, and rheumatism (Duke and Wain, 1981).

The mesocarp comprises about 70 - 80% by weight of the fruit and about 45 -50% of this mesocarp is oil. The rest of the fruit comprises the shell, kernel, moisture and other non fatty fiber. The extracted oil is known as crude palm oil (CPO) which until quite recently was known as the golden commodity.

Palm oil like all natural fats and oils comprises mainly Triglycerides, mono and diglycerides. Free fatty acids, moisture, dirt and minor components of non oil fatty matter referred to collectively as unsaponifiable matter.

1. Tryglyceride

It is a chemical compound of one molecule of glycerol bound to three molecules of Fatty Acid.

$CH_2 - OH$		R1-COOH		$CH_2 - COOR1$		
$CH - OH$	+	R2-COOH	$\rightleftharpoons$	$CH - COOR2$	+	$3H_2O$
$CH_2 - OH$		R3-COOH		$CH_2 - COOR3$		
Glycerol		Fatty Acid		Triglyceride		Water

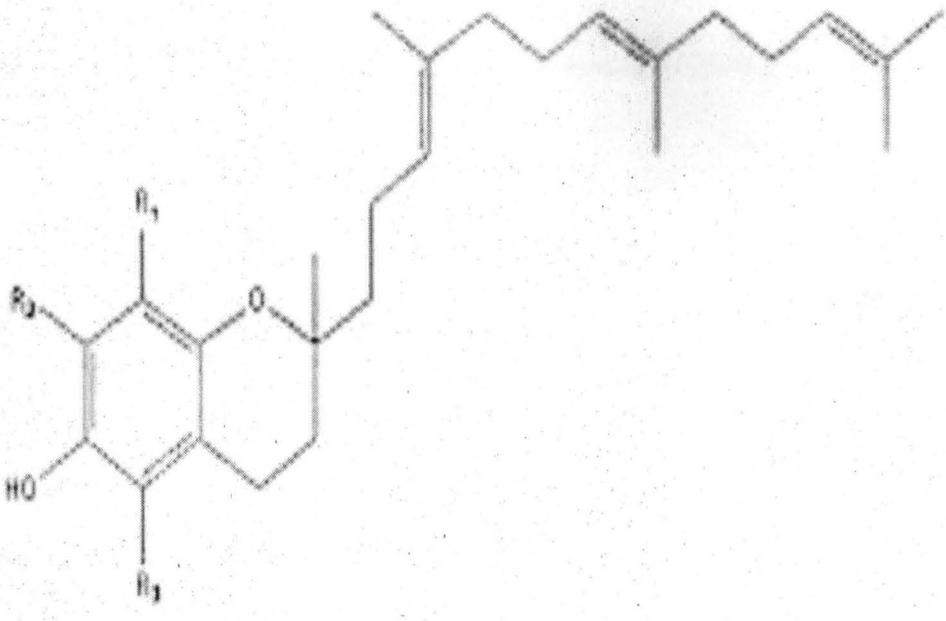


Figure 2.1 Chemical Structure of Tocotrienol

The fatty acids could be of the same type or they could be different. The property of a triglyceride will depend on the different fatty acids that combine to form the triglyceride.

The fatty acids themselves are different depending on their chain length and degree of saturation. The short chain fatty acids are of lower melting point and are more soluble in water. Whereas, the longer chain fatty acids have higher melting points. The melting point is also dependent on degree of non-saturation. Unsaturated acids will have a lower melting point compared to saturated fatty acids of similar chain length.

The 2 most predominant fatty acids in palm oil are palmitic acid (saturated) and oleic acid (unsaturated). Typical fatty acid composition of palm oil is given in Table 2.2.

2. Mono and di-glycerides and FFA

In the presence of heat and water the triglycerides break up by a process known as hydrolysis to form free fatty acids thus yielding mono and di-glycerides and FFA which is of crucial importance to the refiners.

Hydrolysis can be represented as below:

$$\begin{array}{lcccccc} CH_2-COOR1 & & & & CH_2-OH & & \\ | & & & & | & & \\ CH-COOR2 & + & H_2O & \rightleftharpoons & CH-COOR2 & + & R1COOH \\ | & & & & | & & \\ CH_2-COOR3 & & & & CH_2-COOR3 & & \\ \text{Triglyceride} & & \text{Water} & & \text{Diglyceride} & & \text{Free Fatty acid} \end{array}$$

Mono and diglycerides account for about 3 to 6% by weight of the glycerides in the oil. Good oils having lower amount of mono and diglycerides is said to be of great importance in the fractionation process because they act as emulsifying agents inhibiting crystal formation and making filtration difficult.

The amount of mono and diglycerides and FFA is reduced in the process of refining as can be seen from their concentration in the DFA (Distillate Fatty Acid).

3. Moisture and Dirt

This is a result of milling practice. Good milling will reduce moisture and dirt in palm oil but normally it is in the range of 0.25%.

Table 2.2 Composition of Fatty acid in Palm Oil

Fatty acid	Percentage compositions (%)
Lauric acid	0.2
Myrstic acid	1.1
Palmitic acid	44.0
Stearic acid	4.5
Oleic acid	39.2
Linoleic acid	10.1
Others	0.9

4. Minor Component

These are classified into one category because they are fatty in nature but are not really oils. They are referred to as unsaponifiable matter and they include the following:

a. Carotineoids
b. Tocopherols
c. Sterols
d. Polar Lipids
e. Impurities

Palm fruits contain two major types of oils, the palm oil which also contain palm olein oil and palm strearin and palm kernel oil that contain palm kernel olein and palm kernel strearin.

2.2.4 Properties of palm oil

(i) Colour: - Palm oil is dark yellow to yellow-red oil (high carotene content) of vegetable origin

(ii) Density: - All fats and oils have a particular density (approx. 0.9 g/cm^3). With a rise in temperature, however, density diminishes, thereby leading at the same time to an increase in volume. This behavior is described by the coefficient of cubic expansion and is known as thermal dilation.

The coefficient of cubic expansion amounts to: g = approx. 0.000727°C^{-1}

As a rule of thumb, oils may be expected to increase in volume by 1% of their total volume for each 14°C temperature increase. In particular in the case of oils requiring heating, such as palm oil, the ullage space (space unoccupied by the oil) must be calculated accordingly.

(iii) Humidity/Moisture content: - Fats and fatty oils are insoluble in water. However, contact with water may give rise to soluble lower fatty acids and glycerol, which cause rancidity together with changes in colour (yellow to brown), odour and taste as well as gelling and thickening. For this reason, the tanks must be absolutely dry after cleaning.

(iv) Odour: - As an active behaviour, Palm oil has an unpleasant, sweet (violet- like) odor and a neutral taste. While the passive behaviour Palm oil is sensitive to contamination by ferrous and rust particles and water (especially saline water).Tanks and barrels must always be odour-free, since there is a risk that quality will be diminished in particular where the previous cargo had a strong odour

2.3 Palm oil Refining and Fractionating

2.3.1 Quality of crude palm oil

The acid value of oil may be used as a measure of its quality. However, the acid value of the oil must not be too high, as this denotes an excessively high content of free fatty acids, which causes the oil to turn sour. Discoloration may also occur. Palm oil should have an acid value of at most 5%. Moreover, rancidity can also be used as a measure of quality of oil (palm oil). Rancidity is promoted by light, atmospheric Oxygen and moisture and leads to changes in odor and taste. Thus, the tanks and barrels must be filled as full as possible, taking into consideration the coefficient of cubic expansion, so that as little ullege space as possible is left above the cargo. Rancid oil does not meet quality requirements. Maximum duration of storage for the oil is six months at a temperature of 30°C

2.3.2 Physical and chemical refining

2.3.2.1 Physical refinery

Palm oils consist mainly of glycerides and, like other oils in their crude form, small and variable portions of non-glyceride components as well. In order to process the oils to an edible form, some of these non-glycerides need to be either removed or reduced to acceptable levels.

In term of solubility study – glycerides are of two broad types: oil insoluble and oil soluble. The insoluble impurities consisting of fruit fibres, nut shells and free moisture mainly, are readily removed. The oil soluble non-glycerides which include free fatty acids, phospholipids, trace metals, carotenoids, tocopherols or tocotrienols, oxidation products and sterols are more difficult to remove and thus, the oil needs to undergo various stages of refining.

Not all of the above non-glyceride components are undesirable. The tocopherols and tocotrienols not only help to protect the oil from oxidation, which is detrimental to flavour and quality of the finished oil, but also have nutritional attributes. Alpha (α) and beta (β) carotene are the major constituents of carotenoids, and precursors of vitamin A.

The other impurities generally are detrimental to the oil's flavour, odour, and colour and quality and thus influence the oil's usefulness.

The aim of refining is therefore to convert the crude oil to quality edible oil by removing objectionable impurities to the desired levels in the most efficient manner (Kheiri, 1973). This also means that, where possible, losses in the desirable component are kept minimal. The impurities which are contained in the crude palm oil (CPO) are shown in Table 2.3.

Generally speaking, the refining routes of palm oil are quite identical. There are two routes taken to process crude oil into refined oil; which are chemical (basic) refining and physical refining. The methods differ basically in the way the fatty acids are removed from the oil. Physical refining, which eliminates the need for an effluent plant for the soap stock, involves subjecting the oil to steam distillation under higher temperature and vacuum for removal of the free fatty acids (Swoboda, 1985). The oil is physically refined by subjecting the oil to steam distillation.

2.3.2.1.1 Physical refinery process description

The raw material which is used by physical plant is crude palm oil (CPO) from the CPO storage tank. CPO is feed at the flow rate of 35-60 tons/hour. The initial temperature of CPO is at 40 – 60°C. The feed is pumped through the heat recovery system that is plate heat exchanger to increase the temperature to 60 – 90 ° C.

After that, about 20% of the CPO fed to into the slurry tank and mix with the bleaching earth (6 – 12kg/ton CPO) to form slurry (CPO + Bleaching earth). The agitator inside the slurry tank will mix the CPO and bleaching earth completely. Then, the slurry will go into the bleacher. At the same time, another 80% of the CPO is pumped through another plate heat exchanger (PHE) and steam heater to increase the CPO temperature to 90 – 130°C (it is a desired temperature for the reaction between CPO and phosphoric acid). Then, the CPO feed is pumped to static mixers and the phosphoric acid is dosed at 0.35 – 0.45 kg/ton. Inside there, the intensive mixing is carried out with the crude oil for precipitation of the gums. The precipitation of gums will ease the filtration and prevent the formation scale in deodorizer and other heating surfaces. The degummed CPO will then move into the bleacher.

Table 2.3 Impurities in Crude Palm Oil

Substances	Content
Free Fatty Acid (FFA)	3 – 5%
Gums (phospholipids, phosphotides)	300 ppm
Dirt	0.01%
Shell	Trace
Moisture and Impurities	0.15%
Trace metal	0.50%
Oxidation Products	Trace
Total Carotenoids	500 - 1000 mg/kg

In the bleacher, 20% slurry and 80% degummed CPO will mix together and the bleaching process occur. The practice of bleaching involves the addition of bleaching earth to remove any undesirable impurities (all pigments, trace metals, oxidation products) from CPO and this improves the initial taste, final flavor and oxidative stability of product. It also helps to overcome problems in subsequent processing by adsorption of soap traces, pro-oxidant metallic ions, decomposed peroxides, reduced colour, and other minor impurities. The temperature inside the bleacher must be around 100°C – 130°C to get the optimum bleaching process for 30 minutes of bleaching period. The low pressure steam is purged into bleacher to agitate the concentrated slurry for a better bleaching condition.

The slurry containing the oil and bleaching earth is then passed through the guard filter to separate bleaching earth particles from the oil. The temperature must be maintained at around 80 – 120°C for good filtration process (Lim, 1984). In the guard filter, the slurry passes through the filter leaves and the bleaching earth is trapped on the filter leaves. Actually, the bleaching earth must be clear from Guard filter after 45minutes in operation to get a good filtration. Bleached palm oil (BPO) from Guard filter is then pumped into buffer tank as a temporary storage before further processing.

Usually, a second check filter, trap filter is used in series with the Guard filter to double ensure that no bleaching earth escaped. The presence of bleaching earth fouls deodorizer, reduces the oxidative stability of the product oil and acts as a catalyst for dimerizaition and polymerization activities. So, the "blue test" is carried out for each batch of filtration to ensure the perfect filtration process. This test indicates whether any leaking is occurring in Guard filter or trap filter. Hence, any corrective actions can be taken immediately. The BPO comes out from the filter and passes through another series of heat recovery system, Schmidt plate heat exchanger and spiral heat exchanger to heat up the BPO from 80 – 120°C until 210 – 250°C.

The hot BPO from spiral heat exchanger then proceeds to the next stage where the free fatty acid content and the color are further reduced and more importantly, it is deodorized to produce a product which is stable and bland in flavour.

In the pre-stripping and deodorizing column, deacidification and deodorization process happen concurrently. Deodorization is a high temperature, high vacuum and steam distillation process. A deodorizer operates in the following manner:

(i) Deaerates the oil,

(ii) Heats up the oil,

(iii) Steam strips the oil

(iv) Cools the oil before it leaves the system.

In the column, the oil is generally heated to approximately 240 – 280°C under vacuum. Heat bleaching of the oil occurs at this temperature through the thermal destruction of the carotenoid pigments. The use of direct steam ensures prompt removal of residue free fatty acids, aldehydes and ketones which are responsible for unacceptable odour and flavours. The lower molecular weight of vapourized fatty acids rises up the column and pulls out by the vacuum system. The fatty acid vapor leaving the deodorizer are condensed and collected in the fatty acid condenser as fatty acid. The fatty acids then is cooled in the fatty acid cooler and discharged to the fatty acid storage tank with temperature around 60 – 80°C as palm fatty acid distillate (PFAD), a by-product from refinery process.

The bottom product of the pre-stripper and deodorizer is Refined, Bleached, and Deodorized Palm Oil (RBDPO). The hot RBDPO (250 – 280°C) is pumped through Schmidt PHE to transfer its heat to incoming BPO. Then, it passes through another trap filters to have the final oil polishing at 120 – 140°C to prevent traces of bleaching earth material in the product at the prodct tank. After that, the RBDPO will pass through the RBDPO cooler and plate heat exchanger to transfer the heat to the CPO feed. The RBDPO then is pumped to the storage with temperature 50 – 80°C.

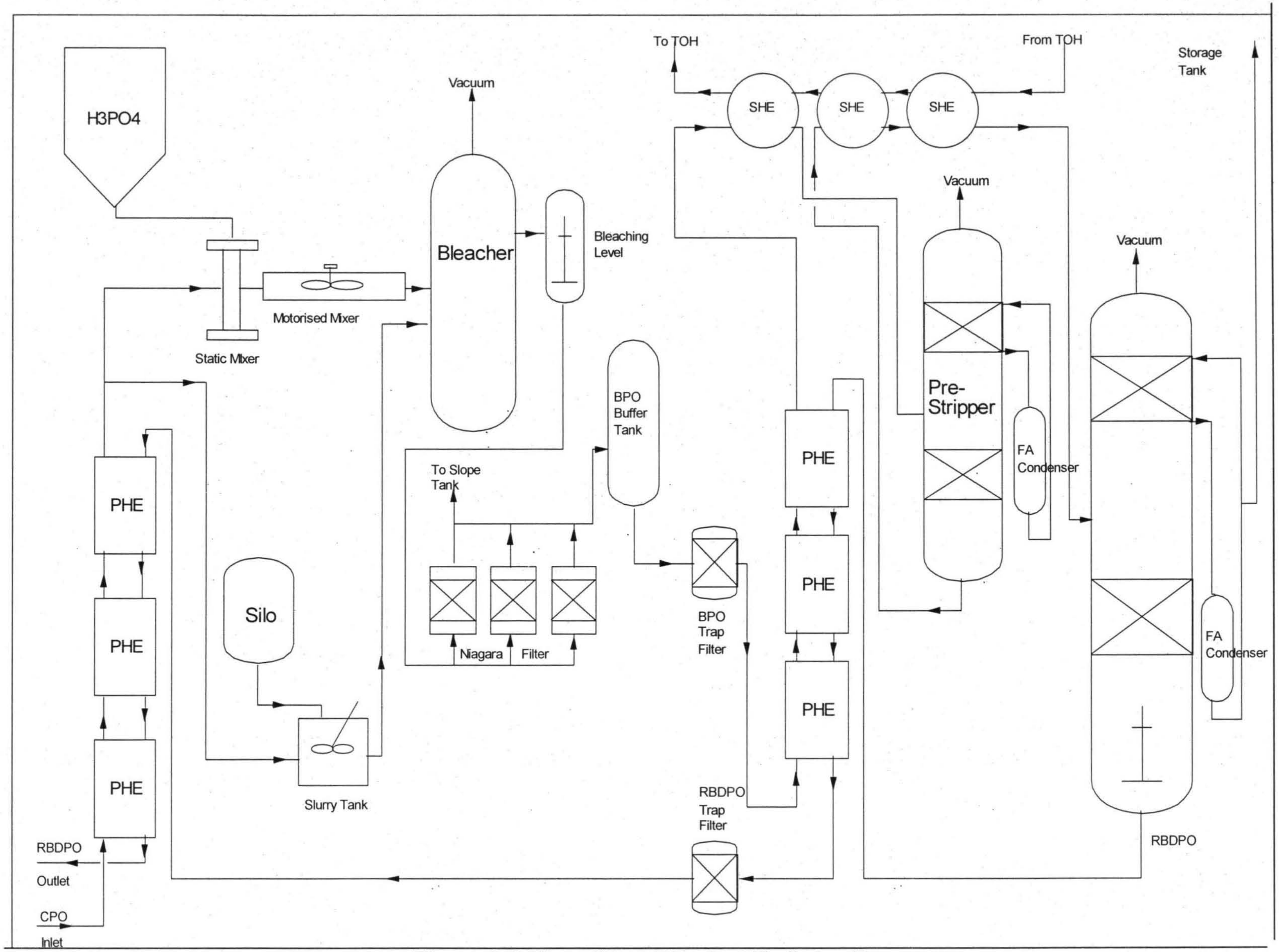


Figure 2.2 Flow Sheet Physical Refining Process

2.3.2.2 Chemical refinery

This process is also referred to as caustic refining and this involves the following major stages:-

(i) Gum conditioning and neutralization

(ii) Bleaching and filtration

(iii) Deodorization

This step wise processes are done to, (i) improve the taste and colour of the oil (ii) removal of phosphatides, free fatty acids and pigments. Removal of traces of soap and moisture occurs in the washing and drying steps. (iii) Remove objectionable volatile components such a ketone, aldehydes and alcohols.

2.3.2.2.1 Chemical refinery process description

2.3.2.2.1.1 Gum conditioning

Gums in vegetable oil need to be removed to avoid colour and taste reversion during subsequent refining steps. The Pennwalt process involves a single stage phosphoric acid treatment and a single stage hot water treatment followed by continuous removal of the hydrated gums in a Degumming Super Centrifuge.

Advantage:

(i)Availability of cleaner oil down stream, reduced oil loss in the caustic refining step, and thus higher yield of refined oil can be obtained.

(ii) Gums obtained are readily marketable.

Soap stock obtained during caustic refining can be directly used for soap-making.

(iii) Removal of gum reduces the quantity of catalyst required during hydrogenation of oil and consequently reduces the oil loss during the proess.

The bulk of certain phosphatides such as lecithin are separated through this operation. The chemical degumming process (figure 2.3) involves mixing of the oil at 80-90^{o}c with water or steam for 30 min. The gummy residue is dehydrated and the oil is then passed through centrifugal separators. Larger amounts of water or steam are used to prepare an oil which is more degummed (Kirschenbauer, 1960).

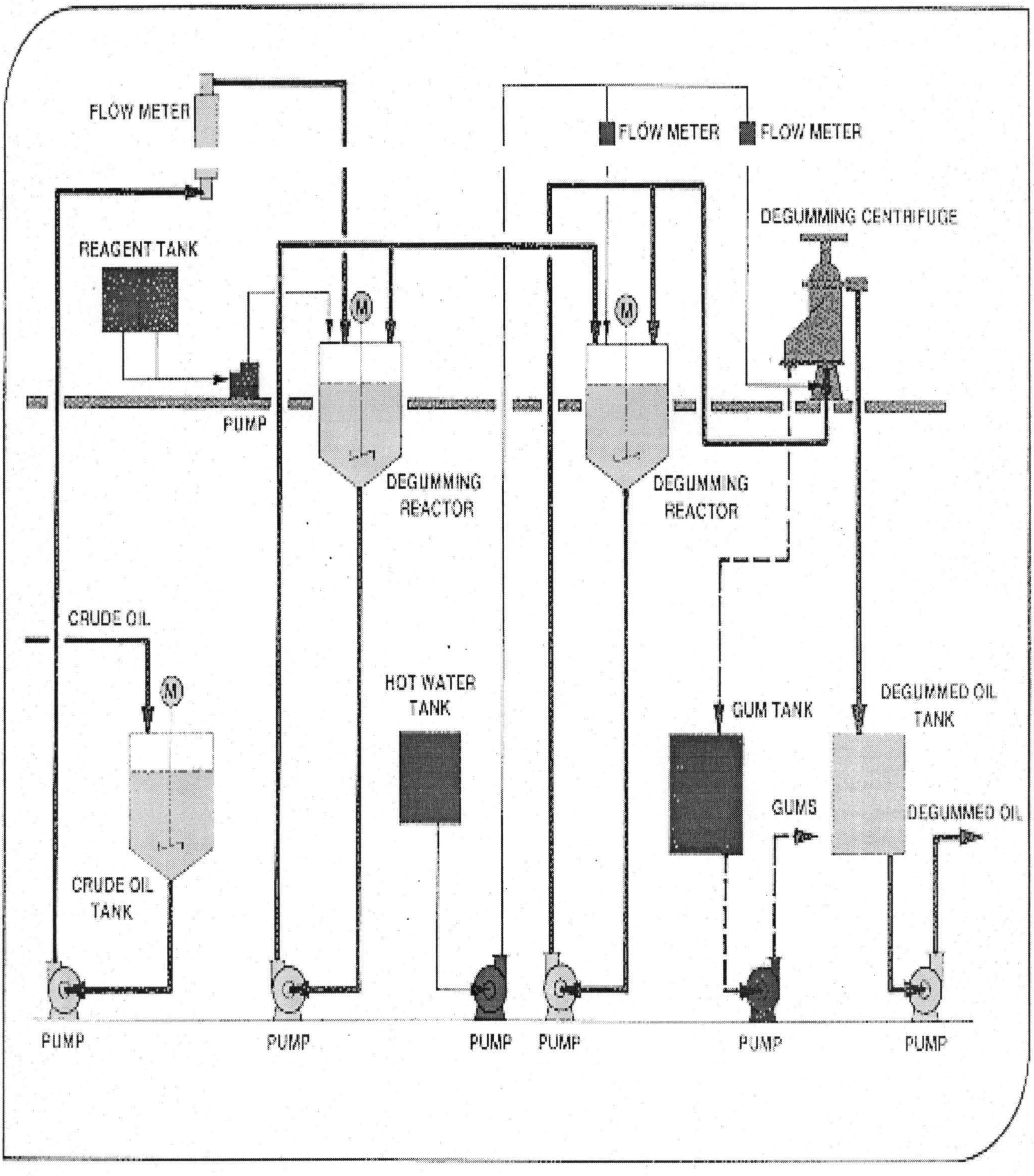


Figure 2.3 Schematic Diagram showing Degumming of Palm Oil

2.3.2.2.1.2 Neutralization of the degummed palm oil

Palm oils destined for human consumption are neutralized to remove free fatty acids, albuminous and mucilaginous matter, and thereafter washed to reduce the soap content of neutral oil to produce a more stable product. Effective neutralization results in enhanced effectiveness of subsequent steps such as bleaching, hydrogenation, winterizing, deodorizing and furthermore results in high yields of a quality product. Neutralization also results in removal of phosphatides, removal of free fatty acids and removal of pigments. Removal of traces of soap and moisture occurs in the washing and drying steps. The neutralization process (figure 2.4) consists of caustic refining and re-refining (wherever required) first water washing, second water washing and vacuum drying.

In the refining and washing steps, the degummed palm oil is treated with caustic soda of concentration of about 2M. This results in the formation of sodium soap and the oil which is ready for centrifugation.

The separation of neutral oil from soap stock and neutral oil from wash water is carried out in one or more high super centrifuges. The neutral oil is then treated with hot water to wash and rewash followed by centrifugation then vacuum drying.

2.3.2.2.1.3 Bleaching process

The neutral, washed and dried Palm Oil still contains some pigments and small traces of soap (<50 ppm) which have to be removed. This is achieved by adsorption of the pigment on an adsorbent material. Bentonite, silica gel and activated carbon are used as bleaching adsorbents (Strecker. L.R. et al, 1985). Bleaching Plant equipped with guarded filters, operates under vacuum to prevent oil oxidation. The oil is cold mixed with metered quantities of bleaching earth and/or other bleaching agents and thereafter heated to the correct temperature and pumped to the Continuous Bleacher operating under vacuum where an adequate retention time is provided to ensure effective bleaching. The oil earth slurry is further pumped to two guarded filters operating sequentially resulting in continuous bleached oil (filtrate) discharge.

