C++ Programming

PART – B

# 1. Using constructors and proper methods, design a class graphics which stores shapes, area, back colour and fore colours. Use this class in the main program to input any ‘N’ shapes and perform the following operations and print the list in a neat format.

# a) Sort according to area

# b) Search for a specified shape

#include<iostream.h>

#include<conio.h>

#include<string.h>

#include<iomanip.h>

class graphics

{

public:

char shape[10];

float area;

char back\_col[10],fore\_col[10];

graphics()

{

shape[10]=' ';

area=0;

back\_col[10]=' ';

fore\_col[10]=' ';

}

void input()

{

cout<<"Enter the shapes:";

cin>>shape;

cout<<"\nEnter the area:";

cin>>area;

cout<<"\nEnter background colour:";

cin>>back\_col;

cout<<"Enter the fore colour:";

cin>>fore\_col;

}

void display()

{

cout<<setw(10)<<shape<<setw(10)<<area<<setw(10)<<back\_col<<setw(15)<<fore\_col<<endl;

}

};

void main()

{

graphics g[20],max;

int i,j,k,n;

char search[10];

clrscr();

cout<<"Enter the number of graphics:";

cin>>n;

for(i=0;i<n;i++)

{

cout<<"\n";

cout<<"\nEnter characters of graphics "<<i+1<<":"<<endl;

g[i].input();

}

cout<<"\n";

for(i=0;i<n;i++)

for(j=i+1;j<n;j++)

{

if(g[i].area>g[j].area)

{

max=g[i];

g[i]=g[j];

g[j]=max;

}

}

cout<<"\nAfter sorting graphs are:\n";

cout<<setw(10)<<"Shape"<<setw(10)<<"Area "<<setw(10)<<" Background colour"<<setw(10)<<" Fore colour"<<endl;

for(i=0;i<n;i++)

g[i].display();

cout<<"\nEnter the shapes to be searched \n";

cin>>search;

for(i=0;i<n;i++)

{

k=strcmp(g[i].shape,search);

if(k==0)

{

cout<<"\n Shapes is found \n";

g[i].display();

break;

}

}

if(k!=0)

cout<<"\n Shapes is not found";

getch();

}

/\*

*Output-1:*

Enter the number of graphics:2

Enter characters of graphics 1:

Enter the shapes: square

Enter the area:16

Enter background colour:pink

Enter the fore colour:white

Enter characters of graphics 2:

Enter the shapes:triangle

Enter the area:30

Enter background colour:black

Enter the fore colour:green

After sorting graphs are:

Shape Area Background colour Fore colour

square 16 pink white

triangle 30 black green

Enter the shapes to be searched

rectangle

Shapes is not found

*Output-2:*

Enter the number of graphics:3

Enter characters of graphics 1:

Enter the shapes: circle

Enter the area:10

Enter background colour:red

Enter the fore colour:white

Enter characters of graphics 2:

Enter the shapes:square

Enter the area:23

Enter background colour:black

Enter the fore colour:white

Enter characters of graphics 3:

Enter the shapes:rectangle

Enter the area:21

Enter background colour:blue

Enter the fore colour:white

After sorting graphs are:

Shape Area Background colour Fore colour

circle 10 red white

rectangle 21 blue white

square 23 black white

Enter the shapes to be searched

circle

Shapes is found

circle 10 red white

\*/

**2. Create a class ‘Bank’ which includes data members – Acno, Name, Balance and a parameterized constructor to initialize the data members and other methods like deposit, withdrawal, and display the details of the customer.**

**(Hint: Check for minimum balance of Rs. 500/- while opening the account and during the withdrawal. Also, amount should be positive integer. Otherwise show appropriate message.)**

#include<iostream.h>

#include<conio.h>

#include<ctype.h>

#include<string.h>

class bank

{

int ac\_no,balance,amt;

char name[20];

public:bank(char na[],int num,int bal);

public:void deposit();

public:void withdrawal();

public:void display();

};

bank::bank(char na[],int num,int bal)//parametarised //constructor to data member

{

strcpy(name,na);

ac\_no=num;

balance=bal;

}

void bank::deposit()

{

cout<<"Enter the amount to be deposited:";

cin>>amt;

while(amt<0)

{

cout<<"Amount has to be positive number,enter again the amount:";

cin>>amt;

}

balance=balance+amt;

}

void bank::withdrawal()

{

cout<<"Enter the amount to be withdraw:";

cin>>amt;

while(amt<0)

{

cout<<"Amount has to be positive number,enter again the the amount:";

cin>>amt;

}

if(amt>balance-500)

cout<<"Withdrawal is not possible due to low balance (minimum balance >=500) \n";

else

balance=balance-amt;

}

void bank::display()

{

cout<<"Account number:"<<ac\_no<<endl;

cout<<"Customer name:"<<name<<endl;

cout<<"Balance:"<<balance<<"Rs"<<endl;

}

void main()

{

int choice,r=1,p\_balance;

char p\_name[10];

clrscr();

cout<<"Enter the customer name to whom account to be opened:";

cin>>p\_name;

cout<<"Enter the opening balance amount:";

cin>>p\_balance;

while(p\_balance<=500)

{

cout<<"Opening balance to be minimum of Rs 500, Enter again the opening balance amount:";

cin>>p\_balance;

}

bank b(p\_name,123,p\_balance);//created object 'b' of type class bank and initialized through the parametarised constructor

while(r)

{

cout<<"Enter option\n[1. Deposit 2.Withdrawal 3.Display]\n";

cin>>choice;

switch(choice)

