VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

OBJECT ORIENTED PROGRAMMING

DIGITAL ASSINGMENT – 2

REG NO: 19BCS0012

NAME: NITHISH.G

DATE: 28.02.2020

Qno-1

1. By applying the constructor overloading, create the two vectors or arrays which hold the Internal

Assessment Marks (IAM) and Final Assessment marks (FAM) of a class of students and grade the

students based on the total marks in Internal as well as Final. Follow the VIT grading system.

```
#include<iostream>
using namespace std;
#include<string.h>


class Students
{

   private:
```

```
        float IAM,FAM,total;

        char *grade;

    public:

Students()
        {
           IAM = 0;
           FAM = 0;
           total=0;
           grade = "nil";
        }

Students(float a,floatb,floatd,char *c)
        {
           IAM = a;
           FAM = b;
           total=d;
strcpy(grade,c);
        }

        void get();
        void calculation();
        void disp();

};

void Students::get()
```

```
{
cout<<"\n enter the students internal assesmentmark :";
cin>>IAM;
cout<<"\n enter the students FINAL assesmentmark :";
cin>>FAM;
}

void Students::calculation()
{
   total=IAM+FAM;

if(total<=100 && total>=90){grade="S";}
if(total<90 && total>=80){grade="A";}
if(total<80 && total>=70){grade="B";}
if(total<70 && total>=60){grade="C";}
if(total<60 && total>=50){grade="D";}
if(total<50 && total>=40){grade="E";}
if(total<40 ){grade="N";}

}

void Students::disp()
{
cout<<"\n\n internals : "<<IAM;
cout<<"\n\n finals : "<<FAM;
cout<<"\n\n grade : "<<grade;
}

int main()
```

```
{
   Students *a;
   int n,i;
cout<<"\n\n enter the internal marks for 60 and final marks for 40 : ";
cout<<"\n\n enter the number of students : ";
cin>>n;

   a= new Students[n];
   for(i=1;i<=n;i++)
   {
cout<<"\n enter the student "<<i<<" marks";
      a[i].get();
   }

   for(i=1;i<=n;i++)
   {
      a[i].calculation();
   }
   for(i=1;i<=n;i++)
   {
cout<<"\n the student "<<i<<" grades";
      a[i].disp();
   }
}
```

Qno-3

3**. Develop an OOP to get the purchase date of a machine and find the age of machine currently.**

```
#include<iostream>
using namespace std;

class Age
{
	private:
		int pd,pm,py,bd,bm,by;
		int md[12] = { 31,28,31,30,31,30,31,31,30,31,30,31 };
		int d,m,y;

	public:
		void gets();
		void calculation();
		void disp();

};
void Age::gets()
{
	cout<<"\n enter the purchased date of machine \n\n dd: ";
	cin>>bd;
	cout<<" mm: ";
	cin>>bm;
	cout<<" yy:";
	cin>>by;

	cout<<"\n enter the current or present date of machine \n\n dd: ";
	cin>>pd;
	cout<<" mm: ";
	cin>>pm;
```

```
        cout<<" yy:";
        cin>>py;
}

void Age::calculation()
{
        //year calculatuion
        y=py-by;
        //mont calculation
        if(pm<bm)
        {
                y--;
                m=bm-pm;
        }
        else
        {
                m=pm-bm;
        }
        //day calculation
        if(pd<bd)
        {
                m--;
                d=md[pm-1]-(bd-pd);
        }
        else
        {
                d=pd-bd;
        }
```

```
}

void Age::disp()
{
	cout<<"\n the current age of the machine is: ";
	/*cout<<"\n\n"<<y<<" years "<<m<<"months "<<d<<"days";*/
	if(y!=0)
	{
		cout<<"\n\n\t\t"<<y<<" years ";
	}
	if(m!=0)
	{
		cout<<m<<" months ";
	}
	if(d!=0)
	{
		cout<<d<<" days ";
	}
}

int main()
{
	class Age d;
	d.gets();
	d.calculation();
	d.disp();
}
```

Qno-5

5. Construct an OOP to find the distance travelled by a Vehicle in "t" seconds is given by

Distance = u x t = ((a x t_2) / 2).

Here, "u" is the initial velocity (meters per second) and "a" is the acceleration (meters per second$_2$).

Write an OOP to evaluate the distance travelled at regular intervals of time, given the values of "u"

and "a". The program should provide the flexibility to the user to select his own time intervals and

repeat the calculations for different values of "u" and "a".

```
#include<iostream>
using namespace std;

class k
{
        private:
                float u,t,a,d;
        public:
                void gets();
                void calculation();
                void puts();

};

void k::gets()
{

        cout<<"\n  enter the initial velocity (meters per second) : ";
        cin>>u;
```

```
        cout<<"\n  enter the acceleration (meters per secondsquare) : ";
        cin>>a;
        cout<<"\n  enter time taken to travel(seconds) :  ";
        cin>>t;
}

void k::calculation()
{
        d = u*t+((a*(t*t))/2);
}

void k::puts()
{
        cout<<"\n\n  distance traveled by the vehical : " <<d<<" meteres";
}

int main()
{
        class k d;
        int n;
        do
        {
        d.gets();
        d.calculation();
        d.puts();
        cout<<"\n\n\n if u want to do the calculation again press 1 else press any key :";
        cin>>n;
        }while(n==1) ;
        cout<<"\n\n\n.....thank you for using calculator.....\n\n\n"
```

```
}
```