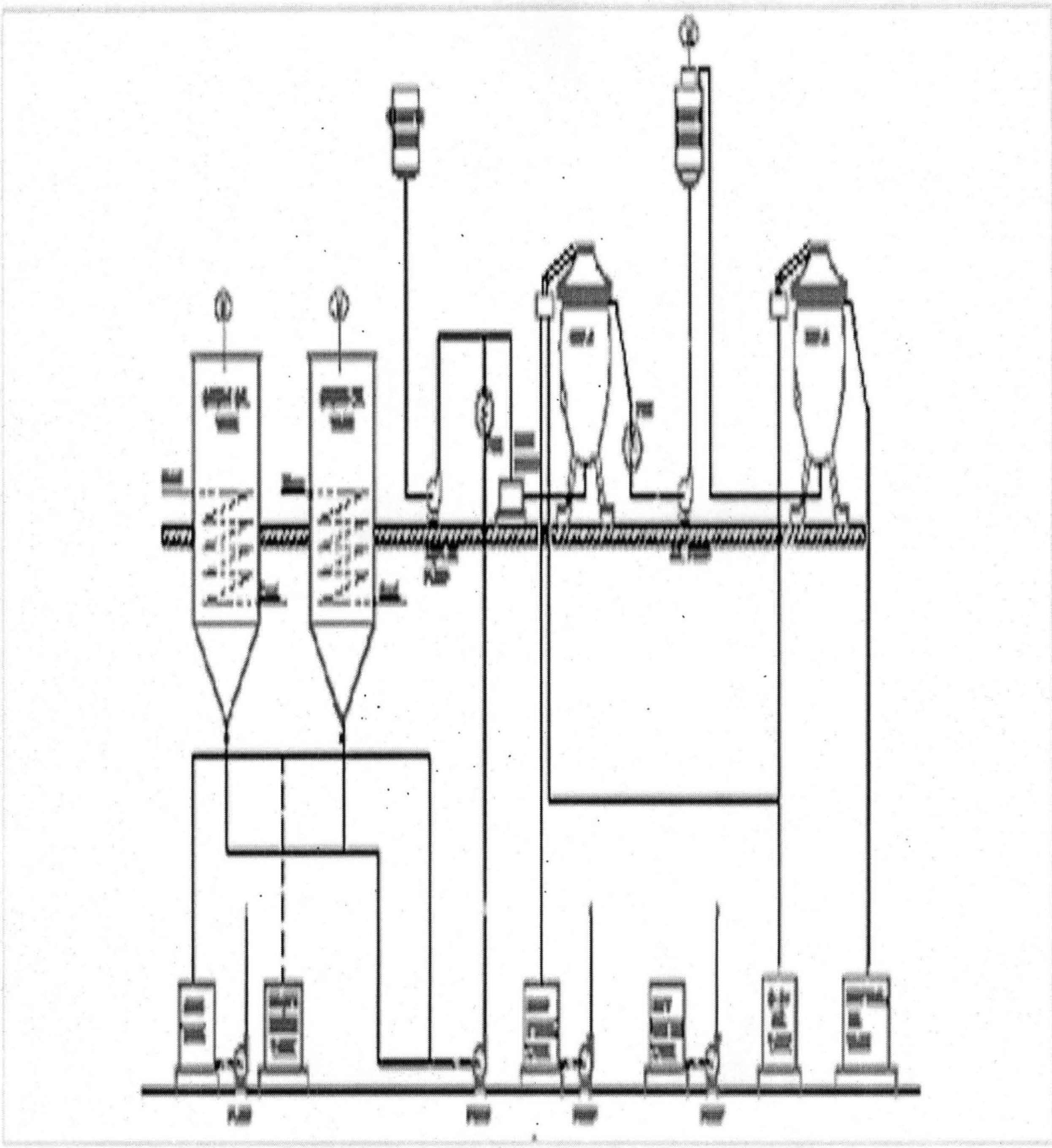


Figure 2.4 Schematic Diagram showing Neutralization of Palm Oil

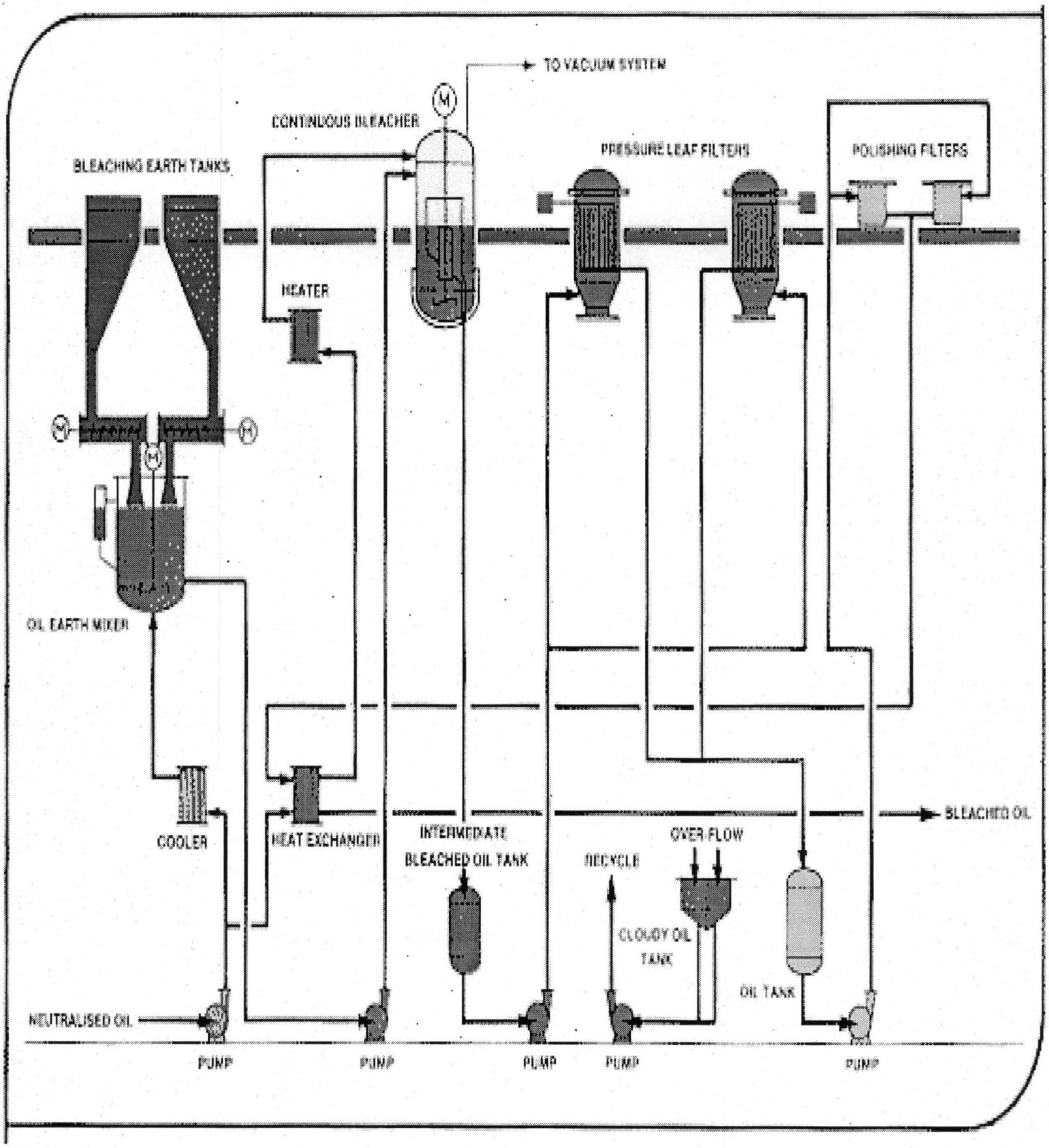


Figure 2.5 Schematic Diagram showing Bleaching of Palm Oil

2.3.2.2.1.4 Deodorization

The deodorization process (figure 2.6) is highly specialized type of vacuum steam distillation process for the purpose of removing undesirable flavourful and odorous volatile components such as ketone, aldehydes, alcohols mostly arising from oxidation, in fats and oils. Using steam under reduced pressure the volatile compounds are removed from fats and oils. Typical conditions approximate 250 °C at 2.0 mmHg absolute pressure for 1-4 hours with strong steam sparging (Lim et al 1984). The deodorization utilizes the differences in volatility between off-flavour and an off-odour substance in the triglycerides.The bleached oil is pumped by the feed pump to the deaerator where the pretreated oil is degassed. This deaerated oil is passed through a heat exchanger where the oil is heated by exchanging the heat of the deodorized oil. The oil is further heated to the stripping temperature in the pre-heater wherein thermo fluid is circulated. The above oil is fed to a flash chamber and flows thereafter to an oil distributor inside falling film deodorizer. The oil descends counter current to the stripping steam in the form of very thin film and gets completely deodorized. The fatty acids distilled are condensed, cooled and stored. The oil from the bottom flows to an intermediate vessel having an arrangement for dosing citric acid. This deodorized oil is then pumped through a heat exchanger to the polishing filter. The filtered oil is thereafter passed through a cooler and discharged for collection.

2.3.2.3 Fractionation

Fractionation involves removal of solids from the liquid oil at a given temperature. In order to cater for a wide range of markets, palm oil is fractionated to produce a product which is "harder" (Stearin) and "more liquid" (olein) than palm oil. The fractionation is accomplished by two fundamental processes namely:-

1) Crystallization
2) Filtration

Fractionation of palm oil can be described as follow. The triglycerides found in the oil have different melting points. At certain temperature, the lower melting temperature triglycerides will crystallize into solid separating the oils into both liquid (Olein) and solid (Stearin) fraction.

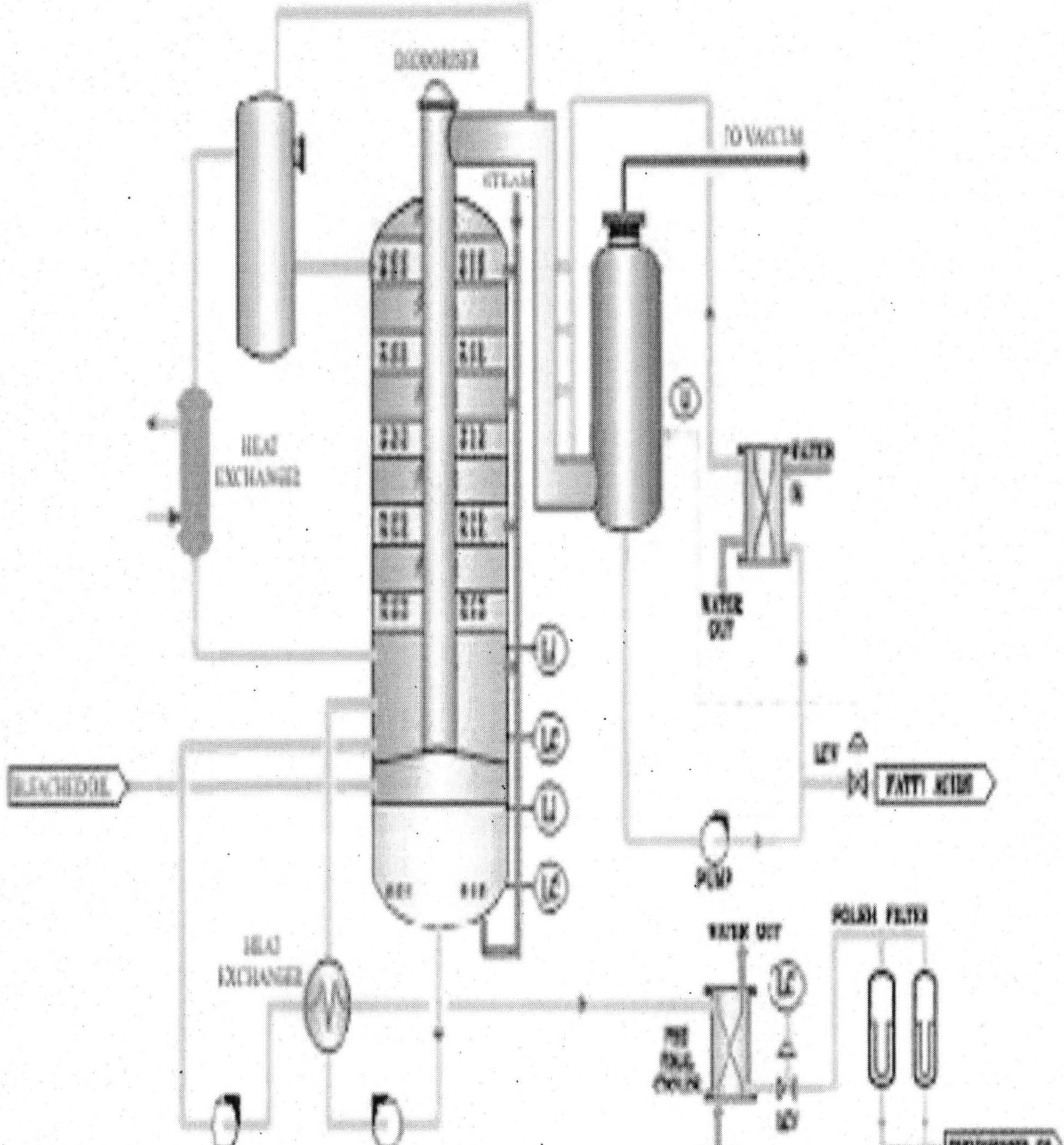


Figure 2.6 Schematic Diagram showing Deodorization of Palm Oil

1. The lowering of temperature causes cloudness and this increase oil viscosity
2. Agitation facilitates the formation of small crystals.
3. Time with a gradual decrease in temperature and stillness, promotes the formation of longer crystals.

The solution is pumped batch-wise into the crystallizer according to a pre-established program. In the crystallizer, the crystal formation and growth occurs as the oil is agitated and cooled. Cooling can be governed by controlling either the oil or water temperature.

2.3.2.3.2 Filtration process

After the crystallization process as shown above, the slurry from buffer tank passes through the filtration process for the physical separation between RBD palm stearin and RBD palm olein. Presently, the membrane filter is used for this filtration. Another alternative for this purpose is by employing drum filter for separation.

The membrane filter is pressure filter where the filter pack comprising alternatively plates and frames, or a series of chamber is compressed between one fixed and one movable cover or bulk-head. The filter media are located between each individual element. Cake will build up in the hollow space between the elements and fall out of the press when the filter pack is opened.

2.4 Design of Palm Oil Refinery and Fraction Plant

The goal of a plant design is to develop and present a complete plant that can operate on an effective industrial basis. This is achieved when the chemical engineer in charge can combine many separate units or pieces of equipment into one smoothly operating plant taking special consideration on the design specification of each unit. The design of Palm oil refinery and fractionation plant can not definitely be an exception.

Moreover the design specifications for each unit of such refinery include equipment capacities in term of volume, material of construction (as this process involves corrosive substances, mainly fatty acid), pressure and temperature.

The fraction can then be separated by filtration which is then squeezed or pressed to completely separate the liquid oil. This process is used to produce hard butter.

It is worth mentioning that in palm oil fractionation, palm olein is the premium product and the palm stearin is the discount product. Fractionation of palm oil into palm olein and palm stearin can be accomplished using two types of processes which are "Viz Dry" and "Detergent Fractionation".

The dry fractionation is used to separate the palm olein and palm stearin from the RBDPO produced by physical treatment. The RBDPO is passed through the further fractionation process to get various grades of palm olein and palm stearin. Usually three types of olein are produced:

(1) Normal grade olein, (2) super grade olein and (3) olein with cloud point 7 – 8°C.

2.3.2.3.1 Crystallization process

Firstly, the RBDPO feed must pass the quality specification, colour<2.6R and FFA< 0.075 is fed into the heat exchanger. The RBDPO feed is heated up by hot waters around 75°C. After that the oil is kept homogenized at about 70°C before the start of crystallization. The idea is to destroy any crystals present and to induce crystallization in a controlled manner in the crystallizer.

After that, the oil is pumped to the crystallizer. The crystallization system is a batch type and is equipped with special crystallizers operating alternatively. These crystallizers are made up of vertical cylindrical vessel filled with thermo-regulated water. The barrels containing the oil to be fractionated are then submerged in the water and each of these barrels is fitted with a mechanical agitator. An automatic station controls the temperature in the various crystallizers.

The crystallization process is carried out to remove the higher melting glycerides which cause liquid oils to become cloudy and more viscous at low temperature. There are three factors that are fundamentally important in the crystal formation and its characteristics:

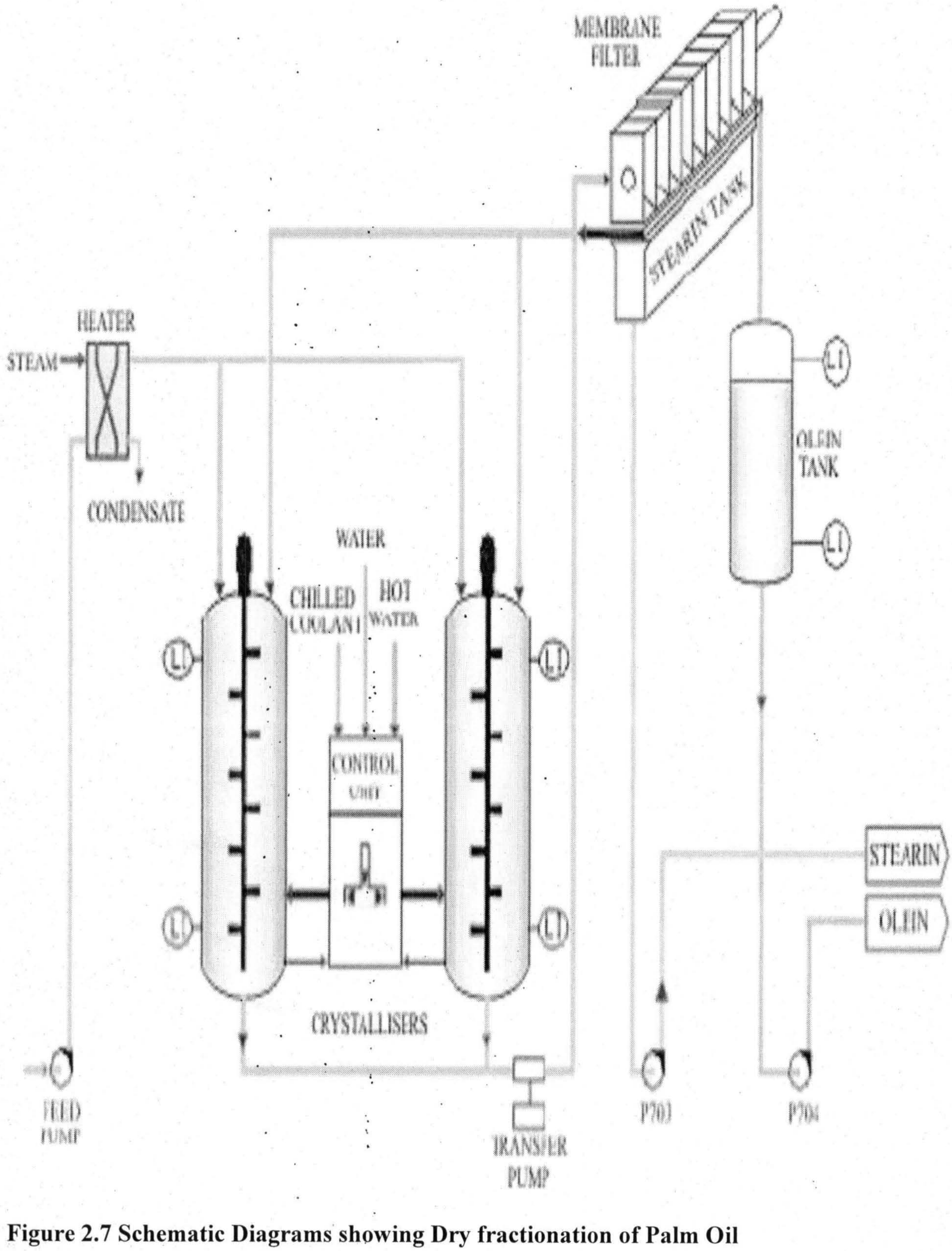
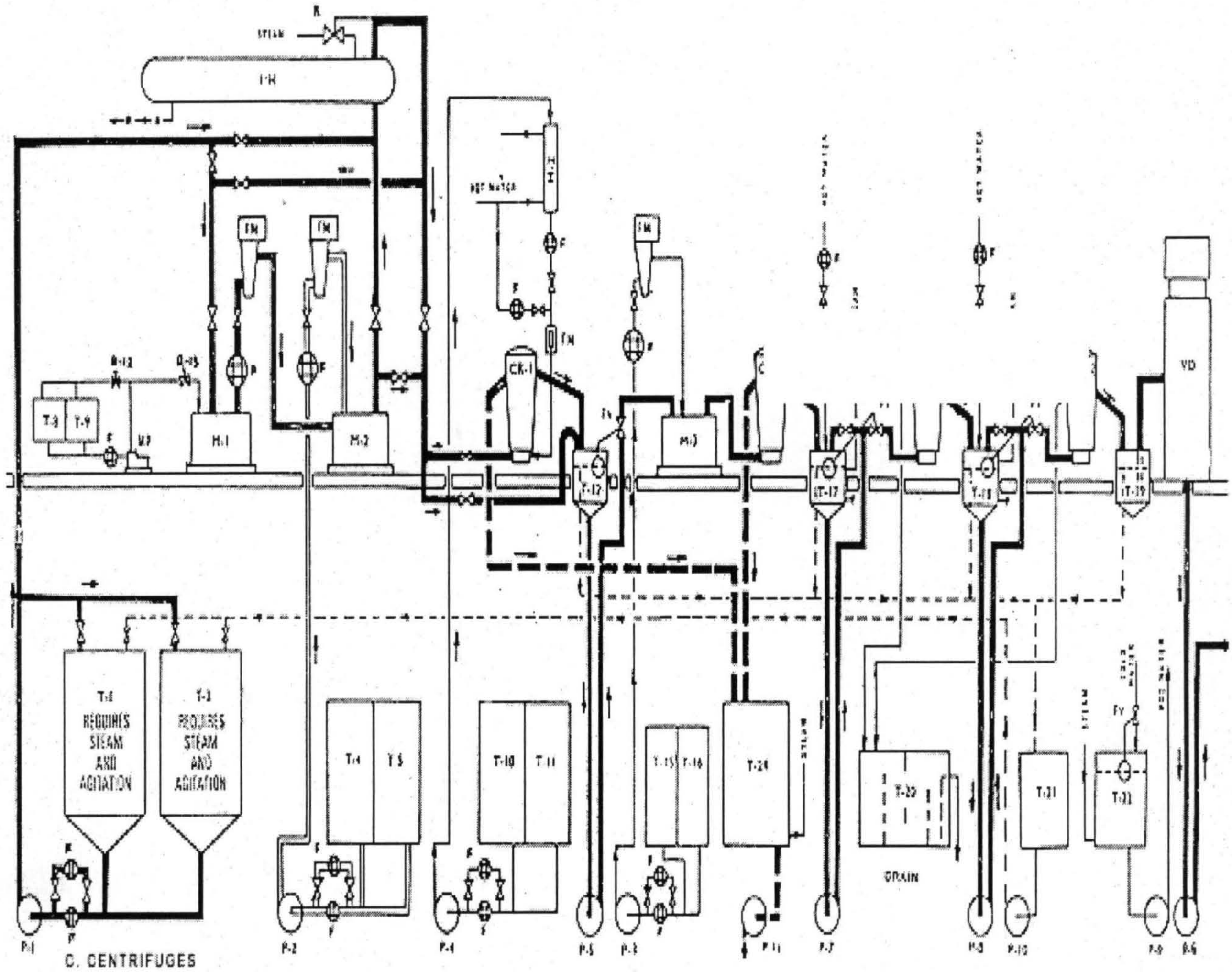


Figure 2.7 Schematic Diagrams showing Dry fractionation of Palm Oil

Others are feed flow rate for reactors, residence time, surface area etc. It is a pertinent fact that design specifications of the various units of equipment determine the unit cost of this equipment in plant design. Consequently, the total capital investment of the plant is greatly influenced. However, detailed design of the oil Palm refinery and fractionation is not considered as this is beyond the scope of this research. Therefore, only important few design specifications will be used in the cost estimation of the plant.



C. CENTRIFUGES
CR1-REFINING CENTRIFUGE
CR2 RE-REFINING CENTRIFUGE
CW1-FIRST WATER WASH CENTRIFUGE
CW2-SECOND WATER WASH CENRIFUGE

F. FILTERS OR STRAINERS

FM -FLOWMETERS
H- HEATERS
H-1 CRUDE DIL HEATER
H-2 FLUSH REAGENT HEATER

M- MIXERS
M-1 MIXER, CUM COND.
M-2 MIXER, REFINING
M-3 MIXER RE-REFINING

MP-METERING PUMP

P- PUMPS
P-1 CRUDE OIL
P-2 REFINING REAGENT
P-3 RE-REFINING REAGENT
P-4 FLUSH REAGENT
P-5 REFINED OIL
P-6 DRY OIL
P-7 FIRST WATER WASH
P-8 SECOND WATER WASH
P-9 HOT WATER
P-10 SUMP
P-11 SOAP STOCK

R- REGULATORS

T- TANKS
T-1,2 CRUDE OIL
T-4,5 REFINING REAGENT
T-8,9 GUM CONDITIONING REAGENT
T-10,11 FLUSH REAGENT
T-12 REFINED OIL
T-15,16 RE-REFINING REAGENT
T-17 FIRST WATER WASH
T-18 SECOND WATER WASH
T-19 VACUUM DRIER FEED
T-20 WASTE WATER
T-21 SUMP
T-22 HOT WATER
T-23 SOAP STOCK

VD- VACUUM DRIER
V VALVES
FV- FLOAT VALVES

Figure 2.8 Schematic Diagram of the Units of Crude Palm oil Refinery

Table 2.4 Relevant Cost Data and Specifications for the Units of the Palm Oil Refinery and Fractionation Plant as at 1996

Piece of Equipment (S.C.R.M)	Historical Cost (N) (S.C.R.M)	Capacity of equipment A(1) (S.C.R.M)	Capacity of equipment B(2) (S.C.R.M)	Exponent (E_i) (Peters and Timmerhaus, 2002)
Calcium carbonate tank	78773	$2m^3$	$4m^3$	0.49
Mixing tank	78773	$2m^3$	$4m^3$	0.49
Phosphoric acid tank	98200	$2.5m^3$	$5m^3$	0.47
Drier	103537	$3m^3$	$6m^3$	0.47
Bleaching earth tank	104604	$3m^3$	$6m^3$	0.47
Continuous bleaching reactor	640436	$3m^3$	$6m^3$	0.76
Bernardinni filter	693806	$15m^3$	$30m^3$	0.54
Steel super filter	117413	$0.2\ m^3$	$0.4\ m^3$	0.66
Guard filter 1	111596	$2.5\ m^3$	$5\ m^3$	0.66
Decanter	66712	$3\ m^3$	$6\ m^3$	0.66
Storage tank	105031	$2.5\ m^3$	$5\ m^3$	0.49
Deaerator/Drier	111649	$2.5\ m^3$	$5\ m^3$	0.49
FFA Recuperator	98200	$2.5\ m^3$	$5\ m^3$	0.49
Preheating tank	104604	$3\ m^3$	$6\ m^3$	0.49
Vacuum system	39368	$0.4\ m^3$	$0.8\ m^3$	0.66
Screw worm	329825	10 m	20 m	0.99
Pump	200136	$9.5x10^{-3}m^3/s$	$19.5x10^{-3}m^3/s$	0.33
Tube and shell heat exchanger	1050316	$14\ m^3$	$28\ m^3$	0.66
Guard filter II	111596	$0.0335\ m^3$	$0.067\ m^3$	0.66
Crystallizer	144098	$0.314\ m^3$	$0.628\ m^3$	0.37
R.B.D. tank	104605	$3\ m^3$	$6\ m^3$	0.49
Cold water tank	96065	$2.5\ m^3$	$5\ m^3$	0.49
Warm water tank	96065	$2.5\ m^3$	$5\ m^3$	0.49
Welders' filter press	960654	$9.29\ m^3$	$18.58\ m^3$	0.66
Stearin tank	104605	$3\ m^3$	$6\ m^3$	0.47
Olein tank	104605	$3\ m^3$	$6\ m^3$	0.47

Table 2.5 Ratio Factor for Estimating Capital Investments Item Based on Delivered Equipment cost (Peters and Timmerhaus, 2002)

IT EM	FACTOR
Direct costs	
Delivered purchased equipment	1.00
Purchased equipment installation	0.47
Instrument and control	0.18
Piping (including services)	0.66
Building (including service)	0.18
Electrical (installed)	0.11
Yard improvements	0.10
Service facilities (installed)	0.70
Land	0.06
Indirect costs	
Engineering and supervision	0.33
Construction expenses	0.41
Contractor's fees	0.21
Contingency	0.42

2.5 Cost Indices

The cost data that are available in most cases for the immediate use are mostly from historic data. Updating of these cost data is therefore necessary since prices change with time consequent upon changes in economic conditions. This is mostly accomplished by the use of cost indexes.

A cost index is a number showing the ratio of the present index value to the index value applicable when the original cost was obtained. If a specific data can be given for a past price, the present price can be determine as follows,

$$\text{Present cost} = \text{original cost}\left(\frac{\text{Index value at present time}}{\text{Index value at time original cost was obtained}}\right) \quad \ldots 2.1$$

Cost indexes can be used to give a general estimate, but no index can take into account all factors such as special technological advancements or local conditions. The common indexes permit fairly accurate estimates if the time period involved is less than or equal to ten years. The most common of these methods are the Marshall Swift all-industry and process-industry equipment indexes, the Engineering News-Record construction index, the Nelson-Farrar refinery construction index and the Chemical Engineering plant index.