{

case 1: b.deposit();

break;

case 2:b.withdrawal();

break;

case 3:b.display();

break;

default:cout<<"Wrong choice \n";

}

cout<<"Do you want to continue (1/0) \n";

cin>>r;

}

getch();

}

/\*

*Output:*

Enter the customer name to whom account to be opened: Vignesh

Enter the opening balance amount: 100

Opening balance to be minimum of Rs 500.,Enter again the opening balance amount:600

Enter option

[1. Deposit 2.Withdrawal 3.Display]

1

Enter the amount to be deposited:-200

Amount has to be positive number,enter again the amount:300

Do you want to continue (1/0)

1

Enter option

[1. Deposit 2.Withdrawal 3.Display]

3

Account number:123

Customer name:Vignesh

Balance:900Rs

Do you want to continue (1/0)

1

Enter option

[1. Deposit 2.Withdrawal 3.Display]

2

Enter the amount to be withdraw:1000

Withdrawal is not possible due to low balance (minimum balance >=500)

Do you want to continue (1/0)

1

Enter option

[1. Deposit 2.Withdrawal 3.Display]

2

Enter the amount to be withdraw:200

Do you want to continue (1/0)

1

Enter option

[1. Deposit 2.Withdrawal 3.Display]

3

Account number:123

Customer name:vignesh

Balance:700Rs

Do you want to continue (1/0)

0

\*/

**3 Write a program to accept two strings and using operator overloading perform the following.**

**a) Concatenation of two strings.**

**b) Comparison of two strings alphabetically.**

**(Note : For concatenation (+), for comparison (==, > or<)**

#include<iostream.h>

#include<conio.h>

#include<iomanip.h>

#include<string.h>

class string

{

private: char str[100];

public:

string()

{

strcpy(str," ");

}

string(char s[])

{

strcpy(str,s);

}

void display()

{

cout<<str<<endl;

}

string operator+(string ss)

{

string temp;

strcpy(temp.str,str);

strcat(temp.str,ss.str);

return temp;

}

int operator==(string ss)

{

if(strcmp(str,ss.str)==0)

return (1);

else

return (0);

}

int operator<(string ss)

{

if(strcmp(str,ss.str)<0)

return (1);

else

return (0);

}

};

void main()

{

clrscr();

char str1[25],str2[25];

cout<<"Enter two strings: "<<endl;

cin>>str1>>str2;

string s1(str1);

string s2(str2);

string s3;

s3=s1+s2;

cout<<"Concatenated string is : ";

s3.display();

if(s1==s2)

cout<<"Two strings are equal";

else if(s1<s2)

cout<<"First string is alphabatically above the second string";

else

cout<<"Second string is alphabatically above the first string";

getch();

}

/\*

*Output-1:*

Enter two strings:

vig

nesh

Concatenated string is : vignesh

Second string is alphabatically above the first string

*Output-2:*

Enter two strings:

vignesh

vignesh

Concatenated string is : vigneshvignesh

Two strings are equal

*Output-3:*

Enter two strings:

computer

science

Concatenated string is : computerscience

First string is alphabatically above the second string

\*/

**4 Create a class ‘Time’ which includes the data members – hours, minutes and seconds. Write a menu driven program with the following methods to**

**a) accept time**

**b) display time**

**c) increment time by one second by overloading unary operator ++**

**d) decrement time by one second by overloading unary operator - -**

**(Hint: Validate minutes and seconds to be in the range of 0-59 in input and output).**

#include<iostream.h>

#include<conio.h>

#include<iomanip.h>

#include<process.h>

class time

{

private: int hr;

int min;

int sec;

public:time()

{

hr=0;

min=0;

sec=0;

}

void get\_time()

{

cout<<"Enter the time:";

cin>>hr>>min>>sec;

if(hr<0||hr>23||min<0||min>59||sec<0||sec>59)

{

cout<<"Enter the proper time, hours to be in the range 0-23 , minutes and seconds to be in the range 0-59 : \n";

cin>>hr>>min>>sec;

}

}

void show\_time()

{

cout<<hr<<":"<<min<<":"<<sec<<endl;

}

void operator++(int)

{

sec++;

if(sec>=60)

{

min++;

sec=0;

}

if(min>=60)

{

hr++;

min=0;

}

if(hr>23)

{

hr=0;

}

}

void operator--(int)

{

sec--;

if(sec<0)

{

min--;

sec=59;

}

if(min<0)

{

hr--;

min=59;

}

if(hr<0)

{

hr=0;

}

}

};

void main()

{

time t1;

int ch;

clrscr();

do

{

cout<<endl<<"1.Accept the time:\n";

cout<<"2.Show time \n";

cout<<"3.Increment the time \n";

cout<<"4.Decrement the time \n";

cout<<"5.Exit \n";

cout<<"Enter your choice:";

cin>>ch;

switch(ch)

{

case 1:t1.get\_time();

break;

case 3:t1++;

break;

case 4:t1--;

break;

case 2:

cout<<"The time is - ";

t1.show\_time();

break;

case 5:exit(0);

}

}

while(ch!=0);

getch();

}

/\*

*Output:*

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:1

Enter the time:10 20 59

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:2

The time is - 10:20:59

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:3

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:2

The time is - 10:21:0

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:5

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:1

Enter the time:10 20 00

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:4

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:2

The time is - 10:19:59

1.Accept the time:

2.Show time

3.Increment the time

4.Decrement the time

5.Exit

Enter your choice:5

\*/