جَامعة
المَنارة
MANARA UNIVERSITY

مقرر برمجة ٢
الجلسة الخامسة عملي

أوجد خرج البرنامج الآتي.
ثم قم بالتعديل بحيث يكون t4 هو ساكن.
عدل مرة أخرى بحيث يكون t1,t2 هما global.

```
// constrDeconP47.cpp : main project file.

#include "stdafx.h"
#include <iostream>
using namespace std;
class Time {
public:
Time(int hr,int mn,int sc, char sp)
{    cout<<"construction object by values: ";
   cout<<hr<<"\t"<<mn<<"\t"<<sc<<endl;
   hour = hr;   minute =mn;   second = sc;
sep= sp;
}
~Time()
{   cout<<"deconstruction object having
values: ";

cout<<hour<<"\t"<<minute<<"\t"<<second<<endl;
}
```

```
private:
   int hour;          // 0 - 23 (24-hour clock
format)
   int minute;        // 0 - 59
   int second;        // 0 - 59
    char sep;          //:
}; // end clas Time
void main()
{  cout<<"MAIN START HERE "<<endl;
Time t1(10,20,30,':');
Time t2(30,40,50,':');
   {Time t3(15,25,35,':');
    Time t4(40,45,50,':');
   Time t5(11,22,33,':');}
   cout<<"MAIN STOP HERE "<<endl;
   system("pause");
} // end main
```

التوابع الأعضاء الثابتة والأغراض الثابتة:

عرف الملف الرأسي الآتي time.h

```
#ifndef TIME_H
#define TIME_H
class Time { public:
Time(int=0,int=0,int=0,char=':');
   void setTime(int,int,int,char);
   void setHour( int) ;
void setMinute( int) ;
void setSecond( int) ;
void setSep( char) ;
   int getHour() const;
   int getMinute() const;
   int getSecond() const;
   int getSep() const;
   void printUniversal() const;
   void printStandard();
private:   int hour;    // 0 - 23 (24-hour clock format)
   int minute;  /* 0 - 59  */    int second;  // 0 - 59
   char sep;    /* :   */ }; // end class Time
#endif
```

التوابع الأعضاء الثابتة والأغراض الثابتة:

الملف الثاني.

```
#include "stdafx.h"
#include <iostream>
#include <iomanip>
#include "time.h"
using namespace std;

Time::Time( int hour, int minute, int second,char sep)
{    setTime( hour, minute, second, sep);  } // end
constucter Time

void Time::setTime(int hour,int minute,int second,char
sep)
{   setHour( hour) ;                  setMinute( minute) ;
   setSecond( second) ;     setSep(sep) ;                }
// end setTime

void Time::setHour( int h)
{   hour = (h >= 0 && h < 24) ? h: 0; }        // end
setHour

void Time::setMinute( int m)
{    minute = (m >= 0 && m < 60) ? m: 0; }     // end
setMinute
void Time::setSecond( int s)
{    second = (s >= 0 && s < 60) ? s: 0; }     // end
setSecond
```

التوابع الأعضاء الثابتة والأغراض الثابتة:

الملف الثاني.

```
void Time::setSep(char sp)
{    sep = (sp == ':') ? sp: ':';}    // end setSep

int Time::getHour() const
{    return hour; }          // end getHour

int Time::getMinute() const
{   return minute; }         // end getMinute

int Time::getSecond() const
{    return second;}         // end getSecond

int Time::getSep() const
{    return sep;}  // end getSep

void Time::printUniversal() const
{   cout << setfill( '0') << setw( 2) << hour << sep<<
setw( 2) << minute << sep
        << setw( 2) << second;} // end printUniversal

void Time::printStandard()
{   cout<<((hour==0||hour==12)?12:hour%12)<< sep
<<setfill('0')<<setw(2)<<minute
        << sep <<setw(2)<< second <<(hour<12?" AM":" PM")
;} // end printStandard
```

التوابع الأعضاء الثابتة والأغراض الثابتة:

