

Design and Implementation OF probabilistic assignment Investment Management System. Using OOP

Dr.Rafah shihab Al-hamdani

Assistant prof

Iraqi commission for computers & informatics

Informatics Institute for Postgraduate Studies

IRAQ –Baghdad

Abstract:-

Computerized information system has recently become widely available. indeed ,it has become an important instrument that can be applied in different aspects of life .one of the important applications of computer systems is that adopted by commercial establishments as well as by small business .that ,we may admit that no management can successfully operate in any business without the use of computerized information system.

One of the important applications of computer is to use it in the management of establishments such as companies and banks. There are many different systems that are used to manage investment; these systems use different design philosophy such as top down approach or modular approach. In this paper, the object oriented programming concept is applied to design and implement **probabilistic assignment investment** management system for commercial bank. Most applications of OOP use specialized object oriented languages, but this paper shows that C++ can be used effectively to developed object oriented models of typical investment management ,and implement OOP is more flexible and reliable and least costing.

**تصميم وتنفيذ نظام إدارة تخصيص الاستثمار الاحتمالي باستخدام البرمجة الكيانية
oop.**

الخلاصة :-

في السنوات الأخيرة أصبحت نظم المعلومات المحوسبة واسعة الانتشار مما جعلها أداة مهمة في مختلف حقول الحياة.

من أهم التطبيقات لنظم المعلومات هي استخدامها في إدارة المؤسسات مثل الشركات والبنوك وحتى الأعمال الصغيرة، لذلك نستطيع أن نقرر انه ليس هناك إدارة ناجحة في مختلف الأعمال بدون أنظمة المعلومات

.هناك أنظمة مختلفة استخدمت في إدارة الاستثمار ولها فلسفات مختلفة في تصميم تلك الأنظمة مثل أسلوب من الأعلى إلى الأسفل، وأسلوب المعدل.في هذا البحث تم تطبيق البرمجة الكيانية لتصميم و

تنفيذ نظام ادارة التخصيص الاحتمالي للاستثمار في البنوك التجارية . معظم تطبيقات البرمجة الكيانية استخدمت لغة ++C وفي هذا البحث اتضح أنها تتسم بكفاءة عند الاستخدام بما يعزز تطور نماذج البرمجة الكيانية وخصوصا في إدارة تخصيص الاستثمار الاحتمالي كما اظهر البحث إن تنفيذ البرمجة الكيانية يعد مرنا واقل كلفة عند التطبيق .

Introduction:-

Computerized information system has recently become widely available... indeed; it has become an important instrument that can be applied in different aspects of life.

one of the important applications of computer systems is that adopted by commercial establishments as well as by small business .that, we may admit that no management can successfully operate in any business without the use of computerized information system .

Competition is one of the major factors which lead establishments to make profits or vise versa .however, adopting a rational competitive strategy may assist us in searching for competitive advantages, which the later has fruitful end result.

One of these results is to maximize the profit of the establishment, and at the same time to keep hold its position in the face of other competitive forces.

The competitive advantage could be obtained by designing strategic information system. The objected oriented program (OOP) is one of the philosophies that are being used in designing and implementing strategic information systems.

The investment has an important role in the economic growth process the investment variation leads to change in the employment, input, income and consumption.

In business, investment means that the purchase by a producer of a physical good, such as durable equipment or inventory, in the hope of improving future business Program.

Investment management is the professional management of various securities (shares, bonds etc) and other assets (e.g. real estate), to meet specified investment goals for the benefit of the investors. may be institutions (insurance companies, pension funds, corporations etc.) or private investors (both directly via investment contracts and more commonly via Collective investment scheme egg mutual funds)and The provision of 'investment management services' includes elements of, financial analysis ,asset selection, stock selection, plan implementation and ongoing

monitoring of investments .Investment management is a large and important global industry in its own right responsible for creating of trillions of dollars, euros, pounds and yen. Coming under the remit of financial services. Many of the world's largest companies are at least in part **investment managers** and employ millions of staff and create billions in revenue. May often refer to their services as **wealth management** or **portfolio management** often within the context of so-called "private banking".

This paper is devoted to the study of the probabilistic assignment model. To motivate this study, we first discuss the deterministic assignment model. Think of allocating indivisible objects to agents who have strict preferences over these objects. Each agent can consume one and only one object. The idea is to discuss general methods (henceforth rules) of assigning objects to agents, as a function of agents' preferences. As each object may be given to at most one agent, there may be conflicts of interests.

Suppose a rule recommends an assignment for a particular group of Agents, objects, and preferences (henceforth referred to as a "BANK").

To illustrate the concept of probabilistic assignment model we can consider the long term deposit schemes working in the commercial banks .the banks provide different interest rates for different schemes as well as for different periods of investment .Imagine that a subgroup of the agents "reapplies" the rule to the BANK consisting of themselves, the objects they were assigned, and their induced preferences over these objects. The assignment recommended by the rule for this "reduced" BANK assigns each agent in the subgroup to the object he was initially assigned.

