Q1.

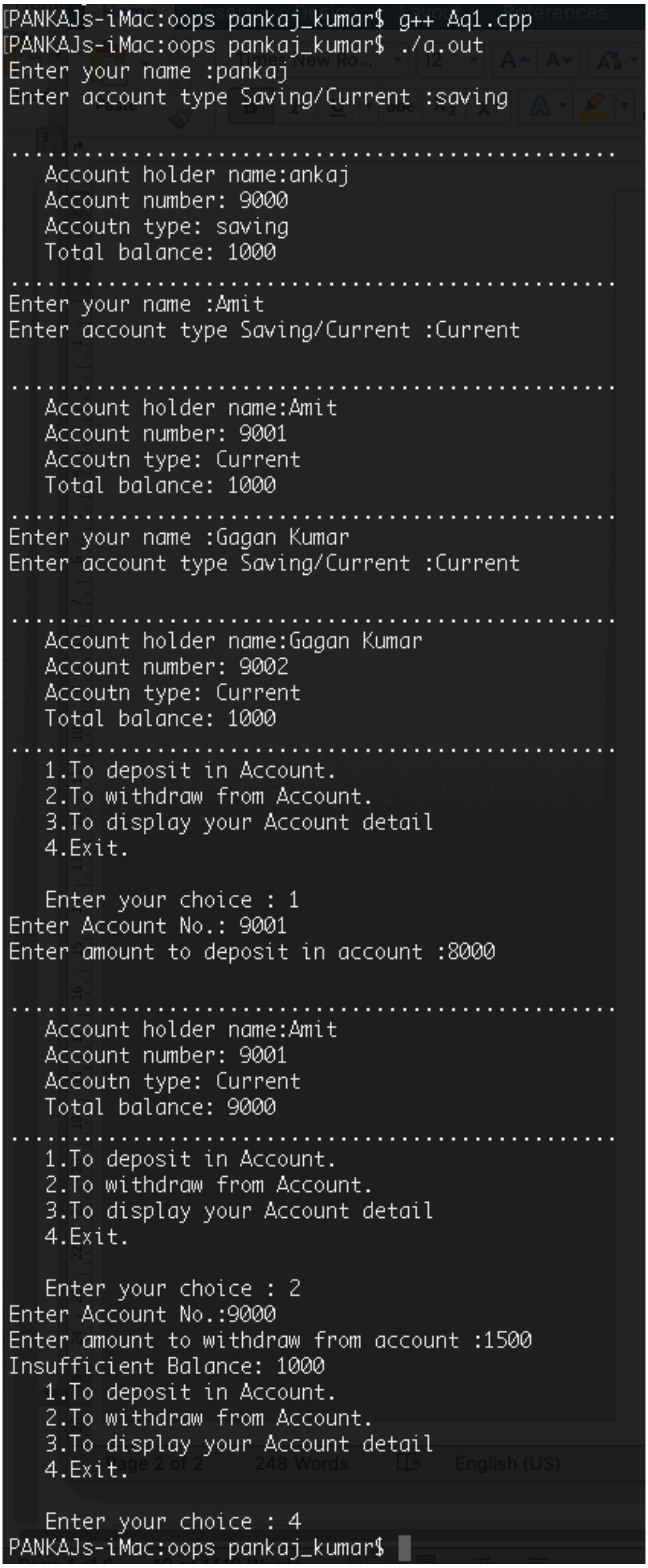
Write a class representing bank accounts. The class should have following data members: Customer Name, Account Number, Type of Account (Savings/Current etc.), Account Balance. The class should allow basic operations like creating a new account, deposit an amount, withdraw money after checking the balance, display account details.

Sol.

```
#include<iostream>

using namespace std;

class Bank_Account{
        char c_name[30];
        int c_acc_no;
        char acc_type[10];
        long balance;
public:
        Bank_Account(){
        balance=1000;
        }
        void create_acc();
        void deposit_amount();
        void withdraw();
        void display();
        int check_acc_no(){
        return c_acc_no;
        }
};
        int acc=9000;
int check_acc_no(int);

int main(){
        int c,i;
        Bank_Account acc[3];
        for(int i=0;i<3;i++){
        acc[i].create_acc();
        }
        do{
        cout<<"   1.To deposit in Account."<<endl;
        cout<<"   2.To withdraw from Account."<<endl;
        cout<<"   3.To display your Account detail"<<endl;
        cout<<"   4.Exit."<<endl;
        cout<<endl<<"   Enter your choice : ";
        cin>>c;
        switch(c){
        case 1:
        cout<<"Enter Account No.: ";
        cin>>c;
        for(i=0;i<3;i++){
        if(acc[i].check_acc_no()==c)
        break;
        }
        acc[i].deposit_amount();
        break;
        case 2:
        cout<<"Enter Account No.:";
        cin>>c;
        for(i=0;i<3;i++){
         if(acc[i].check_acc_no()==c)
         break;
         }
        acc[i].withdraw();
        break;
        case 3:
        cout<<"Enter Account No.: ";
        cin>>c;
        for(i=0;i<3;i++){
         if(acc[i].check_acc_no()==c)
         break;
         }
        acc[i].display();
        break;
```

```
	case 4:
	exit(0);
	}
	}while(true);
	return 0;
}



void Bank_Account::create_acc(){
	cout<<"Enter your name : ";
	cin.ignore();
	cin.getline(c_name,30);
	cout<<"Enter account type Saving/Current :";
	cin>>acc_type;
	c_acc_no=acc;
	acc++;
	display();
}

void Bank_Account::deposit_amount(){
	double amount;
	cout<<"Enter amount to deposit in account :";
	cin>>amount;
	balance+=amount;
	display();
}

void Bank_Account::withdraw(){
	double amount;
	cout<<"Enter amount to withdraw from account :";
	cin>>amount;
	if(amount>balance){
	cout<<"Insufficient Balance: ";
	cout<<balance<<endl;
	return;
	}
	balance-=amount;
	display();
}

void Bank_Account::display(){
	cout<<endl<<"................................................"<<endl;
	cout<<"   Account holder name: "<<c_name<<endl;
	cout<<"   Account number: "<<c_acc_no<<endl;
	cout<<"   Accoutn type: "<<acc_type<<endl;
	cout<<"   Total balance: "<<balance<<endl;
	cout<<"................................................"<<endl;
}
```