2006 Nelson Farrar refinery cost indexes
Carbon steel exchangers - 1079.2

Pumps and compressors - 1736.9

Refinery index - 1971.3

Table 2.6 Chemical Engineering Plant Cost Index Values (2000 – 2006)

Year	CEPI Value
2000	394.1
2001	394.3
2002	390.4
2003	398.2
2004	457.4
2005	468.2
2006	492.6

CHAPTER THREE

METHODOLOGY

3.1 Development of Database

A Database is developed using a Structural Querying Language (SQL) server where tables are created for each calculated and inputted parameters. This is then inserted in each selected table so as to be recalled when needed in subsequent computation or use.

3.2 Development of Equations

3.2.1 Equipment cost estimation

There are three basic methods for estimating plant equipment cost namely:

1. Method of Cost index
2. Six -tenth -Factor Rule
3. Cost Index and Scaling Method

But two of these are used in this research work. the Six-tenth-Factor Rule and the Cost Index and Scaling Method

3.2.1.1 Ratio of indices

This is the ratio of the current and historic chemical engineering plant cost indices. This ratio is common in all the methods. Factoring it reduces input of parameters per unit equipment costing and this is achieved by simply substituting the value of the ratio inserted in the database into the equation when computing the cost for each of the equipment.

Let CurIndex represent the current chemical plant cost index and HisIndex represent historic chemical plant cost index. Therefore,

Ratio Index (ri) = CurIndex / HisIndex 3.1

3.2.1.2 Method of cost index

The method can be use to determine the present cost of a unit equipment when the cost in the past at a particular time is known. This original cost will be multiplied by the ratio of the present index value to the index value applicable at the time of initial cost.

Mathematical presentation of the method

Let original/initial cost of the unit equipment be historic cost.

Let CurIndex represent the current chemical plant cost index and HisIndex represent historic chemical plant cost index. Therefore,

$$\text{Present cost} = \text{historic cost}\left(\frac{\text{CurIndex.}}{\text{HisIndex}}\right) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad 3.2$$

3.2.1.3 Six -tenth factor rule

Cost of a piece of equipment may be estimated when cost data are not available for the particular size of operational capacity involved. This can be achieved by using the logarithmic relationship called six-tenths-factor rule (Peters and Timmerhaus, 2002) if the new piece of equipment is similar to one of another capacity for which cost data are available. According to this rule, if the cost of a given unit at one capacity is known, the cost of a similar unit that is A times the capacity of the first is approximately $(A)^{0.6}$ times the cost of the one with known size.

Mathematical presentation of the rule.

Let historical cost of the equipment B = historic cost B (desired)

Let historical cost of equipment A= historic cost A

QA be known capacity of equipment A

QB be known capacity of equipment B

ri = CurIndex / HisIndex (From equation 3.1)

Where, curindex is the current index

HisIndex is the historical index

ri is the ratio of the indexes

If C = QB/QA 3.3

Therefore, the historical cost of equipment B can be given as,

Historic cost B = historic cost A$[C]^{0.6}$.. 3.4

Also, the present /current cost of equipment A can the be given (from equation 3. 2) as,

Current cost A = historic cost A [ri].. 3.5

Likewise,

The current cost B = Current cost A $[C]^{0.6}$ 3.6

Combining equations 3.1 and 3.6 gives

The current cost B = historic cost A [ri] $[C]^{0.6}$.................................... 3.7

The application of the 0.6 factor rule for most purchased equipment is a simplification of valuable cost concept since the actual value of the cost capacity factor varies from 0.2 to 1.0. However, the cost capacity concept should not be used beyond a tenfold range of capacity (Peters and Timmerhaus, 2002).

3.2.1.4 Cost index and scaling method

Unlike the six-tenth factor rule, cost index and scaling method are used when cost data for a particular size of operational capacity of the same equipment is available, but lack the desired size. Moreover, the exponential constant 0.6 is replaced by a variable E_i.

The present (current) cost of a piece of equipment of desired size B by cost index and scaling method is given (if all the parameters are as defined above) by:

The current cost B = historic cost A [ri} $[C]^{E_i}$.................................... 3.8

Where E_i is the variable exponential which is the slope of correlation curve on the log-log cost.

3.2.2 Estimating delivered purchased equipment cost

The cost of purchased equipment is the basis of several pre-design methods for estimating capital investment. Delivered purchased equipment cost is said to be the summation of the cost of all purchased equipments involved in plant design.

Mathematical presentation of Delivered purchased Equipment cost

$$Pect = \sum_{i=1}^{n} cc_i \quad \text{...} \quad 3.9$$

3.2.3 Construction of Pie chart

The current cost of the four equipments, crystallizer, pump, heat exchanger and olein tank are displayed graphically in a pie chart by using the following equation

Let Crsti, hei, pui and oti be current cost of crystallizer, heat exchanger pump and olein tank.

K is the sum of the current cost of the equipment.

$$K = Crsti + hei + pui + oti \quad \text{..} \quad 3.10$$

Therefore, portion of each equipment in the chart is given as,

$$Portion = (V/K) \times 360 \quad \text{...} \quad 3.11$$

Where V is the current cost of each of the equipment

3.2.4 Fixed capital investment

Manufacturing fixed capital investment represents the capital necessary for the installed process equipment with all auxiliaries that are needed for complete process operation. Expenses for piping, instruments, insulation, foundations and site preparation are typical examples of cost included in the manufacturing fixed capital investment.

However, plant components such as land, factory, administrative blocks (offices), warehouses, laboratories, transportation, shipping and receiving facilities, utility and waste-disposal facilities, shops and other permanent part of the plant are referred to as non-manufacturing fixed capital investment. These are fixed capitals that are basically needed for construction overhead and for all plant components that are not directly related to the process operation (Peters and Timmerhaus, 2002).

The determination of fixed capital investment is done by applying factored estimate method, which is the method by which the investment cost in a complete system can be extrapolated from the delivered cost of the major items of processing equipment (Rudd and Watson, 1968).

It has been observed that the cost of other essential items needed to complete the process system can be correlated with the investment cost in major items of equipment and the capital investment can be estimated by the application of experience factor to the base investment or delivered purchased equipment cost (Rudd and Watson, 1968). The experience factor f is obtained in a similar process system. The values of these factors for both direct and indirect costs are given in Table 2.5 above.

Mathematical presentation of fixed capital investment.

Let sum be the sum of indirect cost factors

Sum$_l$ be the sum of direct cost factors

$f_l = 1 + sum$ 3.12

Thus,

$Fcapn = (Pect + (sum1 \times Pect)) \times f_l$ 3.13

Where,

Fcapn is the fixed capital investment

Pect is the Delivered purchased Equipment cost

3.2.5 Working capital investment

The working capital for an industrial plant can be referred to as the total amount of money invested on raw materials and supplies in stock, the finished product in stock and semi finished products in the process of being manufactured, the account receivable, cash kept in hand for monthly payment of operating expenses, such as salaries and wages, raw material purchased and others that are necessary for the operation of the plant.

The ratio of working capital to that of total capital investment varies with different companies. However, most chemical plants use an initial working capital amounting to 10 to 20 percent of the total capital investment.

Mathematical presentation of Working capital investment.

Since the fixed capital investment was taken as 80 percent of the total capital investment

Then,

Wcapn = 0.20 x Tcapn 3.14

Where, Wcapn is the Working capital investment, Tcapn is the total capital investment.

3.2.6 Total capital investment

This is the total cost needed to start a particular operation such as oil palm refinery and fractionation plant as applicable.

It includes all the funds necessary to get the project underway and this encompasses the regular manufacturing fixed capital investment along with the investment required for necessary auxiliaries and non-manufacturing facilities.

Mathematical presentation of total capital investment.

If the fixed capital investment is taken as 80 percent of the total capital investment,

Thus,

3.4.1 Program CIS

This program computes the current cost of each of the equipment by using cost index and scaling method. The program consists of thirty one computer cost modules called class and a structural querying language (SQL) database file (refiningofoil) where various results are stored and recalled for subsequent use.

The program starts execution with the class called projection which confirms the reference and current year for the cost indexes. While the next class called ratio index computes the ratio of the inputted indexes using equation one, and store the result in database.

Each of the subsequent twenty six classes computes the current cost for each of the twenty six equipments using equation 3.8 for cost index and scaling method.

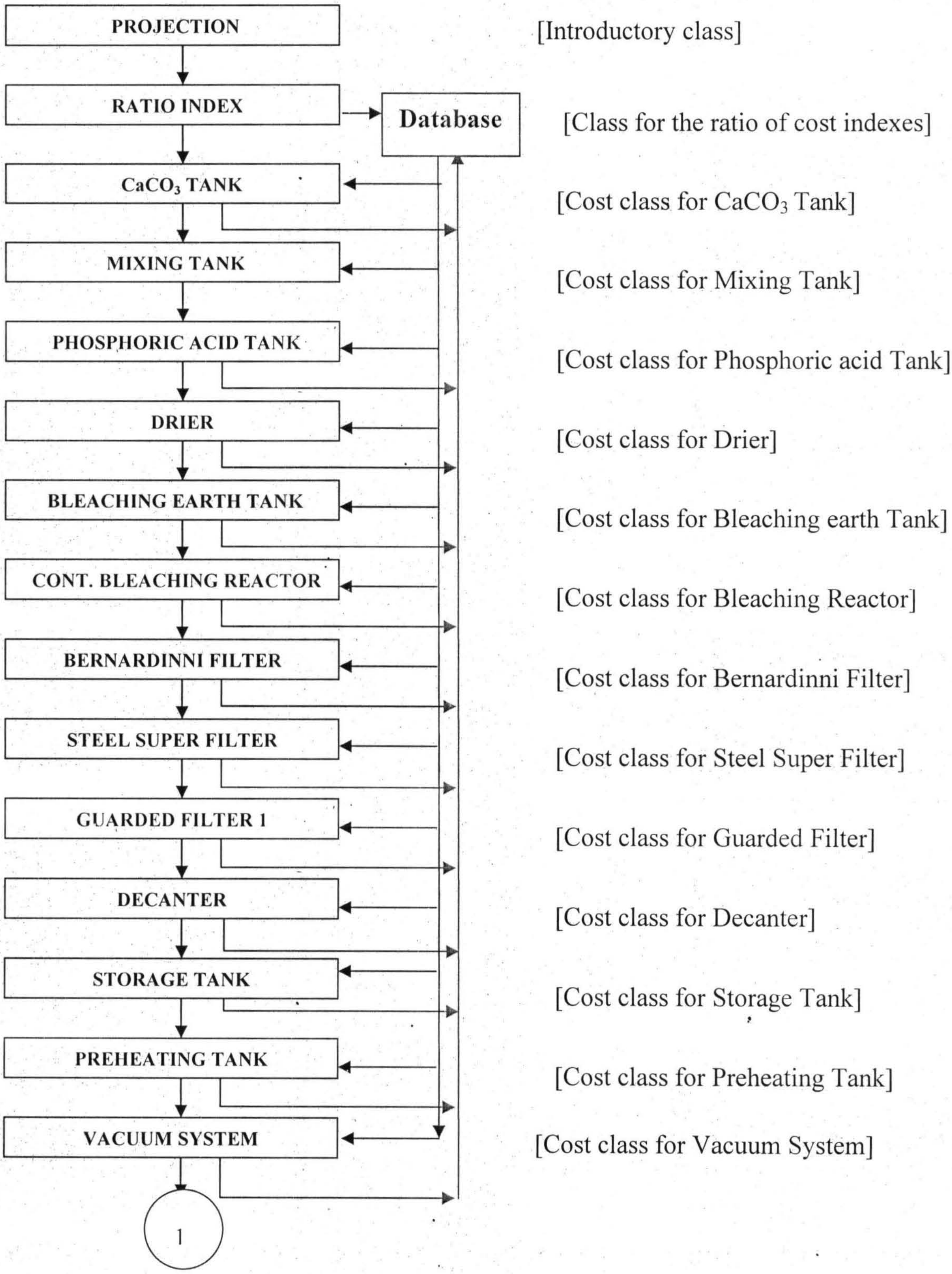
The results are then inserted into the database. This is followed by the class delivered which computes the delivered purchased equipment cost by using equation 3.9 and store the results in the database.

Finally the class ratio factor which computes the fixed capital investment, working capital investment, total capital investment and the graphical equipment cost displays.

3.4.2 Program 610FR

This program also computes the current cost of each of the equipment but it does this by using equation 3.7 which is for six-tenth factor rule method. The program consist of thirty one Java classes as well and a structural querying language (SQL) database file(refiningofoil) where various results are stored and recalled for subsequent use.

The execution of the program follows the same pattern as the first program and this is as shown in the flow chart below.


PROJECTION
[Introductory class]
RATIO INDEX
Database
[Class for the ratio of cost indexes]
CaCO3 TANK
[Cost class for CaCO3 Tank]
MIXING TANK
[Cost class for Mixing Tank]
PHOSPHORIC ACID TANK
[Cost class for Phosphoric acid Tank]
DRIER
[Cost class for Drier]
BLEACHING EARTH TANK
[Cost class for Bleaching earth Tank]
CONT. BLEACHING REACTOR
[Cost class for Bleaching Reactor]
BERNARDINNI FILTER
[Cost class for Bernardinni Filter]
STEEL SUPER FILTER
[Cost class for Steel Super Filter]
GUARDED FILTER 1
[Cost class for Guarded Filter]
DECANTER
[Cost class for Decanter]
STORAGE TANK
[Cost class for Storage Tank]
PREHEATING TANK
[Cost class for Preheating Tank]
VACUUM SYSTEM
[Cost class for Vacuum System]
1

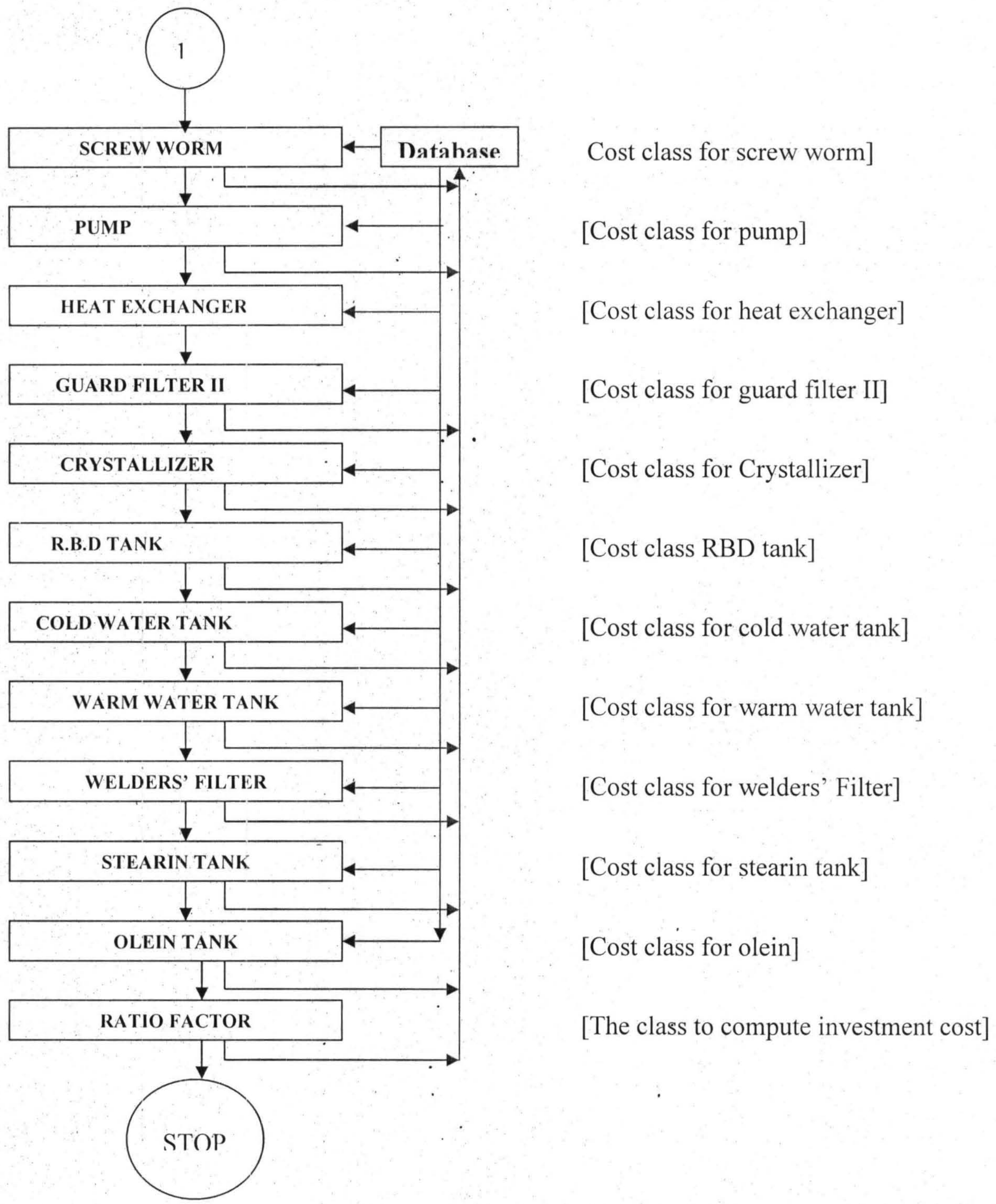


Figure 3.1 Flow charts for the two programs

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 RESULTS

The cost datas and other pertinent design specifications of the plant shown in Table 2.5 and 2.6 were used as historical data to obtain the current cost implications for the plant using the two developed modules, the CIS and 610FR. Various results were obtained through a stepwise functional execution of each class in the program as listed below.

4.1.1 Program listing

There are thirty one sub-modules in each program, each with its different function. The modules are similar and the figures below show a typical module and its flow sheets for CIS. While that of 610FR is in the CD that accompanies the thesis.

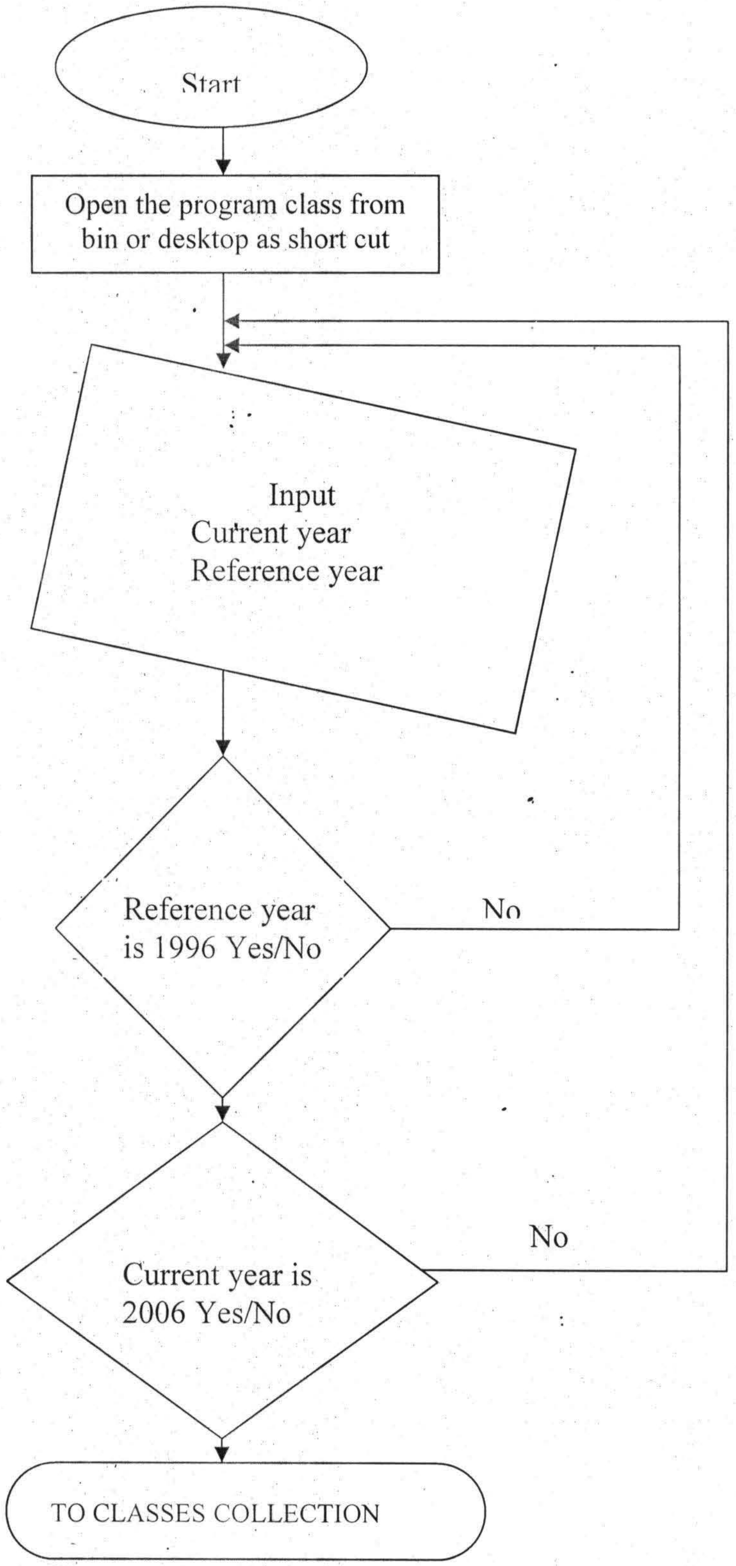


Figure 4.1 Flow Chart to show Introductory Class (Projection)