الملف الثالث main.
كيف يمكننا أن نعدل ساعة غرض ثابت مثل noon.
ما الغاية من تعريف طرق ثابتة؟ وماذا يحدث لو حذفنا منها كلمة const؟
كيف يمكن طباعة الوقت المعياري لغرض ثابت؟

```
// constFunConstObj.cpp : main project file.
#include "stdafx.h"
#include <iostream>
#include "time.h"
using namespace std;
int main()
{
Time wakeUp( 6, 45, 0, ':') ;    // non-constant object
   const Time noon( 12, 0, 0, ':') ;   // constant object
   wakeUp.setHour(18);// non-const object non-const
member function
//1   noon.setHour( 12) ;//const object non-const member
function
   wakeUp.getHour();  // non-const object const member
function
   noon.getMinute();  // const object const member
function
   noon.printUniversal();// const object const member
function
//2   noon.printStandard(); // const object non_const
member function

    system("pause");   return 0;
} // end main
```

تمرين:

- أنشئ صف classاسمه Roomيحتوي على ثلاث خصائص: الطول والعرض والارتفاع.
- أنشئ باني يقوم بتهيئة الحقول بقيم ابتدائية.
- أنشئ باني ثاني لتهيئة الحقول بقيم ممرة.
- أنشئ تابعين لحساب مساحة وحجم .Room
- ومن ثم في التابع الرئيسي:
- اشتق غرض Room1من الصف المنشأ واطبع مساحة الغرفة وحجمها.
- اشتق غرض Room2من الصف المنشأ وهيأ خصائصه بقيم مدخلة من قبل المستخدم واطبع مساحة الغرفة وحجمها.

```
#include <iostream>
using namespace std;
class Room
{private:
double length;
double breadth;
double height;
public:
Room(){length=3;
breadth=4;
height=5;}
Room(double l,double b,double h)
{length=l; breadth=b; height=h;}
double calculateArea()
{return length * breadth;}
double calculateVolume()
{return length * breadth * height;}
 };
```

```
int main() {
 // create object of Room class
Room room1;
cout << "Area of Room = " << room1.calculateArea() <<
endl;
cout << "Volume of Room = " << room1.calculateVolume()
<< endl;
double le,br,he;
// assign values to Parameters
cin>>le ;
cin>>br;
cin>>he;
Room room2(le,br,he);
// calculate and display the area and volume of the room
cout << "Area of Room = " << room2.calculateArea() <<
endl;
cout << "Volume of Room = " << room2.calculateVolume()
<< endl;
return 0;}
```

التوابع الصديقة:

عدل في الكود السابق ليصبح تابع حساب المساحة تابع صديق للصفRoom

```
int main() {
double le,br,he;
// assign values to Parameters
cin>>le ;
cin>>br;
cin>>he;
 // create object of Room class
Room room2(le,br,he);
// calculate and display the area and volume of the room
cout << "Area of Room = " << calculateArea(room2) <<
endl;
return 0;}
```

```
#include <iostream>
using namespace std;
class Room
{private:
double length;
double breadth;
double height;
public:
Room(){length=3;
breadth=4;
height=5;}
Room(double l,double b,double h)
{length=l; breadth=b; height=h;}
Friend double calculateArea();
double calculateVolume()
{return length * breadth * height;}};
double calculateArea(Room R)
{return R. length*R. breadth;}
```

ليكن لدينا الكود الممثل للصف Bank التالي :
أنشئ صف Employee له الخصائص التالية : الرقم الوطني id وقيمة الرصيد Balance. وأجري التعديلات المناسبة ليصبح الصف Bank صديقاً للصف Employee.

```
class Employee{
private:
int id=00;
int balance = 0;
friend class Bank; };
```

```
#include <iostream>
using namespace std;
class Bank {
int account = 60000;
public:
void getSalary(Employee &e , int salary){
cout << "before " <<e.balance << " " << account <<endl;
e.balance = e.balance + salary;
account = account - salary;
cout << "after " <<e.balance << " " << account <<endl;}};
```

جَامعة
المَنارة
MANARA UNIVERSITY

انتهت الجلسة