The probabilistic assignment model is a simple generalization of this deterministic environment, motivated by fairness considerations. For example, a common requirement in Models of fair allocation are that rules should not discriminate between agents with similar characteristics. This is the well-known property **of equal treatment of equals**. Clearly, as objects are indivisible, this property cannot generally be satisfied. By allowing rules to randomize over assignments; however, we can ensure that agents with similar characteristics.

A program that allows an individual to have a set amount electronically transferred from one account to another at specified frequency is an investment management program

Object oriented programming is an approach to computer programming develops over the past twenty years. It is based on the idea that programs

should represent the interaction between abstract representations of real objects. This differs from procedural programming, which corresponded to a step-by-step list of computations without any necessary correspondence to natural processes.

The present model (investment management) has been built using an object oriented approach which has several advantages for this kind of implementation, Investment management system can be described in much more natural form using an object oriented programming (OOP) approach and the different object may interact with each other in different forms offering great flexibility in the implementation.

One of the most useful features of (OOP) is the ease with which major change in models can be introduced with a minimum recording .Thus ,an (OOP) language (c++)can model the complex Investment management system very flexibly and simply ,so this paper presents a brief overview of object oriented programming in investment management system in bank .

A probabilistic assignment model

Let N be a set of potential agents such that $|N| \geq 3$. The set of nonempty, finite subsets

Of N is denoted $\mathcal{N}$, with typical element N . Let X be a set of potential objects such that

$|X| \geq 3$. The set of nonempty finite subsets of X is denoted $\mathcal{X}$, with typical element X .

Let $X \in \mathcal{X}$. A preference R over X is a binary relation over X which is complete, transitive, and anti-symmetric. Thus, preferences are strict. Let the set of preferences over X be denoted $R(X)$. An BANK is consisting of a pair $(N, X) \in \mathcal{N} \times \mathcal{X}$ such that $|N| = |X|$, and a preference profile $R \in R(X)^N$. The set of BANKS is denoted by E .

An assignment for (N, X) is a connection between N and X . A typical assignment will be written μ . The set of all assignments for (N, X) is denoted $A(N, X)$. A probabilistic assignment for (N, X) is a probability distribution over $A(N, X)$. The set of probabilistic assignments for (N, X) is denoted $\Delta(N, X)$. Note that risk preferences of agents are not Observed; so that all assignment decisions must be made using only ordinal information.

A rule r is a correspondence which associates with each BANK (N, X, R) a non empty subset of $\Delta(A(N, X))$.

Any assignment can be identified with a permutation matrix. Thus, by the theorem of Birkhoff and von Neumann, any probabilistic assignment

induces a bi stochastic matrix, and conversely reduced model in which assignments are specified as the set of bi stochastic matrices. For our purposes, we are not justified in working directly with the set of bi stochastic matrices.

The reason is that the mapping which takes the set of probabilistic assignments into the set of bi stochastic matrices is not one-to-one. Thus, many probabilistic assignments may induce the same bi stochastic matrix. This is without loss of generality when discussing concepts which relate only to the welfare levels of agents; but when discussing conditions Which relate ***directly to the structure*** of assignments, it is not without loss of generality.

Look at object oriented programming

An approach provides away of modularizing programs by creating partitioned memory area for both data and functions that can be used as template for creating copies of such modules on demand.

OOP treats data as a critical element in program development and dose not allow it to flow freely around the system .it ties data more closely to function that operate on it and protects it from accidental modification from outside functions.

OOP allows us to decompose a problem into a number of entities called objects and then builds data and functions around these entities The data of an object can be accessed only by functions associated with that object. However functions of one object can access the functions of other objects.

The entire set of data and code of an object can be made a user define data type with help of a class.

In fact, the central concept of object oriented programming is the concept of a class, which also one of the most intuitive and appealing feature the view point of Investment management.

A class describes both the structure of an object and a set of procedures of initializing and using it .once a class has been defined, we can create any number of object belonging to that class with which they are created.

Classes thus a collection of objects of similar type .classes are user – defined data type and behave like the built-in type of programming language. data encapsulation is the most striking of class, the data is not accessible to the out side world and functions, which are wrapped in the class can access it .however, we can say that the wrapping up of data and functions into a single unit (called class) known as encapsulation, also a

classes use the concept of data abstractions that refers to act of representing essential features .Without including the background or explanation.

This paper presents a brief overview of object-oriented programming in the context of Investment management

The structure of object –oriented model

In order to appreciate the potential value of object-oriented modeling it helps to look at the relationship between models and the Investment management they are intended to represent .we put the following set of definitions:

- 1- We design the algorithm of Investment management system by missing OOP philosophy .since this philosophy has many features that make design strategy easier and reliable than other design philosophy.
- 2- We decide to use c++ as a tool to implement the algorithms designed by OOP philosophy because this language is most popular one, and in cooperate all features of design of OOP design methodology.
- 3- The model consider the long term deposit schemes working in commercial banks .the banks provide different interest rates different schemes as well as for different periods of investment .the following program illustrates how to use the class variable for holding account details and how to construct these variable at run time using dynamic initialization.
- 4- OOP simulation model consist of objects with complex internal dynamics which interact with each other. The rules of interaction are determined by properties of the object.
- 5- In this simulation model class objects can be initialized dynamically too. That is the initial value of an object may be provided during run time the advantage of dynamic initialization is that we can provide initialization formats, using overloaded constructors this provide the flexibility of using different format of data at runtime depending upon the situation.
- 6- C++ provides a special member function called constructor which enables an object to initialize itself when it created .this is known as automatic initialization of objects.
- 7- The program uses three overloaded constructors .the parameter values to these .constructors are provided at run time .the user can provide input in one of the following forms:
 - a- A mount, period and interest in decimal form.
 - b- A mount, period and interest in percent form.
 - c- A mount and period.