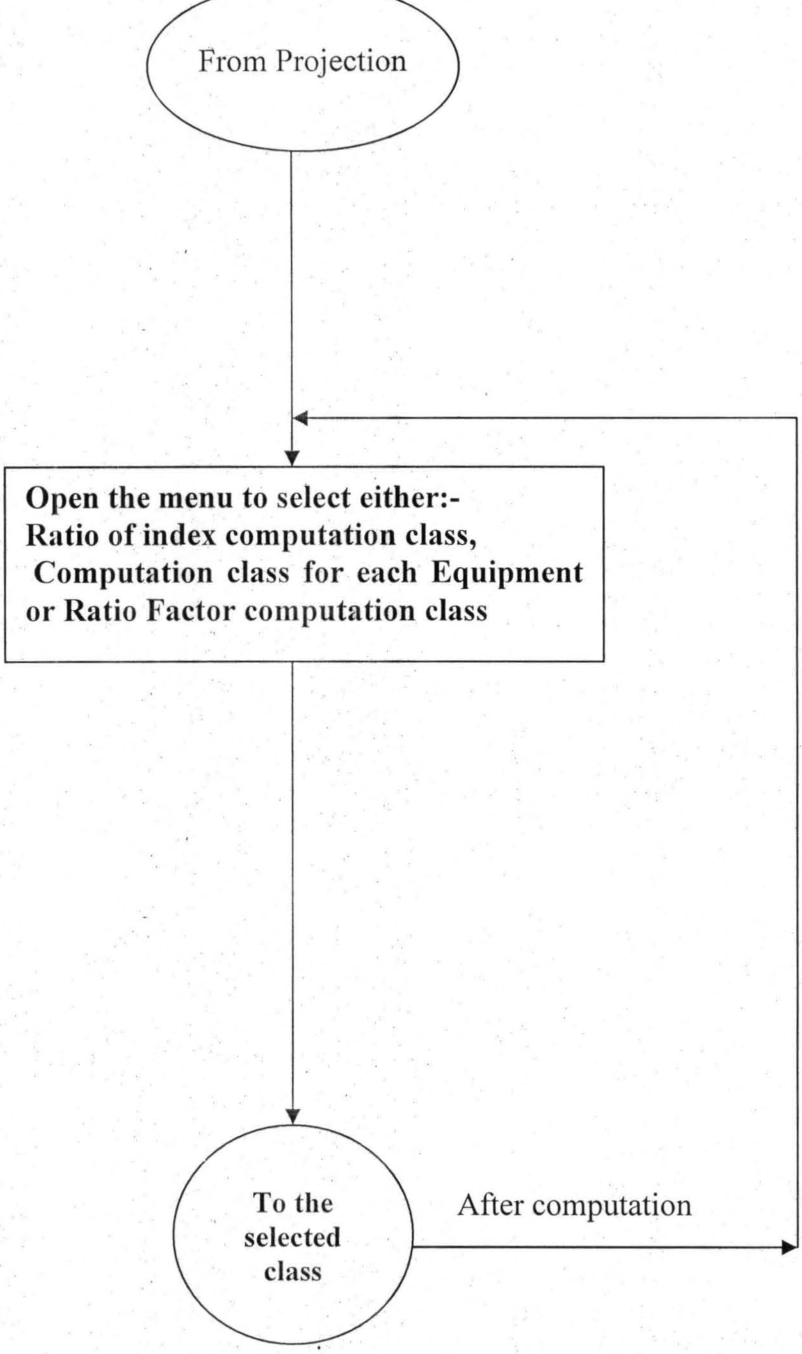


Figure 4.2 Flow Chart to Show the Collection of Class

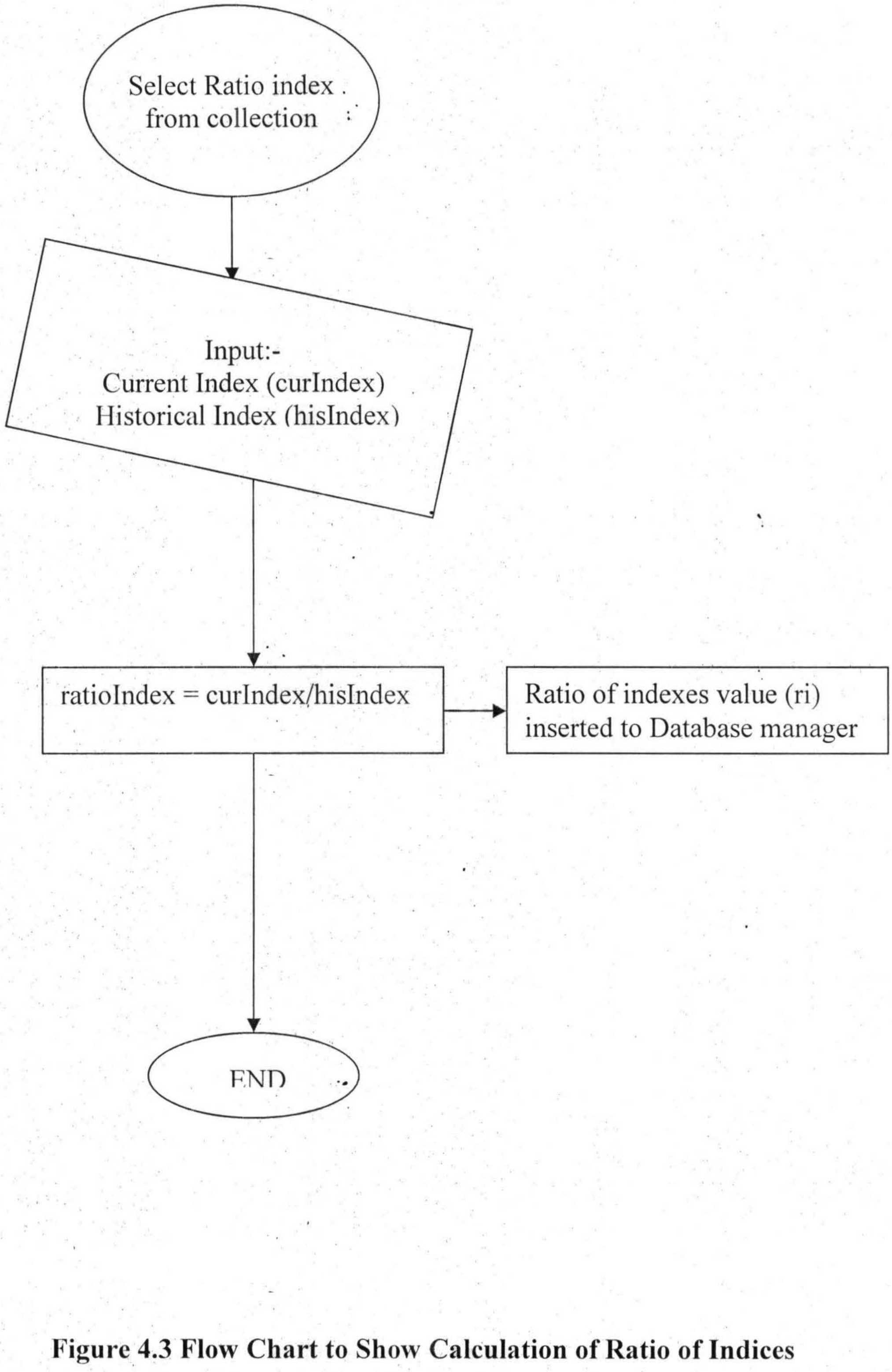


Figure 4.3 Flow Chart to Show Calculation of Ratio of Indices

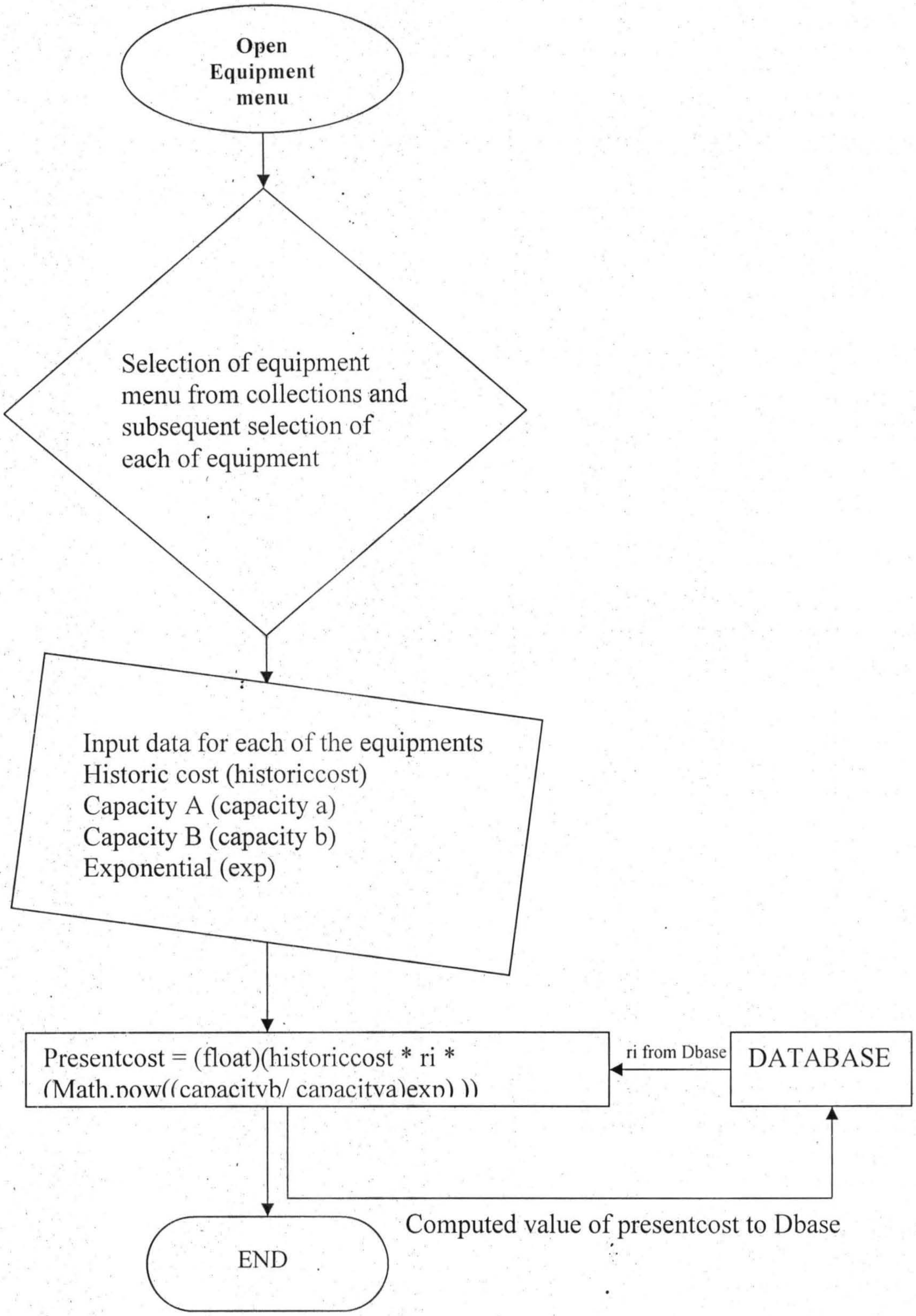


Figure 4.4 Flow Chart to Show Calculation of Current Cost of each of the Equipment

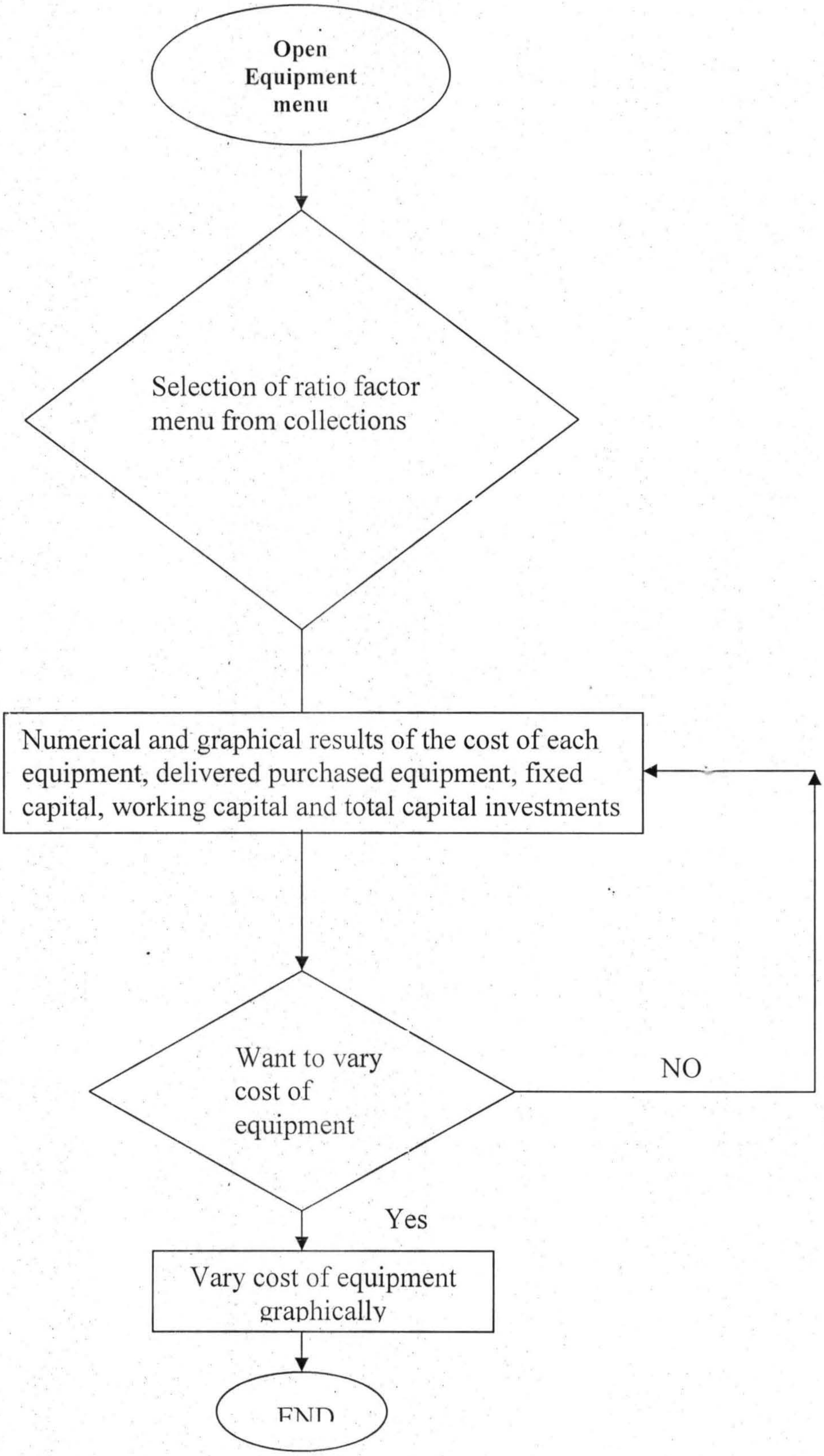


Figure 4.5 Flow Chart to show Estimation of Investment Cost

PROGRAM CIS

```
// PROGRAM CIS REPRESENTING COST INDEX AND SCALING METHOD.
// This program is developed towards the computation of equipment cost, estimation of
// capital investments both graphically and numerically for an oil Palm refinery and
// fractionation plant.
// Developer        : - POPOOLA AYOBAMI OLANREWAJU
// NO               : - MENG/SEET/2005/1142
// PROJECT TITLE: - DEVELOPMENT OF SOFTWARE FOR ESTIMATING THE
// COST OF PALM OIL REFINERY AND FRACTIONATION PLANT.
// SUPERVISOR   : - PROFESOR K.R ONIFADE
// LANGUAGE     : - JAVA
// DEFINITION OF SUBCLASSES AS USED IN THE PROGRAM.
// Class projection                   Introductory /validation subclass.
// Class Collections                  The main class which house other subclasses.
// Class RatioIndex                   The subclass to compute the value of ratio of indexes.
// Class CalciumCarbonateTank         The subclass that compute cost of Calcium
//                                    Carbonate Tank.
// Class MixingTank                   The subclass that compute cost of Mixing Tank.
// Class PhosphoricAcidTank           The subclass that compute cost of Phosphoric Acid
//                                    Tank.
// Class Drier                        The subclass that compute cost Drier.
// Class BleachingEarthTank           The subclass that compute cost of Bleaching Earth
//                                    Tank.
//Class ContinuousBleachingReactor    The subclass that compute cost of Continuous
//                                    Bleaching Reactor.
// Class BernardinniFilter            The subclass that compute cost of Bernardinni Filter
// Class SteelSuperFilter             The subclass that compute cost of Steel Super Filter
// Class GuardFilter1                 The subclass that compute cost of class Guard Filter1.
// Class Decanter                     The subclass that compute cost of Decanter.
// Class StorageTank                  The subclass that compute cost of Storage Tank.
// Class Deaerator                    The subclass that compute cost of Deaerator.
// Class FFARecuperator               The subclass that compute cost of FFA Recuperator
// Clas PreheatingTank                The subclass that compute cost of Preheating Tank.
// Class VacuumSystem                 The subclass that compute cost of Vacuum System.
// Class SrewWorm                     The subclass that compute cost of Screw Worm.
// Class Pump                         The subclass that compute cost of Pump.
// Class HeatExchangers               The subclass that compute cost of Heat Exchangers.
// Class GuardFilterII                The subclass that compute cost of Guard Filter II.
// Class Crystallizers                The subclass that compute cost of Crystallizer.
// Class RBDTank                      The subclass that compute cost of RBD Tank.
// Class ColdWaterTank                The subclass that compute cost of Cold Water Tank.
// Class WarmWaterTank                The subclass that compute cost of Warm Water Tank.
// Class WeldersFilter                The subclass that compute cost of Welders Filter.
// class StearinTank                  The subclass that compute cost of Stearin Tank.
// class OleinTank                    The subclass that compute cost of Olein Tank.
```

// class RatioFactor — The subclass that compute deliver equipment cost,
// — capital investments and graphical interaction.

Introductory /validation subclass.

```
// Java (Application Programming interfaces) APIs
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
// Implements ActionListener for Event handling
public class Projection implements ActionListener
{
// control declarations
JFrame frame;
JLabel lreference;
JLabel lcurrent;
JTextField treference;
JTextField tcurrent;
JPanel panel, panel1,panel2;
JButton button;
GridBagLayout gbl;
GridBagConstraints gbc;
public Projection()
{
//Initializing controls
lreference = new JLabel("Input Reference Year:          ");
lcurrent = new JLabel("Input Current Year:           ");
treference = new JTextField(5);
tcurrent = new JTextField(5);
button = new JButton("Enter");
frame = new JFrame("Data input");
panel = new JPanel();
panel1 = new JPanel();
panel2 = new JPanel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
//arranging controls controls in the panel
panel.setLayout(gbl);gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(lreference,gbc);panel.add(lreference);gbc.anchor=gbc.WEST;
gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(treference,gbc);panel.add(treference);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;gbl.setConstraints(lcurrent,gbc);
panel.add(lcurrent);gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;gbl.setConstraints(t
current,gbc);panel.add(tcurrent);gbc.anchor=gbc.EAST;gbc.gridx=7;gbc.gridy=8;
gbl.setConstraints(button,gbc);panel1.add(button);
```

```
button.addActionListener(this);panel2.setLayout(gbl);gbc.anchor=gbc.CENTER;gbc.grid
x=4;gbc.gridy=4;gbl.setConstraints(panel,gbc);panel2.add(panel);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=8;gbl.setConstraints(panel1,gbc);
panel2.add(panel1);frame.getContentPane().add(panel2);
frame.setSize(700,500);frame.setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
Object obj = ae.getSource();
if(obj == button)
{
int refyear = Integer.parseInt(treference.getText());
int curyear = Integer.parseInt(tcurrent.getText());
if(refyear == 1996 && curyear == 2006)
{
frame.setVisible(false);
new Collections();
}
else
{
JOptionPane.showMessageDialog(null,"Invalid
entery!","",JOptionPane.ERROR_MESSAGE);
}
}
}
public static void main(String dd[])
{
new Projection();
}
}
```

The main class which house other subclasses.

```
// Java (Application Programming interfaces) APIs
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
//Implements ActionListener for Event handling
public class Collections extends JFrame{
JMenu ratioIndex;
JMenu equipments;
JMenu exi;
JMenu ratioFactor;
JPanel p1,p2,p3,p4,p5;
Icon sp=new ImageIcon("JPEG image ");
JLabel ll = new JLabel(sp);
GridBagLayout gbl;GridBagConstraints gbc;Font font,font1,font2;
```

```
JLabel ui = new JLabel("    PROGRAM CIS ");
JLabel il = new JLabel("Implementing cost index and scalling method");
JLabel pr1 = new JLabel("            ");
JLabel pr2 = new JLabel("            ");
public Collections()
 {
super("PROGRAM CIS");
p1 = new JPanel();p2 = new JPanel();p3 = new JPanel();p4 = new JPanel();p5 = new
JPanel();gbl = new GridBagLayout();gbc = new GridBagConstraints();p1.setLayout(gbl);
p2.setLayout(gbl);p3.setLayout(gbl);p4.setLayout(gbl);p5.setLayout(gbl);
font = new Font("System",Font.BOLD,27);
font1 = new Font("Arial",Font.BOLD,20);
font2 = new Font("Arial",Font.BOLD,13);
ui.setFont(font);
il.setFont(font1);
ui.setForeground(Color.blue);
il.setForeground(Color.blue);
ratioIndex = new JMenu("Ratio Index");
equipments = new JMenu("Equipments");
ratioFactor = new JMenu("Ratio factor");
exi = new JMenu("exit");
JMenuItem
rrf,rrx,exit,cco3,mxt,ppat,drier,bht,cbr,bf,ssf,gf1,decanter,st,dr,ddr,ffa,pht,vx,sw,pp,he,gf
2,cry,rbd,cwt,wwt,wfp,xt,ot;
cco3 = new JMenuItem("Calcium carbonate tank");
mxt = new JMenuItem("Mixing tank");
ppat = new JMenuItem("Phosphoric acid tank");
drier = new JMenuItem("Drier");
bht = new JMenuItem("Bleaching earth tank");
cbr = new JMenuItem("Continuous bleaching tank");
bf = new JMenuItem("Bernardinni filter");
ssf = new JMenuItem("Steel super filter");
gf1 = new JMenuItem("Guard filter1");
decanter = new JMenuItem("Decanter");
st = new JMenuItem("Storage tank");
dr = new JMenuItem("Dearator/Drier");
ddr = new JMenuItem("Deodorizer");
ffa = new JMenuItem("FFA Recuperator");
pht = new JMenuItem("Preheating tank");
vx = new JMenuItem("Vacuum system");
sw = new JMenuItem("Screw worm");
pp =  new  JMenuItem("Pump");
he = new JMenuItem("Tube and shell exchanger");
gf2 = new JMenuItem("Guard filter2");
cry = new JMenuItem("Crystaliser");
 rbd = new JMenuItem("RBD tank");
```

```
cwt = new JMenuItem("Cold water tank");
wwt =  new  JMenuItem("Warm water tank");
wfp = new JMenuItem("Weilders' filter press");
xt = new JMenuItem("Stearin tank");
ot = new JMenuItem("Olein tank");
rrx = new JMenuItem("Ratio Index");
rrf = new JMenuItem("Ratio Factor");
exit = new JMenuItem("Exit");
vx.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
 new VacuumSystem();
 }
 }
 );
sw.addActionListener(
 new ActionListener(){
 public void actionPerformed(ActionEvent e)
 {
 new SrewWorm();
 }
 }
 );
pp.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new Pump();
}
}
 );
he.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new HeatExchangers();
}
}
);
gf2.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new GuardFilter2();
}
```

```
}
);
cry.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new Crytalizers();
}
}
);
rbd.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new RBDStorageTank();
}
}
);
cwt.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new ColdWaterTank();
}
}
);
wwt.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new WarmWaterTank();
}
}
);
wfp.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new WeldersFilter();
}
}
);
xt.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
```

```
new StearinTank();
}
}
);
ot.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new OleinTank();
}
}
);
rrx.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new RatioIndex();
}
}
);
cco3.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new CalciumCarbonateTank();
}
}
);
rrf.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new RatioFactor();
}
}
);
mxt.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new MixingTank();
}
}
);
ppat.addActionListener(
new ActionListener(){
```

```
public void actionPerformed(ActionEvent e)
{
new PhosphoricAcidTank();
}
}
);
drier.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new Drier();
}
}
);
bht.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new BleachingEarthTank();
}
}
);
cbr.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new ContinuousBleachingReactor();
}
}
);
bf.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new BernardinniFilter();
}
}
);
ssf.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
{
new SteelSuperFilter();
}
}
);
```

```
gf1.addActionListener(
  new ActionListener(){
  public void actionPerformed(ActionEvent e)
  {
  new GuardFilter1();
  }
 }
);
decanter.addActionListener(
  new ActionListener(){
  public void actionPerformed(ActionEvent e)
  {
  new Decanter();
  }
 }
);
st.addActionListener(
  new ActionListener(){
  public void actionPerformed(ActionEvent e)
  {
  new StorageTank();
  }
 }
);
exit.addActionListener(
 new ActionListener(){
 public void actionPerformed(ActionEvent e)
  {
System.exit(1);
  }
 }
);
dr.addActionListener(
 new ActionListener(){
 public void actionPerformed(ActionEvent e)
  {
  new Deaerator();
  }
 }
);
ddr.addActionListener(
  new ActionListener(){
  public void actionPerformed(ActionEvent e)
  {
  new Deodorizer();
  }
```

```
}
);
ffa.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
 {
new FFARecuperator();

 }
 }
);
pht.addActionListener(
new ActionListener(){
public void actionPerformed(ActionEvent e)
 {
 new PreheatingTank();

 }
 }
);
JMenuBar bar = new JMenuBar();
setJMenuBar(bar); bar.add(ratioIndex);bar.add(equipments);
bar.add(ratioFactor);bar.add(exi);ratioIndex.add(rrx);equipments.add(cco3);
equipments.add(mxt);equipments.add(ppat);equipments.add(drier);
equipments.add(bht);equipments.add(cbr);equipments.add(bf);
equipments.add(ssf);equipments.add(gf1);equipments.add(decanter);equipments.add(st);
equipments.add(dr);equipments.add(ddr);equipments.add(ffa);equipments.add(pht);
equipments.add(vx);equipments.add(sw);equipments.add(pp);equipments.add(he);
equipments.add(gf2);equipments.add(cry);equipments.add(rbd);equipments.add(cwt);
equipments.add(wwt);equipments.add(wfp);equipments.add(xt);equipments.add(ot);
ratioFactor.add(rrf);exi.add(exit);
Container k = getContentPane();
k.setLayout(new FlowLayout());
getContentPane().add(p1);
gbc.anchor=gbc.WEST;
gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(ll,gbc);p2.add(ll);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(ui,gbc);p3.add(ui);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(il,gbc);p3.add(il);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(pr1,gbc);p4.add(pr1);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(pr2,gbc);p5.add(pr2);
gbc.anchor=gbc.CENTER;
```

```
gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(p4,gbc);
p1.add(p4);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(p2,gbc);p1.add(p2);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(p5,gbc);p1.add(p5);
gbc.anchor=gbc.CENTER;gbc.gridx=1;gbc.gridy=14;
gbl.setConstraints(p3,gbc);
p1.add(p3);
 setSize(1024,1024); setVisible(true);


 }
public static void main(String args []){
  Collections application = new Collections();
  application.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}
```

Subclass to compute the value of ratio of indexes.

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.sql.*;
//Implements ActionListener for Event handling
public class RatioIndex implements ActionListener
{
//control declarations
static JFrame frame;
JLabel lcurrentIndex;
JLabel lhistoricIndex;
JTextField tcurrentIndex;
JTextField thistoricIndex;
JPanel panel, panel1,panel2;
JButton button;
GridBagLayout gbl;
GridBagConstraints gbc;
float curIndex, hisIndex, ratioIndex;
Connection con;
PreparedStatement stat;
Statement stat1;
ResultSet rs;
public RatioIndex()
{
```

```
//Initializing controls
lcurrentIndex = new JLabel("Input Current Index:          ");
lhistoricIndex = new JLabel("Input Historic Index:          ");
tcurrentIndex = new JTextField(10);
thistoricIndex = new JTextField(10);
button = new JButton("Calculation of ratio of index");
frame = new JFrame("Ratio index calculation");
panel = new JPanel();
panel1 = new JPanel();
panel2 = new JPanel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
//arranging controls controls in the panel
panel.setLayout(gbl);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(lcurrentIndex,gbc);
panel.add(lcurrentIndex);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(tcurrentIndex,gbc);
panel.add(tcurrentIndex);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(lhistoricIndex,gbc);
panel.add(lhistoricIndex);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(thistoricIndex,gbc);
panel.add(thistoricIndex);gbc.anchor=gbc.EAST;gbc.gridx=7;gbc.gridy=8;
gbl.setConstraints(button,gbc);
panel1.add(button);
//Activating the Enter button
button.addActionListener(this);
panel2.setLayout(gbl);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(panel,gbc);
panel2.add(panel);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(panel1,gbc);
panel2.add(panel1);
frame.getContentPane().add(panel2);
frame.setSize(700,500);
frame.setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
Object obj = ae.getSource();
if(obj == button)
{
```

```
curIndex = Float.parseFloat(tcurrentIndex.getText());
hisIndex = Float.parseFloat(thistoricIndex.getText());
ratioIndex = curIndex/hisIndex;
String ok = String.valueOf(ratioIndex);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con.createStatement();
stat1.executeUpdate("delete from RatioIndex");
con.close();
}
catch(Exception e)
{
}
JOptionPane.showMessageDialog(null,ok,"",JOptionPane.ERROR_MESSAGE);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat = con.prepareStatement("insert into ratioindex values(?,?,?)");
stat.setFloat(1,curIndex);stat.setFloat(2,hisIndex);stat.setFloat(3,ratioIndex);
stat.executeUpdate();
con.close();
}
catch(Exception e)
{
}
JOptionPane.showMessageDialog(null,ok,"",JOptionPane.ERROR_MESSAGE);
}
}
public void rindex1()
{
ratioIndex = ratioIndex;
}
public static void main(String dd[])
{
new RatioIndex();

}
}
```

```
curIndex = Float.parseFloat(tcurrentIndex.getText());
hisIndex = Float.parseFloat(thistoricIndex.getText());
ratioIndex = curIndex/hisIndex;
String ok = String.valueOf(ratioIndex);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con.createStatement();
stat1.executeUpdate("delete from RatioIndex");
con.close();
}
catch(Exception e)
{
}
JOptionPane.showMessageDialog(null,ok,"",JOptionPane.ERROR_MESSAGE);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat = con.prepareStatement("insert into ratioindex values(?,?,?)");
stat.setFloat(1,curIndex);stat.setFloat(2,hisIndex);stat.setFloat(3,ratioIndex);
stat.executeUpdate();
con.close();
}
catch(Exception e)
{
}
JOptionPane.showMessageDialog(null,ok,"",JOptionPane.ERROR_MESSAGE);
}
}
public void rindex1()
{
ratioIndex = ratioIndex;
}
public static void main(String dd[])
{
new RatioIndex();

}
}
```

Subclass that compute cost of Calcium Carbonate Tank.

