**Name - Akriti Choudhary**

**Roll number - 2005776**

**Class - B14**

**Subject - OOP Lab Exam**

**Date - 2/12/2021**

**Question 1)Write a program in C++ that creates a class parcel containing private member variable weight (of the parcel), protected variable shipping\_cost, write a parameterized constructor that initializes these variables. Write a member function with name Cal\_shipping () to calculate the shipping cost according to formula Rs. 10/Kg. Create a class called Box, which is publicly inheriting Parcel class. It should have private member variable no\_of\_boxes which will be initialized using a constructor based on the weight data member of parcel class( if weight exceeds 30 kg then no.of boxes requires 2 and if exceeds 50 kg no.of boxes required 5, otherwise 1 box required). Override class Cal\_shipping () function defined in the class Parcel and add Rs. 30 to the shipping\_cost variable if the volume of the box exceeds 3 units. Display all the information with the final shipping cost.**

**Solution:**

// Akriti Choudhary

// 2005776

#include <iostream>

using namespace std;

class parcel\_776

{

int weight\_776;

protected:

float shipping\_cost\_776;

public:

parcel\_776()

{

weight\_776 = 0;

}

parcel\_776(int we\_776)

{

weight\_776 = we\_776;

}

void calshipping\_776()

{

shipping\_cost\_776 = 10 \* weight\_776;

}

};

class box\_776 : public parcel\_776

{

int no\_of\_boxes\_776;

public:

box\_776(int we\_776) : parcel\_776(we\_776)

{

if ((we\_776 > 30) && (we\_776 < 50))

{

no\_of\_boxes\_776 = 2;

}

else if (we\_776 >= 50)

{

no\_of\_boxes\_776 = 5;

}

else

{

no\_of\_boxes\_776 = 1;

}

}

void calshipping\_776()

{

parcel\_776::calshipping\_776();

if (no\_of\_boxes\_776 > 3)

{

shipping\_cost\_776 += 30;

}

}

void display()

{

cout << "number of boxes : " << no\_of\_boxes\_776 << endl;

cout << "Shipping cost : " << shipping\_cost\_776 << endl;

}

};

int main()

{

int we\_776;

cout << "enter the weight : ";

cin >> we\_776;

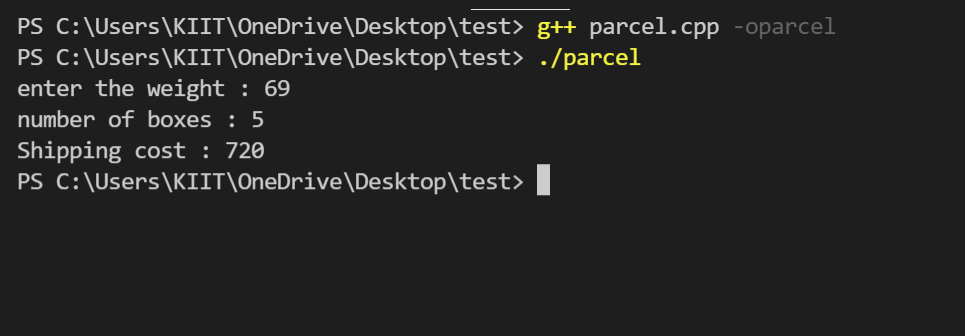
box\_776 Akriti(we\_776);

Akriti.calshipping\_776();

Akriti.display();

return 0;

}



**SET A**

**Question 2)Write a program in C++ that creates a class vect, which contains a pointer to an integer (int \*ptr) and an integer (size).The integer pointer (ptr) will point to a dynamic array of integers and size represents the total capacity of that dynamic array. The class vect should behave as an array with practically unlimited entries.  
  
a) Write a default constructor that will initialize integer pointer (ptr) to NULL and integer variable (size) to zero.  
  
b) Write a parameterized constructor that will initialize integer variable (size) to a value passed as parameter. Initialize the integer pointer (ptr) to a dynamic array of size that equals to the parameter that is passed to constructor.**

**Solution:**

//Akriti Choudhary(2005776)

#include <iostream>

using namespace std;

class vect\_776

{

private:

int size\_776;

int \*ptr\_776;

public:

vect\_776()