The second constructor is invoked for the form (1) and (3) and the third is invoked for the form (2).

8- class objects can be initialized dynamically too .that is the initial value of an object may be provided during run time .one advantage of dynamic initialization is that we can provide variable initialization format , using different format of data at run time depending upon the situation.

9- The program also uses constructors with default arguments.

The program structure is shown in the following figure (1)

Class investment
Data : Long int i-amount; Int time-year; Float investment rate; Float investment rate-value;
Method: Investment (){ }; investment{long int p,int y,float r=0.10}; investment{long int p,int y,float r}; Void output (void);

Figure (1): program structure

Class name =investment

Principle amount = i-amount

Investment period, year= time-year;

Interest rate, decimal form= investment rate;

Interest rate, percent form =investment rate-value;

8- the program consider the long term deposit schemes working in the commercial banks .the banks provide different interest rates for different schemes as well as for different periods of investment .the following program illustrates how to use the class variable for holding account details and how to construct these variables at run time using dynamic initialization

The program:

```
# include <iostream.h>
Class investment
{
    Long int i-amount;
    Int time-year;
    Float investment rate;
    Float investment rate-value;
    Puplic:
    Investment ( ){ };
    investment{long int p,int y,float r=0.10};
    investment{long int p,int y,float r};
    Void output (void);
};

    Investment:: investment (long int p,int y, float r)
    {
    i-amount=p;
    years=y;
    investment rate=r;
    investment rate-value=i-amount;
    for(int i=1;i<=y;i++)
        investment rate-value= investment rate-value*(1.0+r);
    }

Investment:: investment (long int p,int y, float r)
{
i-amount=p;
years=y;
investment rate=r;
investment rate-value=i-amount;
for(int i=1;i<=y;i++)
    investment rate-value= investment rate-value*(1.0+ float r/100);
}
Void investment :: output (void)
{
    Cout <<"\n"
        <<"principle amount<<r-value<<"\n"
        <<"return value =  "<<i-amount<<"\n" ;
}
```

```

Main ( )
{
Investment raf1,raf2,raf3; //deposit created
Long int p ; //principle amount
Int y ;    // investment period ,years
Float r ;  //investment rate ,decimal form
Int R;    //interest rate, percent form
Cout<<"enter amount, period, interest rate (in percent) <<"\n";
Cin>>p>>y>>R;
Raf1=investment (p, y, R);
Cout<<" enter amount, period, interest rate (decimal form) <<"\n";
Cin>>p>>y>>r;

Raf2=investment (p,y,r);
Cout<<" enter amount, period" <<"\n";
Cin>>p>>y;
Raf3=investment(p,y);
C0ut<<"\n Deposit1";
Raf1.output( );
C0ut<<"\n Deposit2";
Raf2.output( );
C0ut<<"\n Deposit3";
Raf3.output( );
}

```

Summary:-

This paper introduces a notion for the probabilistic assignment model, .We show that the axioms *equal treatment of equals* and *probabilistic* characterize the uniform rule, which is the rule which randomizes uniformly over all possible assignments.

Object oriented programming offers many advantages for developers of **a probabilistic assignment** investment management system models .we concluded from this paper that OOP approach to design **a probabilistic assignment** Investment management system introduces some benefits that other approaches can introduce it. Some of these benefits are reusability that enables us to used previously written functions in our paper.

Another feature is flexibility that means we can change some functions without changing program structure due to these features.

Whether Object oriented programming languages have reached the point where they offer the power and efficiency to support large –scale-modeling projects is not yet known.

Reference:

1. Bogomolnaia, A., Moulin, H., 2001. A new solution to the random assignment problem. *Journal of Economic Theory* 100, 295–328.
2. Bogomolnaia, A., Moulin, H., 2002. A simple random assignment problem with a unique solution. *Journal of Economic Theory* 19, 623–635.
3. Bulgar sami(1997),”object oriented programming with C++” ,john wally &sons ln.
4. Coplien,j.(1992),”advanced c++ .programming styles and idioms. reading, mass. : Addison -wisely publishing co.
5. Christopher P. Chambers*(2004),Consistency in the probabilistic assignment model, *Journal of Mathematical Economics* 40 (2004) 953–962
6. Danilij cougar (1975)”introduction to computer based information system”
7. John Wally &sons in.
8. Thomson, W., 2003. Consistent Allocation Rules. Manuscript. University of Rochester.