```
// Java Application Programming interfaces (APIs)
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.sql.*;
//Implements ActionListener for Event handling
public class CalciumCarbonateTank implements ActionListener
{
Connection con;
PreparedStatement stat;
Statement stat1;
ResultSet rs;
//control declarations
JFrame frame;JLabel title;JLabel lhistoricCost;JLabel lcapacityb;JLabel lcapacitya;
JLabel lexp;JTextField thistoricCost;JTextField tcapacityb,tcapacitya,texp;
JPanel panel, panel1,panel2,panel3;JButton button;
GridBagLayout gbl;
GridBagConstraints gbc;
float presentcost,historiccost,capacityb,capacitya,exp;
Font font1;
float ri;
public CalciumCarbonateTank()
{
//Initializing controls
font1 = new Font("Arial",Font.BOLD,17);
lhistoricCost = new JLabel("Input Historic cost of Calcium carbonate tank : ");
lcapacityb = new JLabel("Input Capacity of equipment B: ");
lcapacitya = new JLabel("Input Capacity of equipment A: ");
lexp = new JLabel("Input Exponential: ");
thistoricCost = new JTextField(10);tcapacityb = new JTextField(10);
tcapacitya = new JTextField(10);texp = new JTextField(10);
title = new JLabel("Computation of the current cost of calcium carbonate");
title.setFont(font1);
title.setForeground(Color.blue);
button = new JButton("Compute");
frame = new JFrame("Computation of the current cost of calcium carbonate");
panel = new JPanel();
panel1 = new JPanel();
panel2 = new JPanel();
panel3 = new JPanel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
//arranging controls controls in the panel
panel.setLayout(gbl);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
```

```
gbl.setConstraints(title,gbc);
panel3.add(title);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(lhistoricCost,gbc);
panel.add(lhistoricCost);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(thistoricCost,gbc);
panel.add(thistoricCost);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(lcapacityb,gbc);
panel.add(lcapacityb);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(tcapacityb,gbc);
panel.add(tcapacityb);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(lcapacitya,gbc);
panel.add(lcapacitya);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(tcapacitya,gbc);
panel.add(tcapacitya);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=12;
gbl.setConstraints(lexp,gbc);
panel.add(lexp);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=12;
gbl.setConstraints(texp,gbc);
panel.add(texp);
gbc.anchor=gbc.EAST;gbc.gridx=7;gbc.gridy=8;
gbl.setConstraints(button,gbc);
panel1.add(button);
//Activating the Enter button
button.addActionListener(this);
panel2.setLayout(gbl);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(panel3,gbc);
panel2.add(panel3);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(panel,gbc);
panel2.add(panel);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(panel1,gbc);
panel2.add(panel1);
frame.getContentPane().add(panel2);
frame.setSize(700,500);// setting frame size
frame.setVisible(true);// setting frame visible
}
public void actionPerformed(ActionEvent ae)
```

```
{
Object obj = ae.getSource();
if(obj == button)
{
historiccost = Float.parseFloat(thistoricCost.getText());
capacityb = Float.parseFloat(tcapacityb.getText());
capacitya = Float.parseFloat(tcapacitya.getText());
exp = Float.parseFloat(texp.getText());
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con.createStatement();
rs=stat1.executeQuery("select ratioindex from ratioindex");
while(rs.next())
{
ri = rs.getFloat(1);
}
con.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);

}
float val = capacityb/capacitya;
presentcost = (float)(historiccost * ri * (Math.pow((val),exp) ));
String pc = String.valueOf(presentcost);
JOptionPane.showMessageDialog(null,pc,"",JOptionPane.ERROR_MESSAGE);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con.createStatement();
stat1.executeUpdate("delete from costOfCalciumCarbonateTank");

con.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat = con.prepareStatement("insert into costOfCalciumCarbonateTank
values(?,?,?,?,?)");
stat.setFloat(1,historiccost);
stat.setFloat(2,capacityb);
stat.setFloat(3,capacitya);
stat.setFloat(4,exp);
stat.setFloat(5,presentcost);
stat.executeUpdate();
con.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
}
}
public static void main(String dd[])
{
new CalciumCarbonateTank();
}
}
```

The subclasses that compute the current cost of other equipment are similar to the one stated above. The only variation occurs in equipments like crysallier and some others that are more than one. The complete modules are in appendixes B and D on the CD that accompanies the thesis.

Subclass that computes various capital investments of the Plant.

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;
//Implements ActionListener for Event handling
public class RatioFactor implements ActionListener
{
Connection con1;
PreparedStatement stat;
Statement stat1;
```

```
ResultSet rs;
// declarations of controls
JFrame frame;
JLabel title,head,fac;
JLabel dq,pq,ic,ps,bs,ei,yi,sf,land,es,ce,cf,con;
JTextField text1,text2,text3,text4,text5,text6,text7,text8,text9,text10,text11,text12,text13;
JPanel panel, panel1,panel2,panel3,panel4;
JButton button;
GridBagLayout gbl;// for layout management
GridBagConstraints gbc;//for layout management
// declaring variables
float sum, f1,fcapn,  tcapn , wcapn,ff, ff1,sum1;
Font font1;float ri;
public RatioFactor()
{
//Initializing controls
font1 = new Font("Arial",Font.BOLD,17);
head = new JLabel("Item                    ");
fac = new JLabel("                              Factor");
dq = new JLabel("Delivered purchased equipment : ");
pq = new JLabel("Purchased equipmentinstallation: ");
ic = new JLabel("Instrumentation and controls: ");
ps = new JLabel("Piping(including services): ");
bs = new JLabel("Building(including services) : ");
ei = new JLabel("Electrical(installed): ");
yi = new JLabel("Yard improvements: ");
sf = new JLabel("Service facilities(installed): ");
land = new JLabel("Land : ");
es = new JLabel("Engeneering and sepervision: ");
ce = new JLabel("Construction expences: ");
cf = new JLabel("Contractor's fees: ");
con = new JLabel("Contingency : ");
text1 = new JTextField(10);
text2 = new JTextField(10);
text3 = new JTextField(10);
text4 = new JTextField(10);
text5 = new JTextField(10);
text6 = new JTextField(10);
text7 = new JTextField(10);
text8 = new JTextField(10);
text9 = new JTextField(10);
text10 = new JTextField(10);
text11 = new JTextField(10);
text12 = new JTextField(10);
text13 = new JTextField(10);
title = new JLabel("Computation of the ratio factors for estimating capital investments");
```

```
title.setFont(font1);// setting title's font
title.setForeground(Color.blue);// setting title's font color
button = new JButton("Compute");
frame = new JFrame("Computation of the ratio factors");
panel = new JPanel();
panel1 = new JPanel();
panel2 = new JPanel();
panel3 = new JPanel();
panel4 = new JPanel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
//arranging controls controls in the panel
panel.setLayout(gbl);
panel1.setLayout(gbl);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(title,gbc);
panel3.add(title);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(dq,gbc);
panel1.add(dq);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(head,gbc);
panel4.add(head);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(fac,gbc);
panel4.add(fac);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(text1,gbc);
panel1.add(text1);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(pq,gbc);
panel1.add(pq);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(text2,gbc);
panel1.add(text2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(ic,gbc);
panel1.add(ic);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(text3,gbc);
panel1.add(text3);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=12;
gbl.setConstraints(ps,gbc);
panel1.add(ps);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=12;
gbl.setConstraints(text4,gbc);
```

```
panel1.add(text4);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=17;
gbl.setConstraints(bs,gbc);
panel1.add(bs);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=17;
gbl.setConstraints(text5,gbc);
panel1.add(text5);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=21;
gbl.setConstraints(ei,gbc);
panel1.add(ei);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=21;
gbl.setConstraints(text6,gbc);
panel1.add(text6);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=25;
gbl.setConstraints(yi,gbc);
panel1.add(yi);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=25;
gbl.setConstraints(text7,gbc);
panel1.add(text7);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=28;
gbl.setConstraints(sf,gbc);
panel1.add(sf);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=28;
gbl.setConstraints(text8,gbc);
panel1.add(text8);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=32;
gbl.setConstraints(land,gbc);
panel1.add(land);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=32;
gbl.setConstraints(text9,gbc);
panel1.add(text9);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=36;
gbl.setConstraints(es,gbc);
panel1.add(es);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=36;
gbl.setConstraints(text10,gbc);
panel1.add(text10);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=39;
gbl.setConstraints(ce,gbc);
panel1.add(ce);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=39;
gbl.setConstraints(text11,gbc);
panel1.add(text11);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=42;
gbl.setConstraints(cf,gbc);
panel1.add(cf);
```

```
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=42;
gbl.setConstraints(text12,gbc);
panel1.add(text12);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=46;
gbl.setConstraints(con,gbc);
panel1.add(con);
gbc.anchor=gbc.WEST;
gbc.gridx=4;gbc.gridy=46;
gbl.setConstraints(text13,gbc);
panel1.add(text13);
gbc.anchor=gbc.EAST;gbc.gridx=7;gbc.gridy=8;
gbl.setConstraints(button,gbc);
panel.add(button);
//Activating the Enter button
button.addActionListener(this);
panel2.setLayout(gbl);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(panel3,gbc);
panel2.add(panel3);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(panel4,gbc);
panel2.add(panel4);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(panel1,gbc);
panel2.add(panel1);
gbc.anchor=gbc.CENTER;gbc.gridx=4;gbc.gridy=12;
gbl.setConstraints(panel,gbc);
panel2.add(panel);
frame.getContentPane().add(panel2);
frame.setSize(700,500);// setting frame size
frame.setVisible(true);// setting frame visible
}
// overiding action Performed method to activate the button
public void actionPerformed(ActionEvent ae)
{
Object obj = ae.getSource();// retriving the source of the event
if(obj == button)
{
if(Float.parseFloat(text1.getText()) < 1.0 && Float.parseFloat(text1.getText()) > 0 &&
Float.parseFloat(text2.getText()) < 1.0 && Float.parseFloat(text2.getText()) > 0 &&
Float.parseFloat(text3.getText()) < 1.0 && Float.parseFloat(text3.getText()) > 0 &&
Float.parseFloat(text4.getText()) < 1.0 && Float.parseFloat(text4.getText()) > 0 &&
Float.parseFloat(text5.getText()) < 1.0 && Float.parseFloat(text5.getText()) > 0 &&
Float.parseFloat(text6.getText()) < 1.0 && Float.parseFloat(text6.getText()) > 0 &&
Float.parseFloat(text6.getText()) < 1.0 && Float.parseFloat(text6.getText()) > 0 &&
Float.parseFloat(text7.getText()) < 1.0 && Float.parseFloat(text7.getText()) > 0 &&
```

```
Float.parseFloat(text8.getText()) < 1.0 && Float.parseFloat(text8.getText()) > 0 &&
Float.parseFloat(text9.getText()) < 1.0 && Float.parseFloat(text9.getText()) > 0 &&
Float.parseFloat(text10.getText()) < 1.0 && Float.parseFloat(text10.getText()) > 0 &&
Float.parseFloat(text11.getText()) < 1.0 && Float.parseFloat(text11.getText()) > 0  &&
Float.parseFloat(text12.getText()) < 1.0 && Float.parseFloat(text12.getText()) > 0 &&
Float.parseFloat(text13.getText()) < 1.0 && Float.parseFloat(text13.getText())  > 0 )
{
sum1 =
Float.parseFloat(text1.getText())+Float.parseFloat(text2.getText())+Float.parseFloat(text
3.getText())+Float.parseFloat(text4.getText())+Float.parseFloat(text5.getText())+Float.pa
rseFloat(text6.getText())+Float.parseFloat(text7.getText())+Float.parseFloat(text8.getTex
t())+Float.parseFloat(text9.getText());
sum =
Float.parseFloat(text10.getText())+Float.parseFloat(text11.getText())+Float.parseFloat(te
xt12.getText()) + Float.parseFloat(text13.getText());
f1 = 1 + sum;

try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select pect from TotalCapitalInvestment");
while(rs.next())
{
 ff = rs.getFloat(1);
}
con1.close();
fcapn = (ff + (sum1 * ff))*f1;

//if the fixed capital investment was taken as 80 percent of the total capital investment,
then the remaining 20 //percent becomes working capital investment.
tcapn = fcapn/0.8f;
wcapn = 0.2f * tcapn;
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
stat1.executeUpdate("delete from tcapn");
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
```

```
Connection con1;
PreparedStatement stat;
Statement stat1;
ResultSet rs;
Panel panel, panel1,panel2,panel3,panel4;
TextField input1, input2, input3,
input4,input5;
Button b1,b2;
float number[];
int degrees[];
JFrame chartframe = new JFrame("The graphical effect of the cost of some selected
Equipments on Total capital investment of the Plant");
long  sum;
float mu,pmu,hmu,omu;
int   counter,crsti,hei,pui,oti,tei;
float ff, ff1,crst,
ff2,ff3,he,pu,ot,hst,ca,cb,hhst,hca,hcb,hexp,hpresentcost,phst,pca,pcb,pexp,ppresentcost,o
hst,oca,ocb,oexp,opresentcost;
GridBagLayout gbl;// for layout management
GridBagConstraints gbc;//for layout management
Label ll1,ll2,ll3,ll4,ll5;
TextField tt1,tt2,tt3,tt4,tt5;
Font font;
Label title,title1;
Label
label1,label2,label3,label4,label5,label6,label7,label8,label9,label10,label11,label12,label
13,label14,label15,label16,label17,label18,label19,label20,label21,label22,label23,label2
4,label25,label26,label27,label28,label29,label,lab,label30,label31,label32;
TextField
text1,text2,text3,text4,text5,text6,text7,text8,text9,text10,text11,text12,text13,text14,text1
5,text16,text17,text18,text19,text20,text21,text22,text23,text24,text25,text26,text27,text2
8,text29,text30,text31,text32,text33,text34,text35;
TextField
textx1,textx2,textx3,textx4,textx5,textx6,textx7,textx8,textx9,textx10,textx11,textx12,text
x13,textx14,textx15,textx16,textx17,textx18,textx19,textx20,textx21,textx22,textx23,text
x24,textx25,textx26,textx27,textx28,textx29,textx30,textx31,textx32,textx33,textx34,text
x35;

JPanel panel0,
panel01,panel02,panel03,panel04,panel05,panel06,panel07,panel08,panel09,panel10,pan
el11;
JButton button;
JTextArea tdeliveredcost;
float presentcost,historiccost,capacityb,capacitya,exp;
Font font1,font2;
float ri;
```

```
float te,tee,teee,teq,teei;
float presentcostOfBernardinniFilter;
float pBETank;
float presentcostOfCalciumcabonatetank;
float
presentcostOfColdWaterTank,pCBleachingReactor,presentcostOfCrytalizers,presentcost
OfDeaerator;
float
pDecanter,cpDeodorizer,pDrier,presentcostOfFFARecuperator,pGuardFilter1,presentcost
OfGuardFilter2,presentcostOfHeatExchangers,pMixingTank,presentcostOfOleinTank,pP
hosphoricAcidTank,presentcostOfPreheatingTank,presentcostOfPump,presentcostOfRB
DStorageTank,presentcostOfSrewWorm,presentcostOfStearinTank,pSteel,pStorageTank,
presentcostOfVacuumSystem,presentcostOfWarmWaterTank,presentcostOfWeldersFilter
,deliveredcost;
 public chartimp()
{
//Initializing controls
pro=  new Label("Do you want to vary the Historic cost of equipments used in graph?");
  b2=new Button("Yes");
font1 = new Font("Arial",Font.BOLD,17);
Font font3 = new Font("Arial",Font.BOLD,10);
font2 = new Font("Arial",Font.BOLD,14);
pro.setFont(font3);
pro.setForeground(Color.blue);
label = new Label("                  Present cost        Historic cost/each (Naira)");
lab = new Label("Name of equipments ");
label1 = new Label("Two Bernardinni Filter: ");
label2 = new Label("Bleaching Earth Tank: ");
label3 = new Label("Calcium cabonate tank: ");
label4 = new Label("Cold Water Tank: ");
label5 = new Label("Continuous Bleaching Reactor:                ");
label6 = new Label("Twelve Crytalizers: ");
label7 = new Label("Deaerator: ");
label8 = new Label("Decanter: ");
label9 = new Label("Deodorizer: ");
label10 = new Label("Drier: ");
label11 = new Label("FFA Recuperator: ");
label12 = new Label("GuardFilter I ");
label13 = new Label("Two GuardFilter II: ");
label14 = new Label("Seven Heat Exchangers: ");
label15 = new Label("Mixing Tank: ");
label16 = new Label("Two Olein Tank: ");
label17 = new Label("Phosphoric Acid Tank: ");
label18 = new Label("PreheatingTank: ");
label19 = new Label("Twenty Two Pumps: ");
label20 = new Label("RBD Storage Tank: ");
```

```
label21 = new Label("Srew Worm: ");
label22 = new Label("Stearing Tank: ");
label23 = new Label("Steel Super Filter: ");
label24 = new Label("Storage Tank: ");
label25 = new Label("Vacuum System: ");
label26 = new Label("Warm Water Tank: ");
label27 = new Label("Welders Filter: ");
label28 = new Label("Delivered purchased equipment cost: ");
label30 = new Label("Fixed capital investment: ");
label31 = new Label("Working capital investment: ");
label32 = new Label("Total capital investment: ");
text1 = new TextField(14);text2 = new TextField(14);text3 = new TextField(14);
text4 = new TextField(14);text5 = new TextField(14);text6 = new TextField(14);
text7 = new TextField(14);text8 = new TextField(14);
text9 = new TextField(14);text10 = new TextField(14);text11 = new TextField(14);
text12 = new TextField(14);text13 = new TextField(14);text14 = new TextField(14);
text15 = new TextField(14);text16 = new TextField(14);text17 = new TextField(14);
text18 = new TextField(14);text19 = new TextField(14);text20 = new TextField(14);
text21 = new TextField(14);text22 = new TextField(14);text23 = new TextField(14);
text24 = new TextField(14);text25 = new TextField(14);text26 = new TextField(14);
text27 = new TextField(14);text28 = new TextField(14);text33 = new TextField(14);
text34 = new TextField(14);text35 = new TextField(14);textx1 = new TextField(14);
textx2 = new TextField(14);textx3 = new TextField(14);textx4 = new TextField(14);
textx5 = new TextField(14);textx6 = new TextField(14);textx7 = new TextField(14);
textx8 = new TextField(14);textx9 = new TextField(14);textx10 = new TextField(14);
textx11 = new TextField(14);textx12 = new TextField(14);textx13 = new TextField(14);
textx14 = new TextField(14);textx15 = new TextField(14);textx16 = new TextField(14);
textx17 = new TextField(14);textx18 = new TextField(14);textx19 = new TextField(14);
textx20 = new TextField(14);textx21 = new TextField(14);textx22 = new TextField(14);
textx23 = new TextField(14);textx24 = new TextField(14);textx25 = new TextField(14);
textx26 = new TextField(14);textx27 = new TextField(14);textx28 = new TextField(14);
textx33 = new TextField(14);textx34 = new TextField(14);textx35 = new TextField(14);
tdeliveredcost = new JTextArea(20,20);
title = new Label("Result for program CIS, representing the cost index and scaling
Method");
title1 = new Label("Result for program CIS of Plant Capacity 30,000 Meter cube Per
annum");
title.setFont(font1);// setting title's font
title1.setForeground(Color.blue);// setting title's font color
title1.setFont(font1);// setting title's font
label.setFont(font2);// setting title's font
lab.setFont(font2);// setting title's font
label28.setFont(font2);// setting title's font
title.setForeground(Color.blue);// setting title's font color
button = new JButton("Continue");
panel0 = new JPanel();
```

```
panel01 = new JPanel();
panel02 = new JPanel();
panel03 = new JPanel();
panel04 = new JPanel();
panel05 = new JPanel();
panel06 = new JPanel();
panel07 = new JPanel();
panel08 = new JPanel();
panel09 = new JPanel();
panel10 = new JPanel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
//arranging controls controls in the panel
panel0.setLayout(gbl);panel05.setLayout(gbl);panel06.setLayout(gbl);
panel07.setLayout(gbl);panel04.setLayout(gbl);panel03.setLayout(gbl);
panel02.setLayout(gbl);panel08.setLayout(gbl);panel09.setLayout(gbl);
panel10.setLayout(gbl);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfBernardinniFilter,presentcostOfBernardinniFilter from
costOfBernardinniFilter");
while(rs.next())
{
textx1.setText(String.valueOf(rs.getFloat(1)));
presentcostOfBernardinniFilter = rs.getFloat(2);// retriving
text1.setText(String.valueOf(presentcostOfBernardinniFilter));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);

}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
```

```
rs=stat1.executeQuery("select
historiccostOfBleachingEarthTank,presentcostOfBleachingEarthTank from
costOfBleachingEarthTank");

while(rs.next())
{
textx2.setText(String.valueOf(rs.getFloat(1)));
pBETank = rs.getFloat(2);// retrieving
text2.setText(String.valueOf(pBETank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfCalciumCarbonateTank,presentcostOfCalciumCarbonateTank from
costOfCalciumCarbonateTank");
while(rs.next())
{
textx3.setText(String.valueOf(rs.getFloat(1)));
presentcostOfCalciumcabonatetank = rs.getFloat(2);// retrieving
text3.setText(String.valueOf(presentcostOfCalciumcabonatetank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
```

```
rs=stat1.executeQuery("select
historiccostOfColdWaterTank,presentcostOfColdWaterTank from
costOfColdWaterTank");
while(rs.next())
{
textx4.setText(String.valueOf(rs.getFloat(1)));
presentcostOfColdWaterTank = rs.getFloat(2);// retrieving
text4.setText(String.valueOf(presentcostOfColdWaterTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfContinuousBleachingReactor,presentcostOfContinuousBleachingReactor
from costofContinuousBleachingReactor");
while(rs.next())
{
textx5.setText(String.valueOf(rs.getFloat(1)));
pCBleachingReactor = rs.getFloat(2);// retrieving
text5.setText(String.valueOf(pCBleachingReactor));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfCrytalizers,presentcostOfCrytalizers from
costOfCrytalizers");
```

```
while(rs.next())
{
textx6.setText(String.valueOf(rs.getFloat(1)));
presentcostOfCrytalizers = rs.getFloat(2);// retrieving
text6.setText(String.valueOf(presentcostOfCrytalizers));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfDeaerator,presentcostOfDeaerator from
costOfDeaerator");
while(rs.next())
{
textx7.setText(String.valueOf(rs.getFloat(1)));
presentcostOfDeaerator = rs.getFloat(2);// retrieving
text7.setText(String.valueOf(presentcostOfDeaerator));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfDecanter,presentcostOfDecanter from
costofDecanter");
while(rs.next())
{
textx8.setText(String.valueOf(rs.getFloat(1)));
pDecanter = rs.getFloat(2);// retrieving
```

```
text8.setText(String.valueOf(pDecanter));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfDeodorizer,presentcostOfDeodorizer from
costOfDeodorizer");
while(rs.next())
{
textx9.setText(String.valueOf(rs.getFloat(1)));
cpDeodorizer = rs.getFloat(2);// retrieving
text9.setText(String.valueOf(cpDeodorizer));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfDrier,presentcostOfDrier from costofDrier");
while(rs.next())
{
textx10.setText(String.valueOf(rs.getFloat(1)));
pDrier = rs.getFloat(2);// retrieving
text10.setText(String.valueOf(pDrier));
}
con1.close();
}
catch(Exception e)
```

```
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);

}

try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfFFARecuperator,presentcostOfFFARecuperator from
costOfFFARecuperator");
while(rs.next())
{
textx11.setText(String.valueOf(rs.getFloat(1)));
presentcostOfFFARecuperator = rs.getFloat(2);// retrieving
text11.setText(String.valueOf(presentcostOfFFARecuperator));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfGuardFilter1,presentcostOfGuardFilter1
from costofGuardFilter1");
while(rs.next())
{
textx12.setText(String.valueOf(rs.getFloat(1)));
pGuardFilter1 = rs.getFloat(2);// retrieving
text12.setText(String.valueOf(pGuardFilter1));
}
con1.close();
}
catch(Exception e)
{
```

```
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);

}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfGuardFilter2,presentcostOfGuardFilter2
from costOfGuardFilter2");
while(rs.next())
{
textx13.setText(String.valueOf(rs.getFloat(1)));
presentcostOfGuardFilter2 = rs.getFloat(2);// retrieving
text13.setText(String.valueOf(presentcostOfGuardFilter2));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfHeatExchangers,presentcostOfHeatExchangers from
costOfHeatExchangers");
while(rs.next())
{
textx14.setText(String.valueOf(rs.getFloat(1)));
presentcostOfHeatExchangers = rs.getFloat(2);// retrieving
text14.setText(String.valueOf(presentcostOfHeatExchangers));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
```

```
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfMixingTank,presentcostOfMixingTank from
costofMixingTank");
while(rs.next())
{
textx15.setText(String.valueOf(rs.getFloat(1)));
pMixingTank = rs.getFloat(2);// retrieving
text15.setText(String.valueOf(pMixingTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfOleinTank,presentcostOfOleinTank from
costOfOleinTank");
while(rs.next())
{
textx16.setText(String.valueOf(rs.getFloat(1)));
presentcostOfOleinTank = rs.getFloat(2);// retrieving
text16.setText(String.valueOf(presentcostOfOleinTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfPhosphoricAcidTank,presentcostOfPhosphoricAcidTank from
costofPhosphoricAcidTank");
while(rs.next())
{
textx17.setText(String.valueOf(rs.getFloat(1)));
pPhosphoricAcidTank = rs.getFloat(2);// retrieving
text17.setText(String.valueOf(pPhosphoricAcidTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfPreheatingTank,presentcostOfPreheatingTank from
costOfPreheatingTank");
while(rs.next())
{
textx18.setText(String.valueOf(rs.getFloat(1)));
presentcostOfPreheatingTank = rs.getFloat(2);// retrieving
text18.setText(String.valueOf(presentcostOfPreheatingTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
```

```
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfPump,presentcostOfPump from
costOfPump");
while(rs.next())
{
textx19.setText(String.valueOf(rs.getFloat(1)));
presentcostOfPump = rs.getFloat(2);// retrieving
text19.setText(String.valueOf(presentcostOfPump));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfRBDStorageTank,presentcostOfRBDStorageTank from
costOfRBDStorageTank");
while(rs.next())
{
textx20.setText(String.valueOf(rs.getFloat(1)));
presentcostOfRBDStorageTank = rs.getFloat(2);// retrieving
text20.setText(String.valueOf(presentcostOfRBDStorageTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
```

```
rs=stat1.executeQuery("select historiccostOfSrewWorm,presentcostOfSrewWorm from
costOfSrewWorm");
while(rs.next())
{
textx21.setText(String.valueOf(rs.getFloat(1)));
presentcostOfSrewWorm = rs.getFloat(2);// retrieving
text21.setText(String.valueOf(presentcostOfSrewWorm));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfStearinTank,presentcostOfStearinTank from
costOfStearinTank");
while(rs.next())
{
textx22.setText(String.valueOf(rs.getFloat(1)));
presentcostOfStearinTank = rs.getFloat(2);// retrieving
text22.setText(String.valueOf(presentcostOfStearinTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfSteelSuperFilter,presentcostOfSteelSuperFilter from
costofSteelSuperFilter");
while(rs.next())
```

```
{
textx23.setText(String.valueOf(rs.getFloat(1)));
pSteel = rs.getFloat(2);// retrieving
text23.setText(String.valueOf(pSteel));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select presentcostOfStorageTank,presentcostOfStorageTank from
costofStorageTank");
while(rs.next())
{
textx24.setText(String.valueOf(rs.getFloat(1)));
pStorageTank = rs.getFloat(2);// retrieving
text24.setText(String.valueOf(pStorageTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfVacuumSystem,presentcostOfVacuumSystem from
costOfVacuumSystem");
while(rs.next())
{
textx25.setText(String.valueOf(rs.getFloat(1)));
presentcostOfVacuumSystem = rs.getFloat(2);// retrieving
```

```
text25.setText(String.valueOf(presentcostOfVacuumSystem));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfWarmWaterTank,presentcostOfWarmWaterTank from
costOfWarmWaterTank");
while(rs.next())
{
textx26.setText(String.valueOf(rs.getFloat(1)));
presentcostOfWarmWaterTank = rs.getFloat(2);// retrieving
text26.setText(String.valueOf(presentcostOfWarmWaterTank));
}
con1.close();
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");// loading the driver
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");// establishing
connection to the //database
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfWeldersFilter,presentcostOfWeldersFilter
from costOfWeldersFilter");
while(rs.next())
{
textx27.setText(String.valueOf(rs.getFloat(1)));
presentcostOfWeldersFilter = rs.getFloat(2);// retrieving
text27.setText(String.valueOf(presentcostOfWeldersFilter));
}
con1.close();
```

```
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select fcapn,wcapn,f1 from tcapn");
while(rs.next())
{
te = rs.getFloat(1);
text33.setText(String.valueOf(te));
tee = rs.getFloat(2);
text34.setText(String.valueOf(tee));
teee = rs.getFloat(3);
teq =  te/0.8f;
text35.setText(String.valueOf(teq));
teei = teee - 1;
}
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
deliveredcost =
presentcostOfBernardinniFilter+pBETank+presentcostOfCalciumcabonatetank+presentc
ostOfColdWaterTank+pCBleachingReactor+presentcostOfCrytalizers+presentcostOfDea
erator+pDecanter+cpDeodorizer+pDrier+presentcostOfFFARecuperator+pGuardFilter1+
presentcostOfGuardFilter2+presentcostOfHeatExchangers+pMixingTank+presentcostOf
OleinTank+pPhosphoricAcidTank+presentcostOfPreheatingTank+presentcostOfPump+p
resentcostOfRBDStorageTank+presentcostOfSrewWorm+presentcostOfStearinTank+pSt
eel+pStorageTank+presentcostOfVacuumSystem+presentcostOfWarmWaterTank+prese
ntcostOfWeldersFilter;
text28.setText(String.valueOf(deliveredcost));
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
stat1.executeUpdate("delete from TotalCapitalInvestment");
con1.close();
```

```
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat = con1.prepareStatement("insert into TotalCapitalInvestment
values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");
stat.setFloat(1,presentcostOfBernardinniFilter);
stat.setFloat(2,pBETank);
stat.setFloat(3,presentcostOfCalciumcabonatetank);
stat.setFloat(4,presentcostOfColdWaterTank);
stat.setFloat(5,pCBleachingReactor);
stat.setFloat(6,presentcostOfCrytalizers);
stat.setFloat(7,presentcostOfDeaerator);
stat.setFloat(8,pDecanter);
stat.setFloat(9,cpDeodorizer);
stat.setFloat(10,pDrier);
stat.setFloat(11,presentcostOfFFARecuperator);
stat.setFloat(12,pGuardFilter1);
stat.setFloat(13,presentcostOfGuardFilter2);
stat.setFloat(14,presentcostOfHeatExchangers);
stat.setFloat(15,pMixingTank);
stat.setFloat(16,presentcostOfOleinTank);
stat.setFloat(17,pPhosphoricAcidTank);
stat.setFloat(18,presentcostOfPreheatingTank);
stat.setFloat(19,presentcostOfPump);
stat.setFloat(20,presentcostOfRBDStorageTank);
stat.setFloat(21,presentcostOfSrewWorm);
stat.setFloat(22,presentcostOfStearinTank);
stat.setFloat(23,pSteel);
stat.setFloat(24,pStorageTank);
stat.setFloat(25,presentcostOfVacuumSystem);
stat.setFloat(26,presentcostOfWarmWaterTank);
stat.setFloat(27,presentcostOfWeldersFilter);
stat.setFloat(28,deliveredcost);
stat.executeUpdate();
con1.close();
}
catch(Exception e)
{
```

```
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(title,gbc);
panel03.add(title);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(title1,gbc);
panel04.add(title1);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(lab,gbc);
panel06.add(lab);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(label,gbc);
panel06.add(label);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(label1,gbc);
panel05.add(label1);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(text1,gbc);
panel05.add(text1);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=1;
gbl.setConstraints(textx1,gbc);
panel05.add(textx1);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;gbl.setConstraints(label2,gbc);
panel05.add(label2);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(text2,gbc);
panel05.add(text2);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=4;
gbl.setConstraints(textx2,gbc);
panel05.add(textx2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(label3,gbc);
panel05.add(label3);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints(text3,gbc);
panel05.add(text3);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=8;
gbl.setConstraints(textx3,gbc);
panel05.add(textx3);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=12;
gbl.setConstraints(label4,gbc);
panel05.add(label4);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=12;
gbl.setConstraints(text4,gbc);
```

```
panel05.add(text4);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=12;
gbl.setConstraints(textx4,gbc);
panel05.add(textx4);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=17;
gbl.setConstraints(label5,gbc);
panel05.add(label5);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=17;
gbl.setConstraints(text5,gbc);
panel05.add(text5);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=17;
gbl.setConstraints(textx5,gbc);
panel05.add(textx5);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=21;
gbl.setConstraints(label6,gbc);
panel05.add(label6);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=21;
gbl.setConstraints(text6,gbc);
panel05.add(text6);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=21;
gbl.setConstraints(textx6,gbc);
panel05.add(textx6);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=25;
gbl.setConstraints(label7,gbc);
panel05.add(label7);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=25;
gbl.setConstraints(text7,gbc);
panel05.add(text7);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=25;
gbl.setConstraints(textx7,gbc);
panel05.add(textx7);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=28;
gbl.setConstraints(label8,gbc);
panel05.add(label8);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=28;
gbl.setConstraints(text8,gbc);
panel05.add(text8);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=28;
gbl.setConstraints(textx8,gbc);
panel05.add(textx8);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=32;
gbl.setConstraints(label9,gbc);
panel05.add(label9);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=32;
gbl.setConstraints(text9,gbc);
panel05.add(text9);
```

```
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=32;
gbl.setConstraints(textx9,gbc);
panel05.add(textx9);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=36;
gbl.setConstraints(label10,gbc);
panel05.add(label10);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=36;
gbl.setConstraints(text10,gbc);
panel05.add(text10);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=36;
gbl.setConstraints(textx10,gbc);
panel05.add(textx10);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=39;
gbl.setConstraints(label11,gbc);
panel05.add(label11);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=39;
gbl.setConstraints(text11,gbc);
panel05.add(text11);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=39;
gbl.setConstraints(textx11,gbc);
panel05.add(textx11);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=42;
gbl.setConstraints(label12,gbc);
panel05.add(label12);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=42;
gbl.setConstraints(text12,gbc);
panel05.add(text12);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=42;
gbl.setConstraints(textx12,gbc);
panel05.add(textx12);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=46;
gbl.setConstraints(label13,gbc);
panel05.add(label13);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=46;
gbl.setConstraints(text13,gbc);
panel05.add(text13);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=46;
gbl.setConstraints(textx13,gbc);
panel05.add(textx13);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=50;
gbl.setConstraints(label14,gbc);
panel05.add(label14);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=50;
gbl.setConstraints(text14,gbc);
panel05.add(text14);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=50;
```

```
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=32;
gbl.setConstraints(textx9,gbc);
panel05.add(textx9);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=36;
gbl.setConstraints(label10,gbc);
panel05.add(label10);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=36;
gbl.setConstraints(text10,gbc);
panel05.add(text10);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=36;
gbl.setConstraints(textx10,gbc);
panel05.add(textx10);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=39;
gbl.setConstraints(label11,gbc);
panel05.add(label11);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=39;
gbl.setConstraints(text11,gbc);
panel05.add(text11);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=39;
gbl.setConstraints(textx11,gbc);
panel05.add(textx11);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=42;
gbl.setConstraints(label12,gbc);
panel05.add(label12);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=42;
gbl.setConstraints(text12,gbc);
panel05.add(text12);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=42;
gbl.setConstraints(textx12,gbc);
panel05.add(textx12);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=46;
gbl.setConstraints(label13,gbc);
panel05.add(label13);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=46;
gbl.setConstraints(text13,gbc);
panel05.add(text13);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=46;
gbl.setConstraints(textx13,gbc);
panel05.add(textx13);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=50;
gbl.setConstraints(label14,gbc);
panel05.add(label14);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=50;
gbl.setConstraints(text14,gbc);
panel05.add(text14);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=50;
```

```
gbl.setConstraints(textx14,gbc);
panel05.add(textx14);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=53;
gbl.setConstraints(label15,gbc);
panel05.add(label15);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=53;
gbl.setConstraints(text15,gbc);
panel05.add(text15);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=53;
gbl.setConstraints(textx15,gbc);
panel05.add(textx15);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=56;
gbl.setConstraints(label16,gbc);
panel05.add(label16);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=56;
gbl.setConstraints(text16,gbc);
panel05.add(text16);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=56;
gbl.setConstraints(textx16,gbc);
panel05.add(textx16);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=60;
gbl.setConstraints(label17,gbc);
panel05.add(label17);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=60;
gbl.setConstraints(text17,gbc);
panel05.add(text17);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=60;
gbl.setConstraints(textx17,gbc);
panel05.add(textx17);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=63;
gbl.setConstraints(label18,gbc);
panel05.add(label18);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=63;
gbl.setConstraints(text18,gbc);
panel05.add(text18);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=63;
gbl.setConstraints(textx18,gbc);
panel05.add(textx18);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=66;
gbl.setConstraints(label19,gbc);
panel05.add(label19);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=66;
gbl.setConstraints(text19,gbc);
panel05.add(text19);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=66;
gbl.setConstraints(textx19,gbc);
```

```
panel05.add(textx19);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=69;
gbl.setConstraints(label20,gbc);
panel05.add(label20);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=69;
gbl.setConstraints(text20,gbc);
panel05.add(text20);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=69;
gbl.setConstraints(textx20,gbc);
panel05.add(textx20);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=72;
gbl.setConstraints(label21,gbc);
panel05.add(label21);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=72;
gbl.setConstraints(text21,gbc);
panel05.add(text21);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=72;
gbl.setConstraints(textx21,gbc);
panel05.add(textx21);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=75;
gbl.setConstraints(label22,gbc);
panel05.add(label22);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=75;
gbl.setConstraints(text22,gbc);
panel05.add(text22);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=75;
gbl.setConstraints(textx22,gbc);
panel05.add(textx22);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=78;
gbl.setConstraints(label23,gbc);
panel05.add(label23);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=78;
gbl.setConstraints(text23,gbc);
panel05.add(text23);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=78;
gbl.setConstraints(textx23,gbc);
panel05.add(textx23);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=81;
gbl.setConstraints(label24,gbc);
panel05.add(label24);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=81;
gbl.setConstraints(text24,gbc);
panel05.add(text24);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=81;
gbl.setConstraints(textx24,gbc);
panel05.add(textx24);
```

```
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=84;
gbl.setConstraints(label25,gbc);
panel05.add(label25);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=84;
gbl.setConstraints(text25,gbc);
panel05.add(text25);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=84;
gbl.setConstraints(textx25,gbc);
panel05.add(textx25);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=87;
gbl.setConstraints(label26,gbc);
panel05.add(label26);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=87;
gbl.setConstraints(text26,gbc);
panel05.add(text26);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=87;
gbl.setConstraints(textx26,gbc);
panel05.add(textx26);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=90;
gbl.setConstraints(label27,gbc);
panel05.add(label27);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=90;
gbl.setConstraints(text27,gbc);
panel05.add(text27);
gbc.anchor=gbc.WEST;gbc.gridx=8;gbc.gridy=90;
gbl.setConstraints(textx27,gbc);
panel05.add(textx27);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=93;
gbl.setConstraints(label28,gbc);
panel08.add(label28);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=93;
gbl.setConstraints(text28,gbc);
panel08.add(text28);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=96;
gbl.setConstraints(label30,gbc);
panel08.add(label30);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=96;
gbl.setConstraints(text33,gbc);
panel08.add(text33);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=99;
gbl.setConstraints(label31,gbc);
panel08.add(label31);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=99;
gbl.setConstraints(text34,gbc);
panel08.add(text34);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=102;
```

```
gbl.setConstraints(label32,gbc);
panel08.add(label32);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=102;
gbl.setConstraints(text35,gbc);
panel08.add(text35);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(pro,gbc);
panel10.add(pro);
gbc.anchor=gbc.WEST;gbc.gridx=3;gbc.gridy=4;
gbl.setConstraints(b2,gbc);
panel10.add(b2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(panel04,gbc);
panel07.add(panel04);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(panel06,gbc);
panel07.add(panel06);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=7;
gbl.setConstraints(panel05,gbc);
panel07.add(panel05);
font = new Font("Lucida Console",Font.BOLD,10);
ll1 = new Label("Crystalizer");
ll2 = new Label("Heat Exchanger");
ll3 = new Label("Pump");
ll4 = new Label("Olein Tank");
ll5 = new Label("Total Capital Investment");
tt1 = new TextField(2);
tt2 = new TextField(2);
tt3 = new TextField(2);
tt4 = new TextField(2);
tt5 = new TextField(2);
tt1.setBackground(Color.green);
tt2.setBackground(Color.yellow);
tt3.setBackground(Color.black);
tt4.setBackground(Color.red);
tt5.setBackground(Color.green);
panel = new Panel();
panel1 = new Panel();
panel2 = new Panel();
panel3 = new Panel();
panel4 = new Panel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
panel4.setLayout(gbl);
panel3.setLayout(gbl);
panel2.setLayout(gbl);
```

```
panel1.setLayout(gbl);
setLayout(gbl);
 sum=0;
 counter=0;
 number=new float[4];
  degrees=new int[4];
inputlabel2= new Label("        The effect of variation of historic cost of Equipments");
inputlabel2.setFont(font);
inputlabel2.setForeground(Color.blue);
 inputlabel= new Label("Enter new historic cost of Crystalizer:");
 inputlabel1= new Label("Enter new historic cost of Pump:");
 inputlabel4= new Label("Enter new historic cost of Heat Exchanger:");
 inputlabel5= new Label("Enter new historic cost of Olien Tank:");
 input1=new TextField(12);
 input2=new TextField(12);
 input3=new TextField(12);
 input4=new TextField(12);
 b1=new Button("Alter Chart structure");
 gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(ll1,gbc);
panel4.add(ll1);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(tt1,gbc);
panel4.add(tt1);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(ll2,gbc);
panel4.add(ll2);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(tt2,gbc);
panel4.add( tt2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(ll3,gbc);
panel4.add(ll3);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=8;
gbl.setConstraints( tt3,gbc);
panel4.add( tt3);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=13;
gbl.setConstraints(ll4,gbc);
panel4.add(ll4);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=13;
gbl.setConstraints(tt4,gbc);
panel4.add(tt4);
panel09.setLayout(gbl);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=7;
gbl.setConstraints(panel1,gbc);
panel09.add(panel1);
```

```
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(panel2,gbc);
panel09.add(panel2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=14;
gbl.setConstraints(panel4,gbc);
panel09.add(panel4);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=18;
gbl.setConstraints(panel08,gbc);
panel09.add(panel08);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=22;
gbl.setConstraints(panel10,gbc);
panel09.add(panel10);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=18;
gbl.setConstraints(panel09,gbc);
add(panel09);
gbc.anchor=gbc.WEST;gbc.gridx=16;gbc.gridy=18;
gbl.setConstraints(panel07,gbc);
add(panel07);
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfCrytalizers from costOfCrytalizers");
while(rs.next())
{
 ff = rs.getFloat(1);
}
con1.close();
input1.setText(String.valueOf(ff));// setting the value in the text field for chart
manipulation
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfPump from costOfPump");
while(rs.next())
{
 ff1 = rs.getFloat(1);
```

```
sum= crsti+hei+pui+oti;
 degrees[0]=crsti*360/(int)sum;
 degrees[1]=hei*360/(int)sum;
 degrees[2]=pui*360/(int)sum;
  degrees[3]=360-degrees[0]
        -degrees[1]-degrees[2];

repaint();
 b1.addActionListener(this);
 b2.addActionListener(this);
}
 public void actionPerformed(ActionEvent e)
 {
Object object = e.getSource();
if(object == b2)
{
chartframe.setVisible(false);
chart p = new chart();
p.main1();
}

 }
 public void paint(Graphics g)
 {
 g.setColor(Color.green);
 g.fillArc
  (40,60,200,200,0,degrees[0]);
 g.setColor(Color.yellow);
 g.fillArc(40,60,200,200,
   degrees[0], degrees[1]);
 g.setColor(Color.black);
 g.fillArc(40,60,200,200,
  degrees[0]+degrees[1],
  degrees[2]);
g.setColor(Color.red);
 g.fillArc(40,60,200,200,
  degrees[0]+degrees[1]+degrees[2],degrees[3]);
 }
public  void main1()
{

chartframe.getContentPane().add(this);
chartframe.setSize(1000,1000);
chartframe.setVisible(true);
}
public static void main(String j[])
```

```
{
chartimp p = new chartimp();
p.main1();
}
}
class chart extends Applet implements ActionListener
{
Label inputlabel,inputlabel1, inputlabel2,inputlabel3, inputlabel4,inputlabel5,pro;
Connection con1;
PreparedStatement stat;
Statement stat1;
ResultSet rs;
Panel panel, panel1,panel2,panel3,panel4;
TextField input1, input2, input3,input4,input5;
Button b1,b2;
float number[];
int degrees[];
JFrame chartframe;
long  sum;
float mu,pmu,hmu,omu;
int   counter,crsti,hei,pui,oti,tei;
float ri,ff, ff1,crst,
ff2,ff3,he,pu,ot,te,hst,ca,cb,exp,presentcost,hhst,hca,hcb,hexp,hpresentcost,phst,pca,pcb,p
exp,ppresentcost,ohst,oca,ocb,oexp,opresentcost,tee,teei;
GridBagLayout gbl;// for layout management
GridBagConstraints gbc;//for layout management
Label ll1,ll2,ll3,ll4,ll5;
TextField tt1,tt2,tt3,tt4,tt5;
Font font;
 public chart()
{
font = new Font("Lucida Console",Font.BOLD,10);
ll1 = new Label("Crystalizer");
ll2 = new Label("Heat Exchanger");
ll3 = new Label("Pump");
ll4 = new Label("Olein Tank");
ll5 = new Label("Total Capital Investment");
tt1 = new TextField(2);
tt2 = new TextField(2);
tt3 = new TextField(2);
tt4 = new TextField(2);
tt5 = new TextField(2);
tt1.setBackground(Color.green);
tt2.setBackground(Color.yellow);
tt3.setBackground(Color.black);
tt4.setBackground(Color.red);
```

```
tt5.setBackground(Color.green);
panel = new Panel();
panel1 = new Panel();
panel2 = new Panel();
panel3 = new Panel();
panel4 = new Panel();
gbl = new GridBagLayout();
gbc = new GridBagConstraints();
panel4.setLayout(gbl);
panel3.setLayout(gbl);
panel2.setLayout(gbl);
panel1.setLayout(gbl);
setLayout(gbl);
 sum=0;
 counter=0;
  number=new float[4];
  degrees=new int[4];
inputlabel2= new Label("        The effect of variation of historic cost of Equipments");
inputlabel2.setFont(font);
inputlabel2.setForeground(Color.blue);
 inputlabel= new Label("Enter new historic cost of Crystalizer:");
 pro= new Label("  ");
 inputlabel1= new Label("Enter new historic cost of Pump:");
 inputlabel4= new Label("Enter new historic cost of Heat Exchanger:");
 inputlabel5= new Label("Enter new historic cost of Olien Tank:");
 input1=new TextField(12);
 input2=new TextField(12);
 input3=new TextField(12);
 input4=new TextField(12);
 b1=new Button("Alter Chart structure");
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=1;
gbl.setConstraints(ll1,gbc);
panel4.add(ll1);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=1;
gbl.setConstraints(tt1,gbc);
panel4.add(tt1);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=4;
gbl.setConstraints(ll2,gbc);
panel4.add(ll2);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=4;
gbl.setConstraints(tt2,gbc);
panel4.add( tt2);
gbc.anchor=gbc.WEST;gbc.gridx=1;gbc.gridy=8;
gbl.setConstraints(ll3,gbc);
panel4.add(ll3);
gbc.anchor=gbc.WEST;gbc.gridx=4;gbc.gridy=8;
```

```
input4.setText(String.valueOf(ff3));// setting the value in the text field for chart
manipulation
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfCrytalizers, capacitya,
capacityb,exponential ,presentcostOfCrytalizers from costOfCrytalizers");
while(rs.next())
{
hst = rs.getFloat(1);
ca = rs.getFloat(2);
cb = rs.getFloat(3);
exp = rs.getFloat(4);
crst = rs.getFloat(5);
float div = crst / 100000;
 crsti = Math.round(div);
}
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select
historiccostOfHeatExchangers,capacitya,capacityb,exponential,
presentcostOfHeatExchangers from costOfHeatExchangers");
while(rs.next())
{
hhst = rs.getFloat(1);
hca = rs.getFloat(2);
hcb = rs.getFloat(3);
hexp = rs.getFloat(4);
 he = rs.getFloat(5);
```

```
float div= he/100000;
 hei = Math.round(div);
}
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfPump, capacitya,capacityb ,
exponential,presentcostOfPump from costofpump");
while(rs.next())
{
phst = rs.getFloat(1);pca = rs.getFloat(2);pcb = rs.getFloat(3);pexp = rs.getFloat(4);
 pu = rs.getFloat(5);
float div= pu/100000;
 pui = Math.round(div);
}
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfOleinTank,capacitya, capacityb, exponential,
presentcostOfOleinTank from costOfOleinTank");
while(rs.next())
{
ohst = rs.getFloat(1);
oca = rs.getFloat(2);
ocb = rs.getFloat(3);
oexp = rs.getFloat(4);
ot = rs.getFloat(5);
float div = ot/100000;
oti = Math.round(div);
}
```

```
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfPump from costOfPump");
while(rs.next())
{
 ff1 = rs.getFloat(1);
}
con1.close();
input2.setText(String.valueOf(ff1));// setting the value in the text field for chart
manipulation
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfHeatExchangers from
costOfHeatExchangers");
while(rs.next())
{
 ff2 = rs.getFloat(1);
}
con1.close();
input3.setText(String.valueOf(ff2));// setting the value in the text field for chart
manipulation
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
con1 = DriverManager.getConnection("jdbc:odbc:MyDataSource","sa","");
stat1 = con1.createStatement();
rs=stat1.executeQuery("select historiccostOfOleinTank from costOfOleinTank");
while(rs.next())
{
 ff3 = rs.getFloat(1);
}
con1.close();
```

```
}
catch(Exception e)
{
JOptionPane.showMessageDialog(null,"Error
:"+e.getMessage(),"Error",JOptionPane.ERROR_MESSAGE);
}
// Drawing of pie chart
 sum= crsti+hei+pui+oti;
  degrees[0]=crsti*360/(int)sum;
  degrees[1]=hei*360/(int)sum;
  degrees[2]=pui*360/(int)sum;
   degrees[3]=360-degrees[0]
         -degrees[1]-degrees[2];

repaint();
 b1.addActionListener(this);
 }
 public void actionPerformed(ActionEvent e)
 {
Object object = e.getSource();
if(object == b1)
{
hst = Float.parseFloat(input1.getText());
phst = Float.parseFloat(input2.getText());
hhst = Float.parseFloat(input3.getText());
ohst = Float.parseFloat(input4.getText());
float val = ca/cb;
presentcost = (float)(hst * ri * (Math.pow((val),exp) ));
float hval = hca/hcb;
hpresentcost = (float)(hhst * ri * (Math.pow((hval),hexp) ));
float pval = pca/pcb;
ppresentcost = (float)(phst * ri * (Math.pow((pval),pexp) ));
float oval = oca/ocb;
opresentcost = (float)(ohst * ri * (Math.pow((oval),oexp) ));
 mu = presentcost * 12;
float pre1=mu/100000;
 hmu = hpresentcost * 7;
float hpre1=hmu/100000;
 pmu = ppresentcost * 22;
float ppre1=pmu/100000;
 omu = opresentcost * 2;
float opre1=omu/100000;
int pre = Math.round(pre1);
int hpre = Math.round(hpre1);
int ppre = Math.round(ppre1);
int opre = Math.round(opre1);
```