OUTPUT :



```
[PANKAJs-iMac:oops pankaj_kumar$ g++ Aq1.cpp
[PANKAJs-iMac:oops pankaj_kumar$ ./a.out
Enter your name :pankaj
Enter account type Saving/Current :saving

..................................................
   Account holder name:ankaj
   Account number: 9000
   Accoutn type: saving
   Total balance: 1000
..................................................
Enter your name :Amit
Enter account type Saving/Current :Current

..................................................
   Account holder name:Amit
   Account number: 9001
   Accoutn type: Current
   Total balance: 1000
..................................................
Enter your name :Gagan Kumar
Enter account type Saving/Current :Current

..................................................
   Account holder name:Gagan Kumar
   Account number: 9002
   Accoutn type: Current
   Total balance: 1000
..................................................
   1.To deposit in Account.
   2.To withdraw from Account.
   3.To display your Account detail
   4.Exit.

   Enter your choice : 1
Enter Account No.: 9001
Enter amount to deposit in account :8000

..................................................
   Account holder name:Amit
   Account number: 9001
   Accoutn type: Current
   Total balance: 9000
..................................................
   1.To deposit in Account.
   2.To withdraw from Account.
   3.To display your Account detail
   4.Exit.

   Enter your choice : 2
Enter Account No.:9000
Enter amount to withdraw from account :1500
Insufficient Balance: 1000
   1.To deposit in Account.
   2.To withdraw from Account.
   3.To display your Account detail
   4.Exit.

   Enter your choice : 4
PANKAJs-iMac:oops pankaj_kumar$ 
```

Q2.

Write an Employee class representing an Employee with an Organization. The class has following data: Employee ID, Employee Name, Date of Birth, Salary of the employee. Write functions to add employee details and display employee details**.**

Sol.

```
#include<iostream>

using namespace std;

class Employee{
        int employee_id;
        char employee_name[30];
        char D_O_B[12];
        double salary;
public:

        void add_employee_detail();
        void display_employee_detail();
};

int id=2100;

int main(){
        Employee emp[3];
        for(int i=0;i<3;i++){
        emp[i].add_employee_detail();
        }
        for(int j=0;j<3;j++){
        emp[j].display_employee_detail();
        }
}

void Employee::display_employee_detail(){
        cout<<endl<<"................................."<<endl;
        cout<<endl<< "  Name of the Employee: "<<employee_name<<endl;
        cout<<"   Employee ID: "<<employee_id<<endl;
        cout<<"   Employee DOB: "<<D_O_B<<endl;
        cout<<"   Salary : "<<salary<<endl;
        cout<<"................................."<<endl;
        }

void Employee::add_employee_detail(){
        cout<<"Name of the Employee: ";
        cin.ignore();
        cin.getline(employee_name,30);
        cout<<"Enter date of birth in the format of dd/mm/yyyy: ";
        cin.ignore();
        cin.getline(D_O_B,12);
         cout<<"Enter the salary of the employee: ";
         cin>>salary;
        employee_id=id;
        id++;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ g++ Aq2.cpp
PANKAJs-iMac:oops pankaj_kumar$ ./a.out
Name of the Employee: Pankaj Kumar
Enter date of birth in the format of dd/mm/yyyy: 05/01/1994
Enter the salary of the employee: 35000
Name of the Employee: Amit
Enter date of birth in the format of dd/mm/yyyy: 05/05/1994
Enter the salary of the employee: 40000
Name of the Employee: Gagan Kumar
Enter date of birth in the format of dd/mm/yyyy: 26/12/1995
Enter the salary of the employee: 38000

.................................

  Name of the Employee: ankaj Kumar
   Employee ID: 2100
   Employee DOB: 5/01/1994
   Salary : 35000
.................................

.................................

  Name of the Employee: Amit
   Employee ID: 2101
   Employee DOB: 5/05/1994
   Salary : 40000
.................................

.................................

  Name of the Employee: Gagan Kumar
   Employee ID: 2102
   Employee DOB: 6/12/1995
   Salary : 38000
.................................
PANKAJs-iMac:oops pankaj_kumar$
```

Q3.