{

cout << "Default Constructor is called\n";

size\_776 = 0;

ptr\_776 = NULL;

}

vect\_776(int s\_776)

{

cout << "Parameterized Constructor is called\n";

size\_776 = s\_776;

ptr\_776 = new int(size\_776);

}

void input()

{

cout << "Enter the elements of the array : \n";

for (int i = 0; i < size\_776; ++i)

{

cout << "Enter the " << i + 1 << " element of the array : ";

cin >> ptr\_776[i];

}

}

void display()

{

for (int i = 0; i < size\_776; ++i)

{

cout << ptr\_776[i] << "\t";

}

cout << "\n";

}

~vect\_776()

{

cout << "Destructor is called\n";

free(ptr\_776);

}

};

int main()

{

int s\_776;

cout << "Enter the size of the dynamic array : ";

cin >> s\_776;

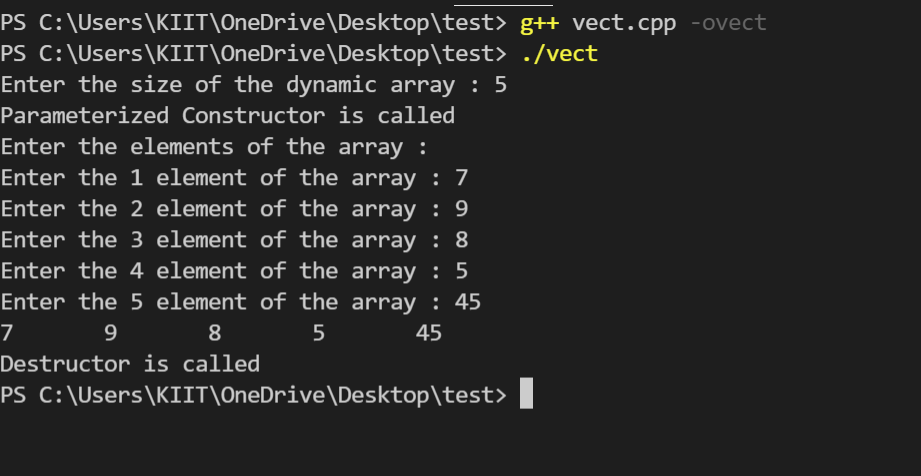
vect\_776 Akriti(s\_776);

Akriti.input();

Akriti.display();

return 0;

}



**Question 3)Write a program to declare three classes a1, a2, a3. The classes have private data member variable of string type. Assign the first name and surname of a student to member variable of class a1 and a2 respectively. Perform concatenation of  two strings using overloading of + operator and store it to data member variable of class a3. Print the full name to the output screen using member function.**

**Solution:**

//Akriti Choudhary(2005776)

#include <iostream>

using namespace std;

class a1\_776

{

private:

string first\_name\_776;

public:

a1\_776()

{

first\_name\_776 = " ";

}

a1\_776(string s\_776)

{

first\_name\_776 = s\_776;

}

string get\_first\_name()

{

return first\_name\_776;

}

};

class a2\_776 : public a1\_776

{

private:

string last\_name\_776;

public:

a2\_776()

{

last\_name\_776 = " ";

}

a2\_776(string l)

{

last\_name\_776 = l;

}

void set\_last\_name(string l)

{

last\_name\_776 = l;

}

string get\_last\_name()

{

return last\_name\_776;

}

friend string operator+(a2\_776 &ob2);

};

class a3\_776 : public a2\_776

{

private:

string name\_776;

string l\_776;

a2\_776 ob2;

public:

a3\_776()

{

cout << "Enter last name : ";

cin >> l\_776;

ob2.set\_last\_name(l\_776);

name\_776 = +ob2;

}

void display()

{

cout << name\_776 << "\n";

}

};

string operator+(a2\_776 &ob2)

{

string s\_776;

cout << "Enter first name : ";

cin >> s\_776;

a1\_776 ob1(s\_776);

string n\_776;

n\_776 = ob1.get\_first\_name() + " " + ob2.last\_name\_776;

return n\_776;

}

int main()

{

a1\_776 obj1\_776;

a2\_776 obj2\_776;

a3\_776 ob3\_776;

ob3\_776.display();

return 0;

}