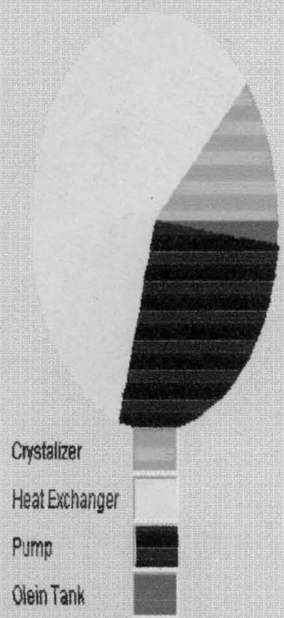
```
int sum = pre+hpre+ppre+opre;
 degrees[0]=pre*360/(int)sum;
  degrees[1]=hpre*360/(int)sum;
  degrees[2]=ppre*360/(int)sum;
  degrees[3]=360-degrees[0]
   -degrees[1]-degrees[2];
repaint();
}
 }
  public void paint(Graphics g)
  {
  g.setColor(Color.green);
  g.fillArc
    (40,60,200,200,0,degrees[0]);
  g.setColor(Color.yellow);
  g.fillArc(40,60,200,200,
    degrees[0], degrees[1]);
  g.setColor(Color.black);
  g.fillArc(40,60,200,200,
   degrees[0]+degrees[1],
   degrees[2]);
g.setColor(Color.red);
  g.fillArc(40,60,200,200,
   degrees[0]+degrees[1]+degrees[2],degrees[3]);
  }
public  void main1()
{
 chartframe = new JFrame("The graphical effect of variation of historic cost of
Equipments");
chartframe.getContentPane().add(this);
chartframe.setSize(500,800);
chartframe.setVisible(true);
}
public static void main(String j[])
{
chart p = new chart();
p.main1();
}
}
```

4.1.2 Result for total cost of palm oil refinery and fractionation plant

The results obtained by running CIS and 610FR are as shown in the figure 4.1 and 4.2 respectively.

The graphical effect of the cost of some selected Equipments on Total capital investment of the Plant

Result for program CIS of Plant Capacity 30,000 Meter cube Per annum



Name of equipments	Present cost	Historic cost/each (Naira)
Two Bernardinni Filter:	2603733.2	693806.0
Bleaching Earth Tank:	228614.25	104604.0
Calcium cabonate tank:	144588.2	79773.0
Cold Water Tank:	174117.38	96065.0
Continuous Bleaching Reactor:	1160787.2	640436.0
Twelve Crytalizers:	2883982.0	144098.0
Deaerator:	207800.78	114649.0
Decanter:	136036.67	66712.0
Deodorizer:	652976.8	320218.0
Drier:	185076.77	103537.0
FFA Recuperator:	177987.06	98200.0
GuardFilter I	227562.48	111596.0
Two GuardFilter II:	455124.97	111596.0
Seven Heat Exchangers:	1.4992362E7	1050316.0
Mixing Tank:	142775.7	78773.0
Two Olein Tank:	373971.72	104605.0
Phosphoric Acid Tank:	175536.66	98200.0
PreheatingTank:	189594.27	104604.0
Twenty Two Pumps:	7142661.0	200136.0
RBD Storage Tank:	189596.1	104605.0
Srew Worm:	822612.8	320925.0
Stearing Tank:	186985.86	104605.0
Steel Super Filter:	239424.3	117413.0
Storage Tank:	190368.2	190368.2
Vacuum System:	803161.2	393868.0
Warm Water Tank:	174117.38	96065.0
Welders Filter:	1958930.6	960654.0

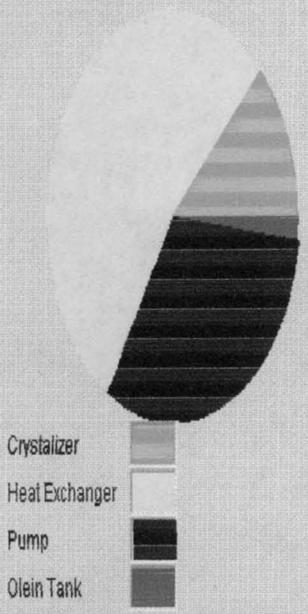
Delivered purchased equipment cost:	3.6820484E7
Fixed capital investment:	2.88762176E8
Working capital investment:	7.2190544E7
Total capital investment:	3.60952704E8

Do you want to vary the Historic cost of equipments used in graph?

Yes

Figure 4.6 Result for Program CIS

The graphical effect of the cost of some selected Equipments on Total capital investment of the Plant



Delivered purchased equipment cost:	3.8305432E7
Fixed capital investment:	3.23190592E8
Working capital investment:	8.0797648E7
Total capital investment:	4.03988224E8

Do you want to vary the Historic cost of equipments used in graph?

Yes

Result for program 610FR of Plant Capacity 30,000 Meter cube Per annum

Name of equipments	Present cost	Historic cost/each (Naira)
Two Bernardinni Filter:	2714298.8	693805.0
Bleaching Earth Tank:	204615.48	104604.0
Calcium cabonate tank:	154087.56	78773.0
Cold Water Tank:	187912.38	96065.0
Continuous Bleaching Reactor:	1252754.4	640436.0
Twelve Crytalizers:	3382434.8	144098.0
Deaerator:	224264.47	114649.0
Decanter:	239876.31	232211.0
Deodorizer:	626377.2	320218.0
Drier:	202528.34	103537.0
FFA Recuperator:	192088.66	98200.0
GuardFilter I	218292.53	111596.0
Two GuardFilter II:	436585.06	111596.0
Seven Heat Exchangers:	1.4381634E7	1050316.0
Mixing Tank:	154087.56	78773.0
Two Olein Tank:	409234.9	104605.0
Phosphoric Acid Tank:	192088.66	98200.0
PreheatingTank:	204615.48	104604.0
Twenty Two Pumps:	8612676.0	200136.0
RBD Storage Tank:	204617.45	104605.0
Srew Worm:	645169.44	329825.0
Stearing Tank:	204617.45	104605.0
Steel Super Filter:	229671.12	117413.0
Storage Tank:	205450.73	205450.73
Vacuum System:	770443.7	393868.0
Warm Water Tank:	187912.38	96065.0
Welders Filter:	1867098.8	960654.0

Figure 4.7 Results for Six-Tenth Factor Rule

4.1.3 Impact of the Selected Equipments on Capital Investment of the Plant

Figure 4.8a for CIS and 4.8b for 610FR show the impact of the selected equipment on the total capital investment of Palm Oil Refinery and Fractionation plant using the developed software. Figures 4.8c (CIS) and 4.8d (610FR) are used to show the various sectors in degrees using Microsoft excel package.

4.1.4 Effect of variation of the equipment cost on the total capital investment

Figure 4.9 shows the cost of crystallizer been increase to twice its initial cost and the corresponding effect on total capital investment of the plant using the two methods were obtained. Likewise, figure 4.10, for Pump and figure 4.11 for Olein Tank.

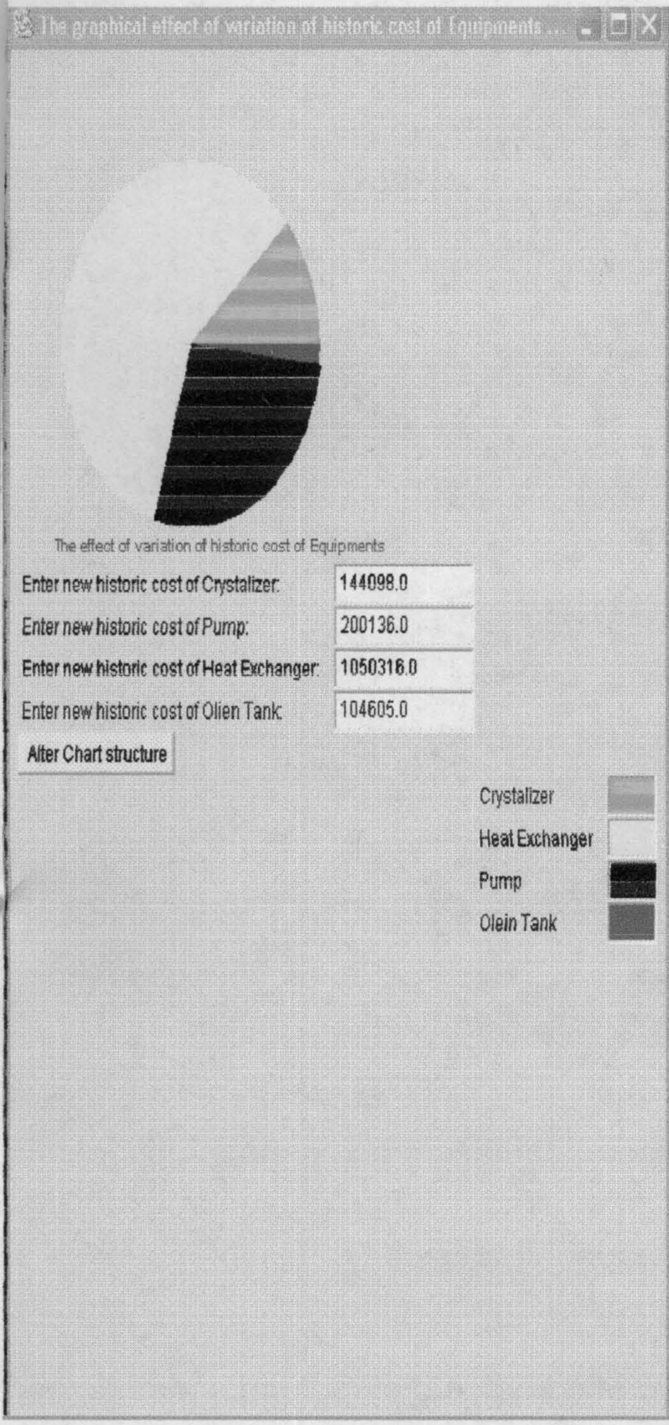


Figure 4.8a (For program CIS)

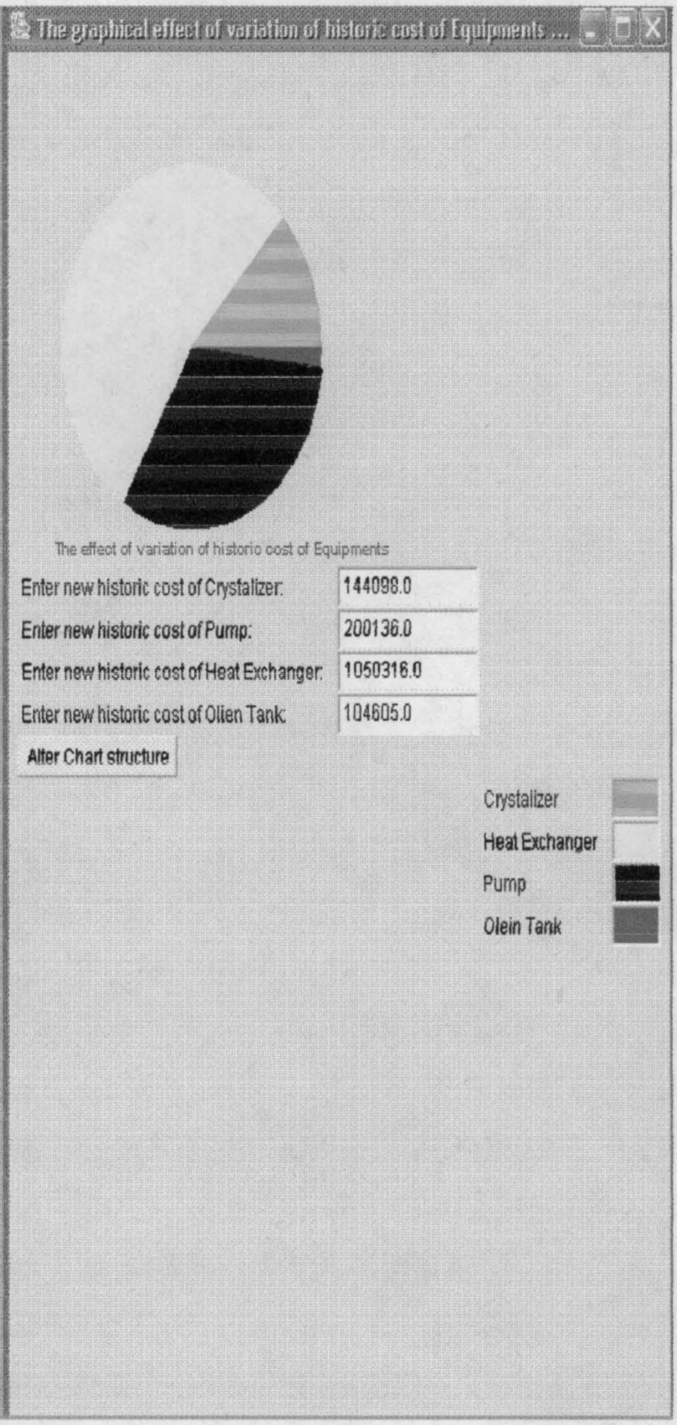
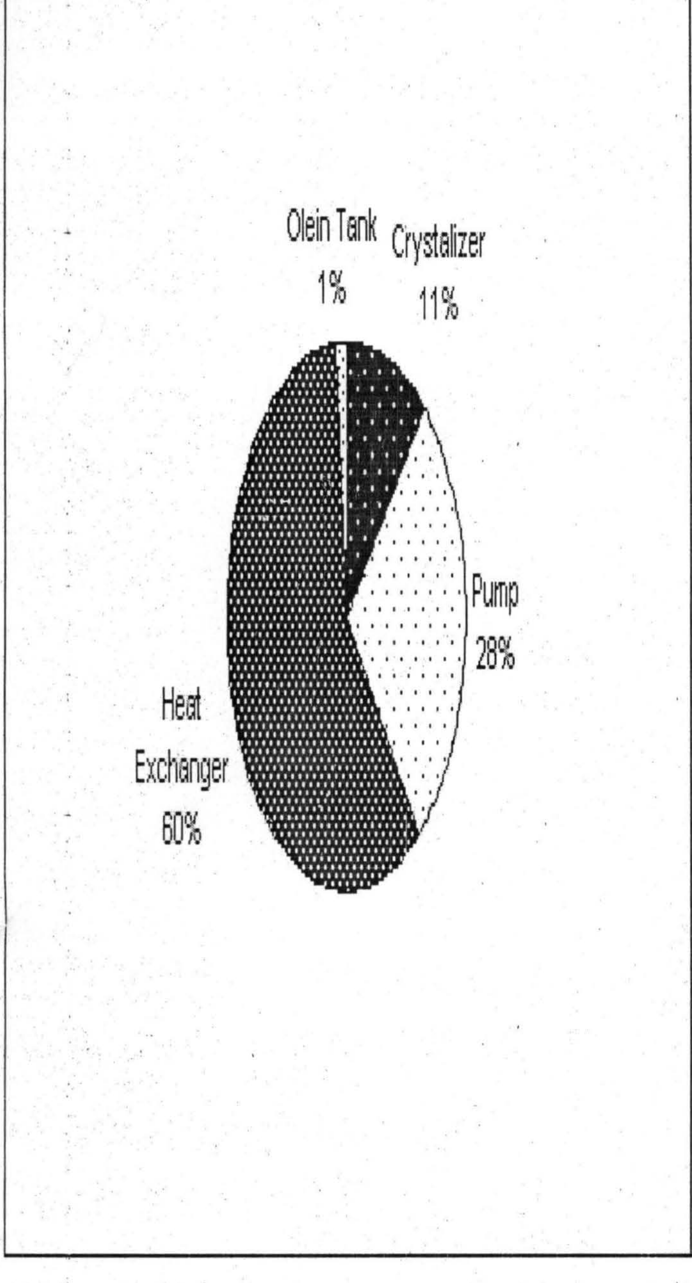


Figure 4.8b (For program 610FR)



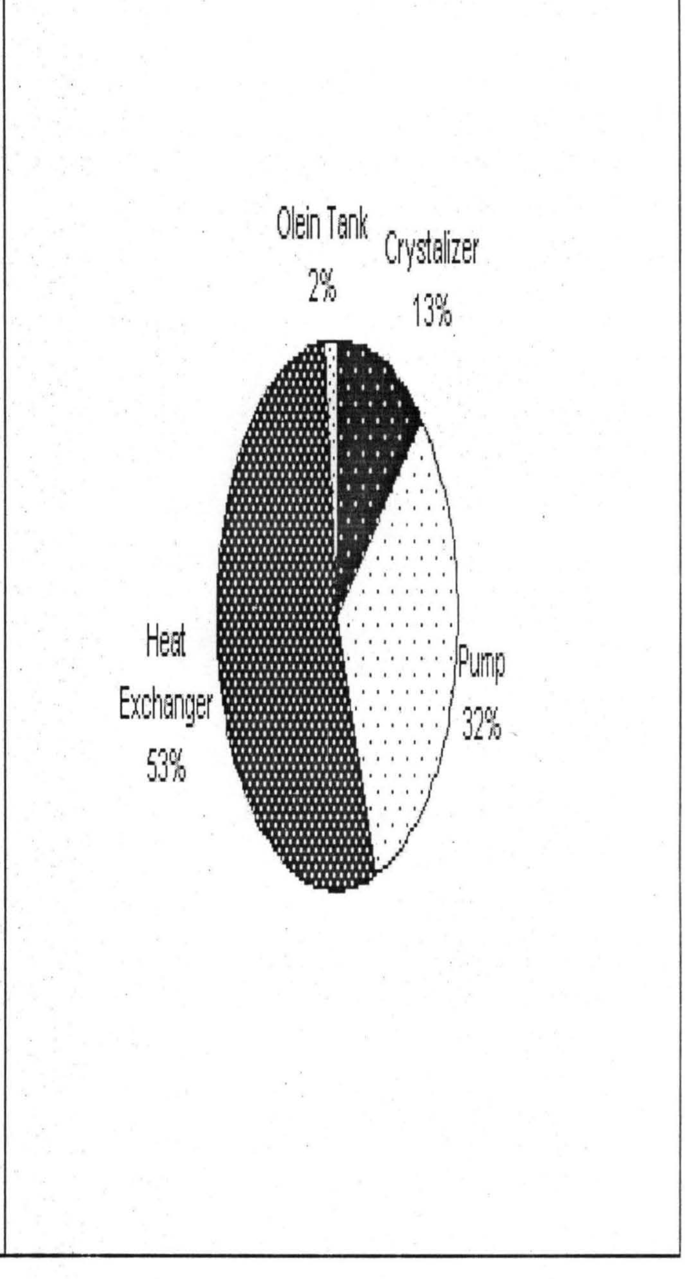


Figure 4.8c (For program CIS) **Figure 4.8d (For program 610FR)**

Graphs using Excel Package to show the Portions in Degrees

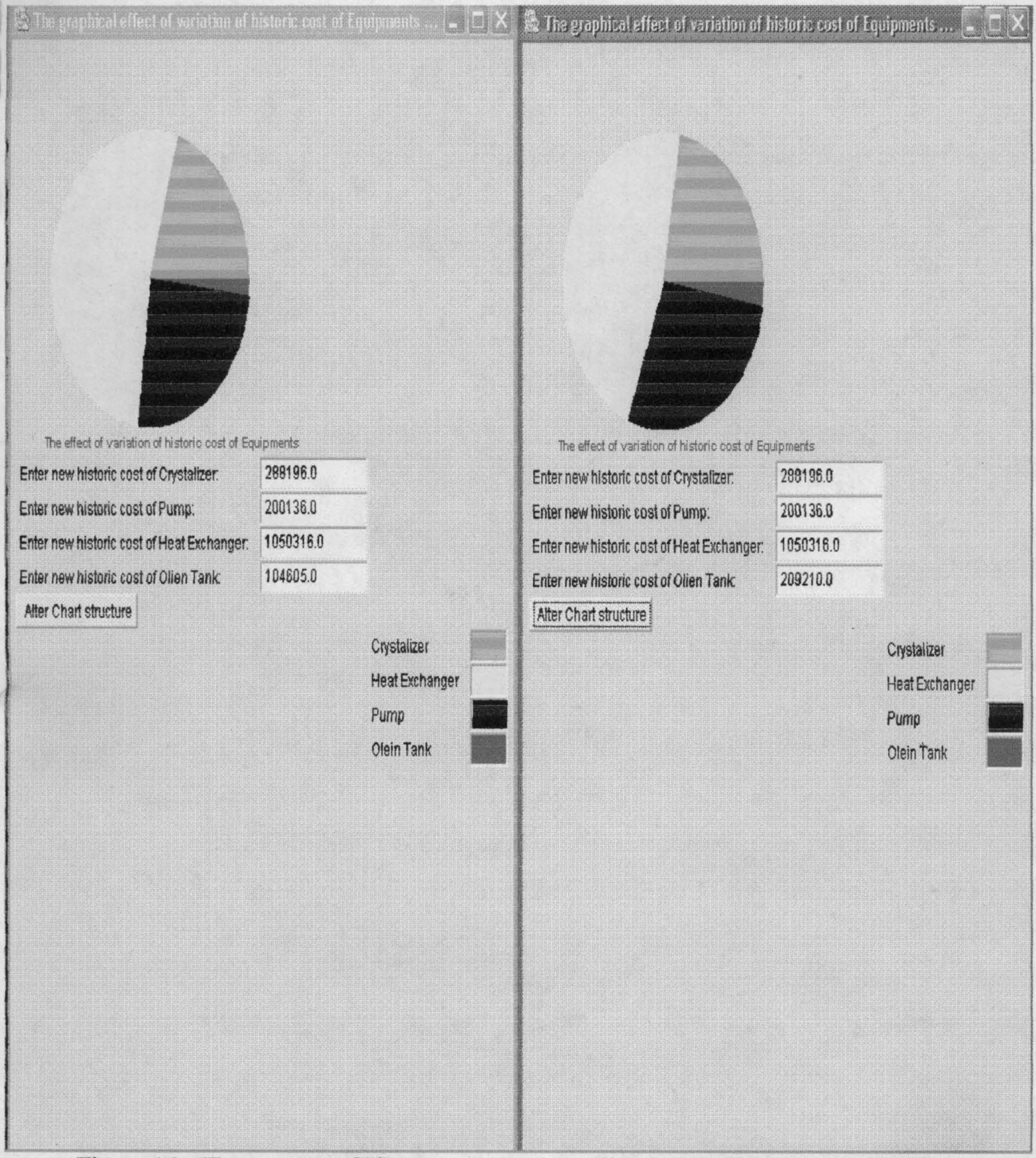


Figure 4.9a (For program CIS) **Figure 4.9b (For program 610FR)**

Graph Showing Twice the Cost of Crystallizer using the Two developed Software

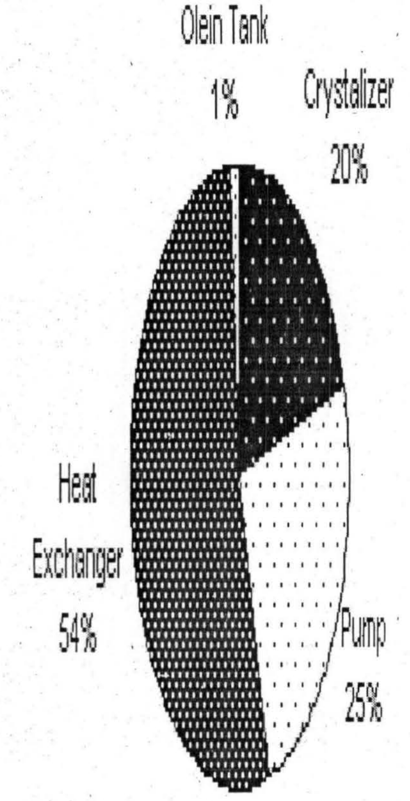


Figure 4.9c (For program CIS)

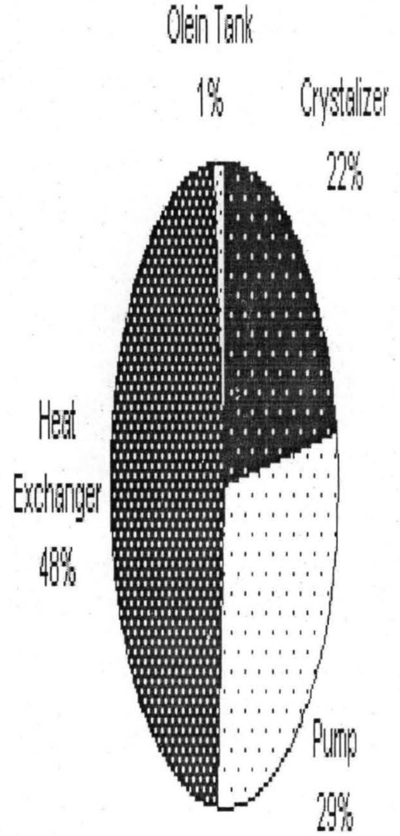


Figure 4.9d (For program 610FR)

Graphs using Excel Package to show the Portions in Degrees

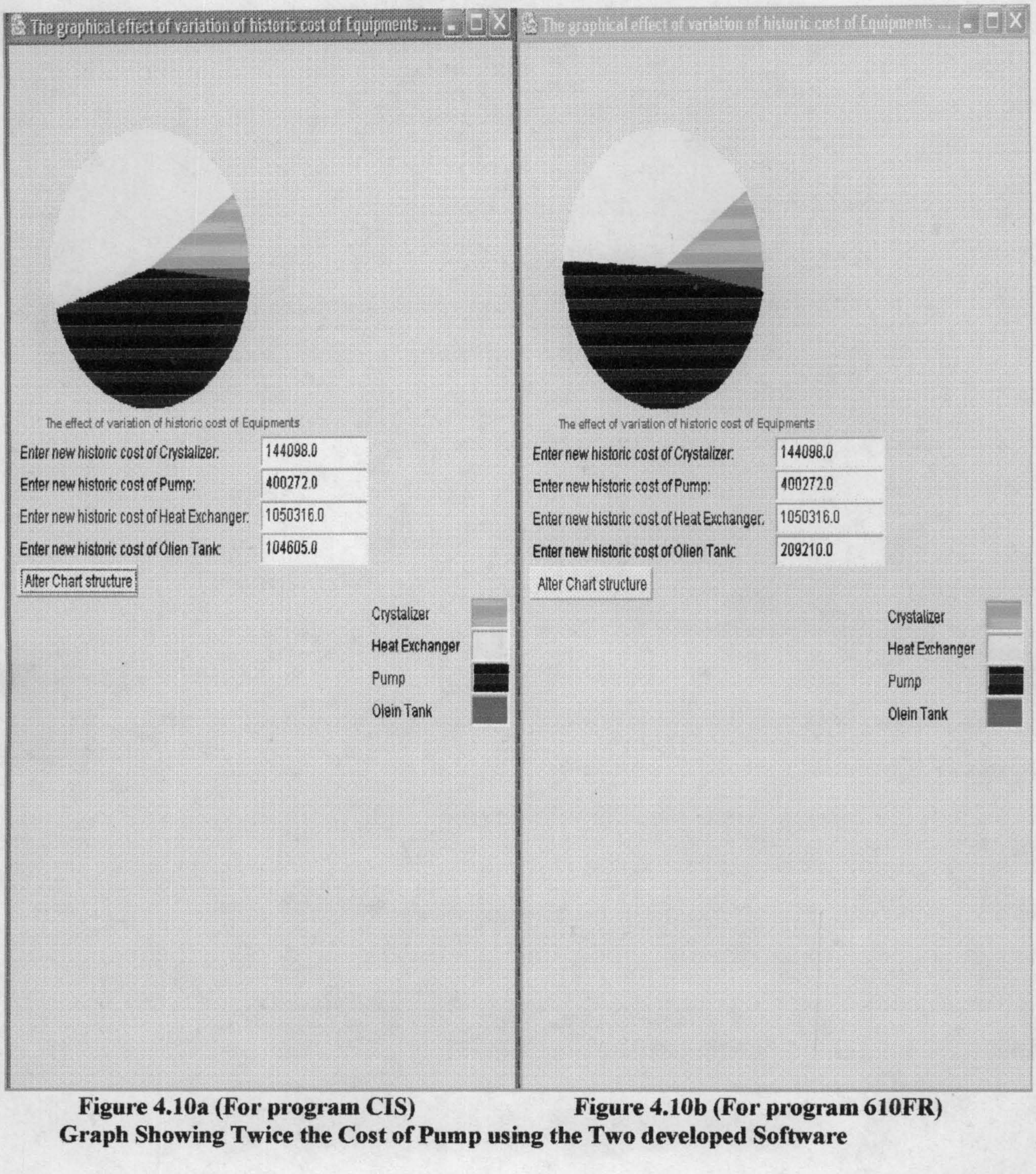
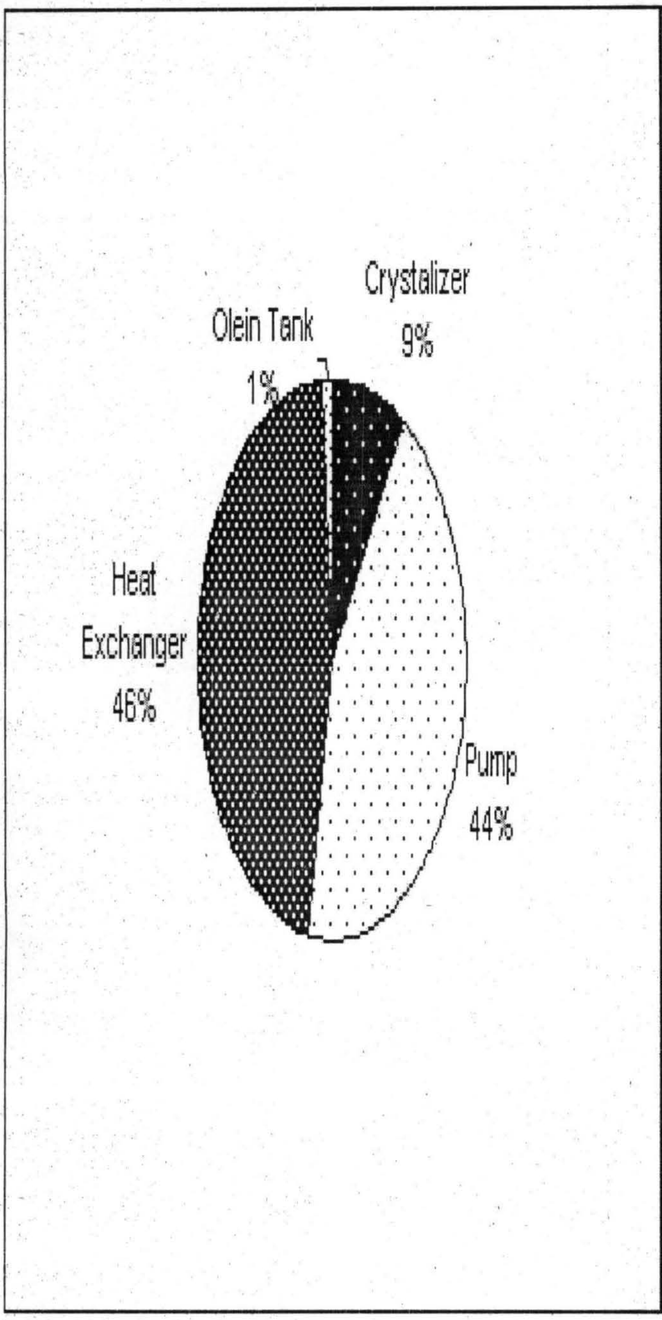


Figure 4.10a (For program CIS) **Figure 4.10b (For program 610FR)**
Graph Showing Twice the Cost of Pump using the Two developed Software



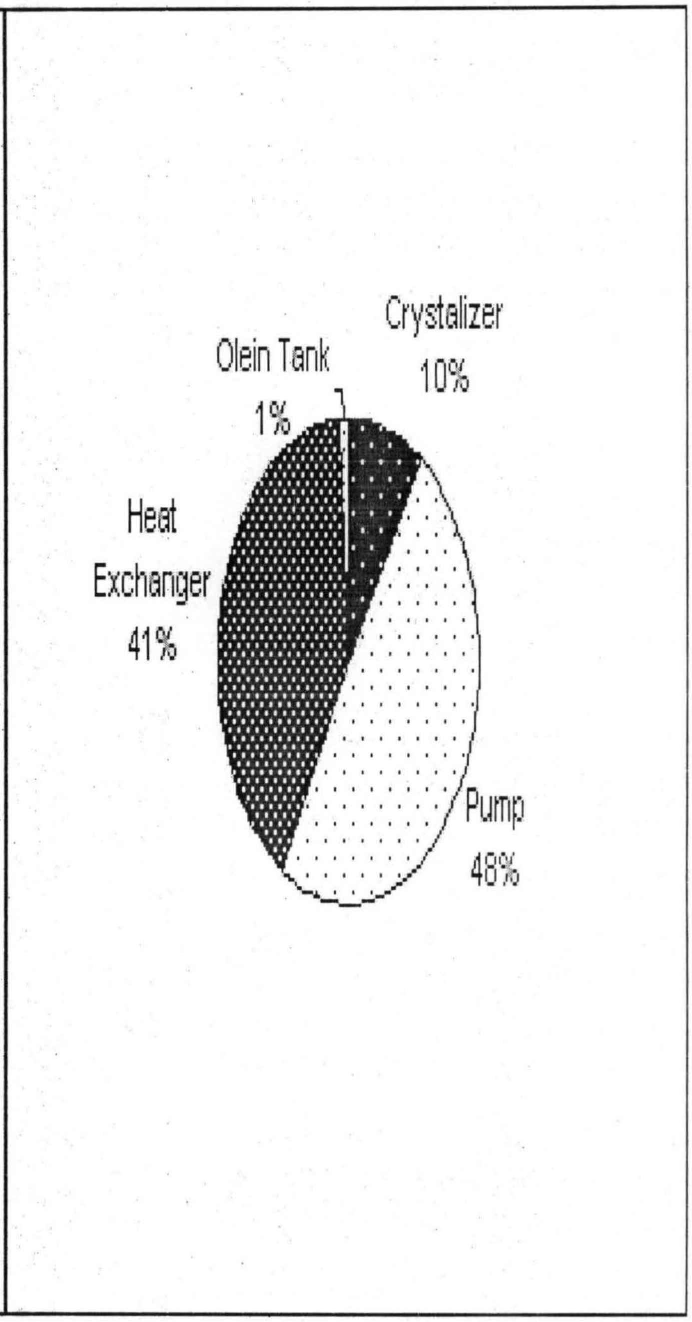


Figure 4.10c (For program CIS) Figure 4.10d (For program 610FR)
Graphs using Excel Package to show the Portions in Degrees

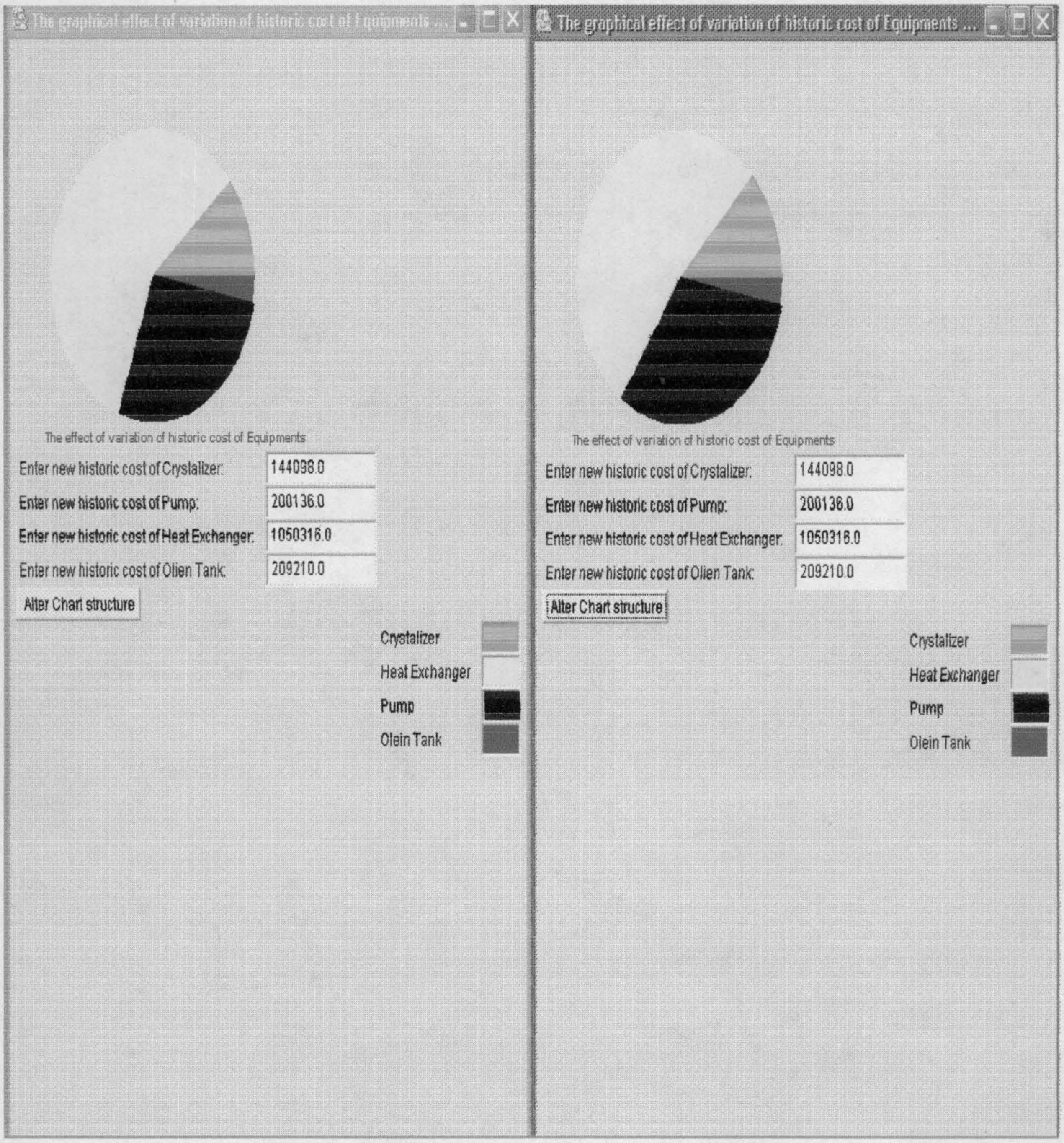
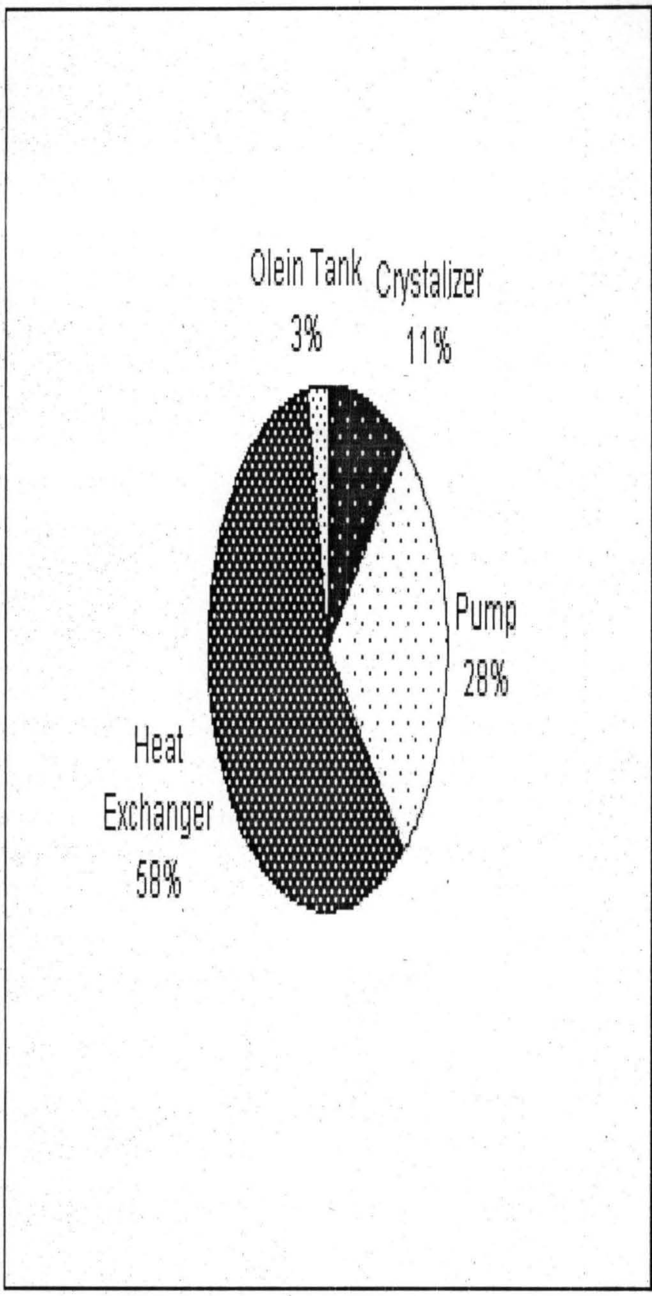


Figure 4.11a (For program CIS) Figure 4.11b (For program 610FR)

Graph Showing Twice the Cost of Olein Tank using the Two developed Software



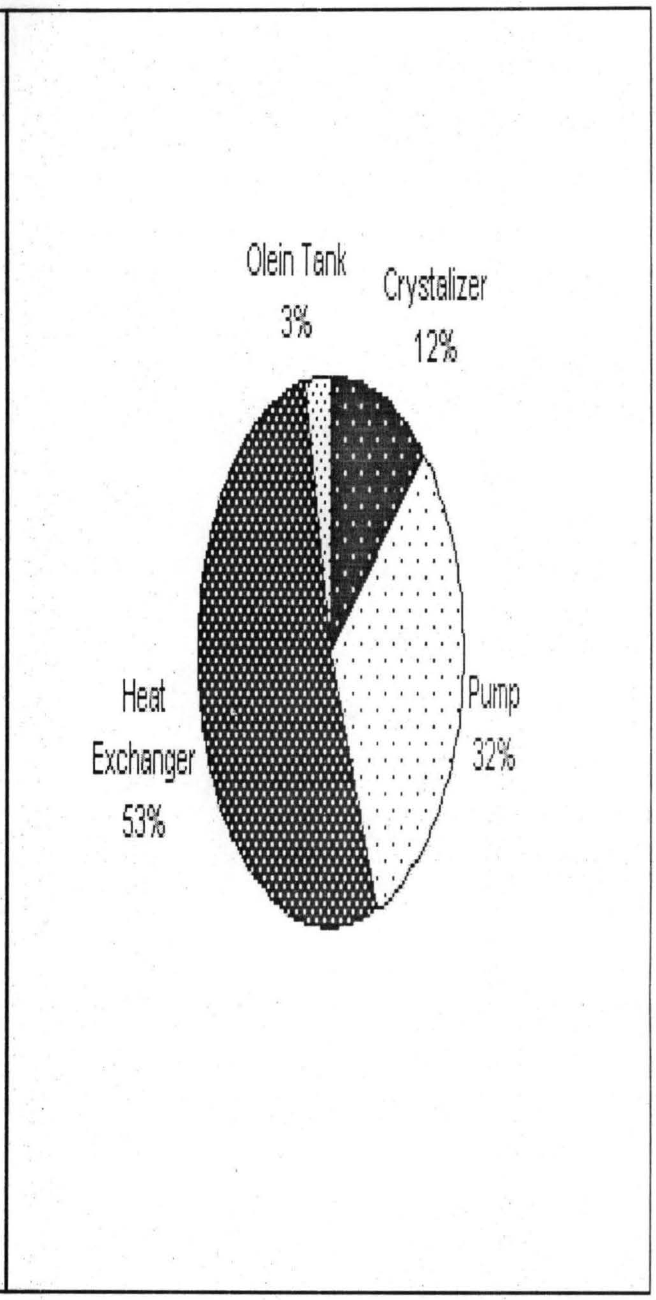


Figure 4.11c (For program CIS) **Figure 4.11d (For program 610FR)**

Graphs in Figure 4.6 using Excel Package to show the Portions in Degrees

.2 DISCUSSIONS OF THE RESULTS

he above obtained results are discussed as follows

.2.1 Result for total cost of palm oil refinery and fractionation plant

A capital investment is required for any industrial process and determination of the necessary ıvestment is an important part of such plant design. Consequently, two cost estimation methods vere used to achieve this goal and these are cost index and scaling method and six-tenth factor ule. The methods were represented by Programs CIS and 610FR whose capabilities can be escribed as flexible. The flexibility can be viewed in terms of input and output of data. Besides his, the ability to vary the design specification of any equipment for quick modification of the lant and ability to show at a glance the result in a graphical presentation is the first realization of he objectives of the project. The results obtained both graphically and numerically are as shown in igures 4.6 and 4.7. However, the result obtained by these methods can only be reliable within a eriod of ten years (Peters and Timmerhaus, 2002, Rudd and Watson, 1968). The time range onsidered in this research work are, 1996 for reference year and 2006 for the current year which s ten years time range and this is within the reliable accepted period.

The current cost which is the cost in 2006 of each of the major equipments for the two nethods is higher than the reference or historical costs, the cost in 1996. The cost of bleaching arth tank in 1996 was ₦104,604.00 while the cost in 2006 is ₦228,614.25 in figure 4.6 and ₦204,615.48 in figure 4.7 respectively. Moreover the cost of fatty acid Recuperator in 1996 was ₦98,200.00 which increased to ₦177,987.06 and ₦192,088.66 respectively in 2006. Comparing the numerical values of the results of the two methods as shown in figure 4.6 and 4.7, it is obvious that the cost data obtained from the Table in figure 4.7 are generally higher than the one from 'able in figure 4.6. The over-view of each table shows that the present cost is almost twice the istorical cost of each of the equipment and this can be said to be as a result of the followings:

- An increase in the value of chemical engineering's plant cost index (CEPCI), from 381.7 in 1996 to 492.6 in 2006(www.Eng-tip.com). This is caused by inflation rate, being the major determinant factor of the value of CEPCI.
- In addition to the change in the value of CEPCI, the plant capacity considered for 2006 is twice that for 1996.

The total cost of the plant using 610FR (figure 4.7) was observed to be about 12% greater than he cost using CIS (figure 4.6). The values are ₦360,952,704 for cost index and scaling method, vhile for six-tenth factor rule the value is ₦403,988,224 . However, better results are obtained by ısing the logarithmic relationship known as six-tenth factor rule as this has a fixed exponential 'alue, 0.6 (Peters and Timmerhaus, 2002). This is confirmed by the result obtained and uphold the 'iew that 610FR method is better because it's safer in most cases to over-estimate in design ›rocess than to under estimate.

I.2.2 Impact of the Selected Equipments on Capital Investment of the Plant

Comparing the numerical cost value for each of the equipments in Figure 4.6 and 4.7, it was liscovered that the cost of Heat exchanger, Crystallizer, Pump and Olein tank are much higher han the cost of other equipments. Figure 4.8a and 4.8b shows the impact of the computed current :ost of these four equipments on the estimated capital cost of the oil palm refinery and 'ractionation plant. From the graphs, it can be deduced that the cost of heat exchanger has the strongest influence on the estimated plant cost, 60% and 53% for both CIS and 610FR 'espectively as shown in figure 4.8c and 4.8d using an excel package. This then implies that any inflation cost trend of heat exchanger will greatly affect the estimated capital cost of the plant. This ı be applied in cost optimization of the plant by optimizing the relevant parameters such as terial, size, capacity etc. necessary to design the heat exchanger,. The cost optimization of the nt can also be achieved, for instance, by choosing a cheaper heat exchanger which can perform same function as the costly one, which in turn minimizes the cost of heat exchanger and nsequently reduces the capital investment of the plant.

.2.1 Effect of variation of the equipment cost on total capital investment

The graphs in figure 4.9 to 4.11 show various impacts of the selected equipments on the imated capital cost of the oil palm refinery and fractionation plant. The set of figures labeled "a" present the graph for cost index and scaling method while those labeled "b" represent six-tenth tor rule respectively and the set "c" and "d" show their respective sectors in degree using excel kage. When the cost of crystallizer is doubled, the cost of heat exchangers still dominates by cupying the largest portion of the chart, 54% and 48% respectively. The portion of heat changer in 4.9a is larger than that in 4.9b which is due to the difference in their computational ponential 0.37 and 0.6 for both CIS and 610FR respectively. This is also true when the cost of mp and Olein tank were also doubled, giving a graph whose heat exchanger's sectors were 46% d 41% for Pump and 58% and 53% for Olein tank. This shows that even though the cost of these uipments increases to as much as twice the initial value, the cost of heat exchanger still minates. This further confirms the influence of heat exchanger on the estimated capital cost of plant. However, the reverse is the case in figures 4.10b and 4.10d for 610FR as the cost of mps has more impact. Nevertheless, the cost of heat exchanger can still be said to have a nificant influence on the total cost despite the fact that the cost of some of the equipments were ubled. Hence any slight change in the cost of the heat exchanger will greatly affect the total pital cost of the Plant.

CHAPTER FIVE

CONCLUSIONS AND RECOMMENDATION

5.1 CONCLUSIONS

Two of the plant cost estimation methods, the method of cost index and scaling and six-tenth factor rule were respectively used in developing two robust computer programs, CIS and 610FR using a graph base language. The programs computed and evaluated equipment cost and various capital investments for the design of an oil palm refinery and fractionation plant. The programs also graphically displayed the equipment cost fluctuation on the estimated capital cost of the plant.

The programs were tested using 1996 cost data for the plant design and the corresponding current cost of the equipments were obtained. The estimated current total capital investment as at 2006 obtained for the plant were ₦360,952,704 as obtained from cost index and scaling method and ₦403,988,224 for six-tenth factor rule with heat exchanger having most influence.

5.2 RECOMMENDATION

There is further work on the software in other to adapt it to HTML so as to upload it into the department web browser or school website which could be a source of income to the department.

REFERENCES

Anon, Malaysian standard MS 814 (1983) in Hui, Y.H., Bailey's industrial oil and fat products Vol.2, Edible Oil and Fat products Oil and Oil seeds, John Wiley and sons Inc. New York. U.S.A p 283.

Baasel, W.D. (1990) Preliminary chemical Engineering Plant Design, 2nd Ed. Van Norstrand Reinhold, New York U.S.A. p442.

Bhagavan, H.N., and Padmanabhan, P. (1992). In Chow, C ed; Fatty Acids in food and their health implications. Marcel Dekker Inc., New Yark In: Bailey's Industrial Oil and Fat products Vol. 2 Edible Oil and Fat products Oil and Oil seeds, John Wiley and sons Inc. New York,U.S.A . p 23.

Codex Alimentarius Commission, Romc (1993), p 61 In: Hui, Y.H (1993). Bailey's Industrial Oil and Fat products Vol. 2 Edible Oil and Fat products Oil and Oil seeds, John Wiley and sons Inc. New York (. U.S.A. p 661.

Codex standard 125 (1981), Codex international standard for Edible Palm Oil. FAa/WHO in: Young, F.V.K (1981). Refining and fractionation of palm oil in: Gunstone, F.D (1981). Critical Report on Applied chemistry: Palm oil, John Wiley and Sons, New York. U.S.A. p 40.

Deitel, P. and Harvey, (2003), Java How to Program 5th Ed., Pearson Education, Inc. Upper River saddle River New Jersey U.S.A, p 1-1412.

Diffense, E.L., Am. Oil Chem. Soc 62: 376-85 (1985) in: Young, F.V.K (1987). Refining and fractionation of Palm Oil in: Gunstone, F.D., (1985). Critical Report on Applied Chemistry: Palm Oil John Wiley and Sons, New York. U.S.A. p 68 & 323.

Duke, J.A., and Wain, K.K. (1981). Medicinal plants of the world. Computer index with more than 85,000 entries. Vol.3

Forster, H.L., (1982). The determination of Oil Palm Fertilizer Requirements in Peninsular Malaysia part II: Effect of Different Environments. PORIM Bull., 4,46-56 (1982) in: Gunstonc, F.D (1982). Critical Report on Applied Chemistry Vol. 15, Palm Oil, John Wiley and Sons, New York, U. S. A. p17.

Green, A.G., (1986). Crop Science, 26, 961 in: Hui, Y.H (1996). Bailey's Industrial Oil and Fat Products Vol.2, Edible Oil and Fat products, Oil and Oil Seeds, John Wiley and sons Inc .. , New York, U. S. A. p 21.

Griffiths, G., Stymme, S. and Stobart, K. (1996). World conference on Biotechnology for oils and fats industry. Am. Oil chem .. Soc in Hui, H.Y (1996) Bailey's Industrial oil and Fat products: Edible Oil and Fat products Oil and Oil seeds. Vol.2. John Wiley and Sons Inc., New York, U. S. A. p 22.

Howes, P.D, Soon, T.C., Shaw, D.B. and Stemp, P.K (1993). Proceedings of PORIM International Conference on Progress, Prospects and Challenges towards the 21st century. Module II. Chem. & Tech., PORIM, Kuala Lumpur, Malaysia. p55.

Hui, Y.H. (1996). Bailey's Industrial Oil and Fat Products Vol. 2, Oil and Oil seeds, John Wiley and Sons Inc., New York, U.S.A. p 26.

Jacobsberg, B. (1983). PORIM Occasional paper No. 12, 11 in Hui, Y.H (1996). Bailey's Industrial Oil and Fat products, Vol. 1, John Wiley and Sons Inc., New York, U.S.A. p 315.

Jacobsberg, B. (1983). Quality of Palm Oil, PORIM Occasional Paper No.10 in: Young, F.V.K (1987). Refining and fractionation of palm oil in Gunstone, F.D. (1987) Critical Report on Applied Chemistry Vol. 15. Palm oil, John Wiley and sons New York U.S.A. p 46.

Kheiri, M.S.A (1985). J. Am oil Chem. Soc (62), 210 in Hui, Y.H. (1996). Bailey's Industrial Oil and Fat products, Vol., John Wiley and Sons Inc., New York, U.S.A. p 315.

Kirschenbauer B. (1960) Grasasy Aceitcs. 26.95. In: Young, F.V.K. (1987) Refining and Fractionation of palm Oil in: Gunstone, F.D (1987). Critical Report on Applied Chemistry; Palm Oil, Vol. 15, John Wiley and Sons, New York, U.S.A. p 63,

Lim, K.H. (1984). Workshop, Proc. Palm oil Res. Inst., Malaysia, (9), p 11- 441. In: Bailey's Industrial Oil and Fat products, Vol., John Wiley and Sons, New York, U.S.A. p 330 ..

Lim, l.H., Quah, S.K., Gillies, D., and Wood, B.J. (1984). Workshop, Proc palm oil Res. Inst, Malaysia p 42-52 in Bailey's Industrial oil and fat products Vol. I, John Wiley and Sons Inc .. , New York, U.S.A. p 340.

Maynard, J. (1972). Modula programming Butter worth & Co. Ltd, London, UK pp 45-1 00

McGraw-Hill Encyclopedia of Science and Technology (1997). p 199-202.

Moll, H.AJ (1987). The Economics of Palm Oil, Pudoe, Wageningen p 103- 104 in. Bailey's Industrial oil and fat products VoU, John Wiley and Sons Inc., New York, U.S.A. p 273.

Mordret, F. and Laurent, A.M (1978) Rev des corps Gras 25,245 in Bailey's Industrial oil and fat products Vol. 1, John Wiley and Sons Inc., New York, U.S.A. p 281.

Peters, M.S. and Timmerhaus, K.D (2002). Plant Design and Economics for Chemical Engineers McGraw-Hill Book Co., Singapore, pp 910 .

Perry R.H and Green D. (1984) Chemical Engineering Handbook, 6th Ed. McGraw-Hill, New York U.S.A.

Popper, H. and Weismantel, G.E. (1970). Costs and Production in the Inflationary 1970's, Chern. Eng., Jan 12, 1970, pp 132-136. In: Modem Cost Engineering: Methods and Data (1979), Chemical Engineering, McGraw-Hill Publishing Co., New York, U.S.A. p 299.

Resnick, W. (1981). Process Analysis and Design for Chemical Engineer, McGraw-Hill Book Co., New York, U.S.A. 363 p.

Rudd, D.E., and Watson, C.C (1968). Strategy of process Engineering, John Wiley and Sons New York, U.S.A. 400 p.

S/OCS-40 (Jan. 1994). Oil Crops in: Hui, Y.H. (1996) Bailey's Industrial Oil and Fat Products Vol. 1. John Wiley and Sons Inc., New York, U.S.A. p91.

Siew, W.L., Tang, T.S., Flingoh, C.H., and Rahman, A.A. (1990) PORIM Rep P.O (172) in: Hui, Y.H (1996). Bailey's Industrial oil and Fat products Vol. I, John Wiley and Sons Inc., New York, U.S.A. p 278.

Sonntag, N.O.V. in Swem, D. (1979). In Hui, Y.H. (1996) Bailey's Industrial oil and Fat products, Vol. I John Wiley and Sons Inc., New York, U.S.A. pp 289-459.

Sinnott R.K.(2005) Chemical Engineering Design, 4th Ed., Elisevier Butterworth-Hieneman, New York U.S.A. p 180.

Strecker, L.R., Hasman, J.N., and Mazar, A. (1985). Proceedings of World conference in Emerging Technologies in Fats and Oil Industries, AOCS, 51 in: Hui, Y.H. (1996) Bailey's Industrial oil and Fat products, Vol. 1, John Wiley and Sons Inc., New York, U.S.A. p 315.

Swem, D. (1979). In Hui, Y.H. (1996) Bailey's Industrial Oil and Fat Products Vol. 1, John Wiley and Sons Inc., New York, U.S.A. p 281.

Swoboda, P.A.T (1985). J. An. Oil Chem. Soc. 62, 287. In Hui, Y.H. (1996). Bailey's Industrial oil and Fat products Vol. 1, John Wiley and Sons Inc., New York, U.S.A. p 315.

Tan, B.K. (1979). Paper at Conference on Chemical Research in Malaysia, Held in Kuala Lumpur, March 19-21, Kuala Lumpur, Malaysian Institute of Chemistry in Hui, Y. H.(1996) Bailey's Industrial oil and fat products Vol. 2, Edible Oil and fat products Oil and Oil seeds, John Wiley and Sons Inc., New York, U.S.A. p 278.

Tang, T.S., Teoh, K. T., and Lee, Y.Y., (1983). Palm oil product Technologies in the Eighties Incorp. Soc. Planter, Kuala Lumpur. P 195 in Hui, YH. 1996). Bailey's industrial oil and Fat products, Vol 2, Edible Oil and fat products Oil and Oil seeds, John Wiley and Sons Inc., New York, U.S.A. p 281.

Vatavuk, W.M., (1990). Estimating Cost of air pollution control, Lewis Publishing, Chelsea, pp1- 142

Wilhems, M.G.A (1985) J. Am oil Chem. Soc. (62),454 in Hui, Y.H (1996) Bailey's Industrial Oil and Fat Products Vol. 2, Edible Oil and fat products and Oil seeds, John Wiley and Sons, New York, U.S.A. p 316.

Wong, S., Goh, L.C., Tan, L.C. and Kerk, P.P., (1993) Proceedings of PORM International Palm Oil Conference Module III: Chem. & Tech. P. 90. In: Hui, YH. (1996) Bailey's Industrial Oil and Fat Products Vol. 2, Edible Oil and fat products Oil and Oil seeds, John Wiley and Sons, New York, U.S.A. p 323.

Wood, B.J. (1987). Growth and Production of Oil Palm Fruits. In: Gunstone, F.D. (1987). Critical on Applied Chemistry; Palm Oil, Vol. 15, John Wiley and Sons, New York, U.S.A. pp 11-12.

www.eng-tips.com/gviewthread.cfm/lev217/lev3/37pidl124/qid/10053(2006). Chemical Plant Operations-cost Index for Equipment in Chemical Plant Design, August 2, 2006

Yap, S.C., Choo, Y.M., Ooi, C.K., Ong, A.S.H., and Goh, S.H. (1991). Elaeis 3(2), (12), 375 in Hui, Y.H (1996) Bailey's Industrial oil and Fat products Vol. 2, Edible Oil and fat products Oil and Oil seeds, John Wiley and Sons, New York, U.S.A. p 281.

Young, F.V.K. (1981) PORIM Tech (I), 31 in Hui, Y.H. (1996). Bailey's Industrial Oil and Fat products, Vol. 2, Edible Oil and fat products Oil and Oil seeds, John Wiley and Sons Inc., New York, U.S.A. p 315.

Young, F.V.K. (1987) Refining and fractionation of palm Oil in: Gunstone, F.D (1987). Critical Report on Applied Chemistry; Palm Oil, Vol. 15, John Wiley and Sons, New York, U.S.A. p 63.

APPENDICES

Appendices A, B, C, D are on the CD that accompanies this thesis.