Passing by reference is also an effective way to allow a function to return more than one value. Write a program that returns the previous and next numbers of the first parameter passed using the concept of pass by reference.

Sol.

```
#include <iostream>

using namespace std;

void prevnext (int &x, int& prev, int& next)
       {
       prev = x-1;
       next = x+1;
}

int main (){
       int a, b, c;
       cout<<endl<<"    .............................";
       cout<<endl<<"     Enter a Number: ";
       cin>>a;
       prevnext(a, b, c);
       cout <<endl<<"    Previous=" << b << ",  Next=" << c;
       cout<<endl<<"    .............................."<<endl;
       return 0;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ g++ Aq3.cpp
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

    .............................
     Enter a Number: 99

    Previous=98,  Next=100
    ..............................
PANKAJs-iMac:oops pankaj_kumar$
```

Q4.

Write a program to sort two numbers using call by reference. Smallest number should be output first.

Sol.

```
#include <iostream>

using namespace std;

void sort (int& p, int& n){
       int temp;
       if(p<n)
       temp=0;
       else
       {
       temp=p;
       p=n;
       n=temp;
       }
}

int main ()
{
       int a,b;
       cout<<endl<<"............................";
       cout<<endl<<"     Enter first no.: ";
       cin>>a;
       cout<<endl<<"     Enter second no.: ";
       cin>>b;
       cout<<endl<<"............................";
       sort(a,b);
       cout <<endl<<"     After sorting : " << a <<", "<< b;
       cout<<endl<<"............................"<<endl;
  return 0;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

    .............................
     Enter a Number: 99

    Previous=98,  Next=100
    ..............................
PANKAJs-iMac:oops pankaj_kumar$ g++ Aq4.cpp
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

............................
     Enter first no.: 56

     Enter second no.: 45

............................
     After sorting : 45, 56
............................
```

Q6.

Create two classes KILOS and POUNDS which store the value of weights. KILOS store weight in kilograms and grams and POUNDS in pounds and ounces. Write a program using friend function to add weight of a KILOS object to weight of POUNDS object. Store the result as POUNDS object. (Use 1 pound = 16 ounces, 1 ounce = 28 grams)

Sol.

```
#include<iostream>

using namespace std;

        class KILOS;

        class POUNDS{
                int pounds;
                int ounces;
        public:
                void getdata();
                void display();
        friend POUNDS add ( POUNDS & , KILOS & );
        };

        class KILOS{
                int kg;
                int grams;
        public :
                void getdata();
        friend POUNDS add ( POUNDS & , KILOS &);
        };

        POUNDS add (POUNDS &x ,KILOS &y){
                POUNDS p;
                p.ounces=16*x.pounds + x.ounces;
                p.ounces+= (1000*y.kg + y.grams )/ 28;
                p.pounds = p.ounces /16;
                p.ounces = p.ounces % 16;
                return p;
        }


        int main(){
                KILOS k;
                POUNDS p;
                k.getdata();
                p.getdata();
                POUNDS total;
                total = add(p,k);
                total.display();
        }

        void KILOS::getdata(){
                        cout<<endl<<"........................................";
                        cout<<endl<<"     Enter the values in Kilogram: ";
                        cin>>kg;
                        cout<<endl<<"     Enter the value of grams: ";
                        cin>> grams;
                        cout<<"........................................"<<endl;
        }

        void POUNDS::getdata (){
                        cout<<endl<<"........................................";
                        cout<<endl<<"     Enter the value in pounds: ";
                        cin>>pounds;
                        cout<<endl<<"     Enter the value in ounces: ";
                        cin>>ounces;
                        cout<<"........................................";
                }
          void POUNDS::display(){
                        cout<<endl<<"........................................"<<endl;
                        cout<<"   Total in Pounds: "<<pounds<<" and Ounces: "<<ounces;
                        cout<<endl<<"........................................"<<endl;
                }
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

........................................
     Enter the values in Kilogram: 15

     Enter the value of grams: 900
........................................

........................................
     Enter the value in pounds: 14

     Enter the value in ounces: 12
........................................
........................................
   Total in Pounds: 50 and Ounces: 3
........................................
```

Q7.
Write an overloaded volume function to calculate the volume of a cube (side*side*side), a cuboid (l*b*h) and a cylinder.

Sol.

```
#include<iostream>
#include<stdlib.h>

using namespace std;

                     void volume(float l,float b, float h);
                     void volume(float r,float h);
                     void volume(float l);

       int main(){
              int c;
              float l,b,h,r;
              do{
                     cout<<endl<<"......................................."<<endl;
                     cout<<endl<<"   1.To calculate volume of cube."<<endl;
                     cout<<"   2.To calculate volume of cuboid."<<endl;
                     cout<<"   3.To calculate volume of cylinder."<<endl;
                     cout<<"   4.Exit."<<endl;
                     cout<<endl<<"......................................."<<endl;
                     cout<<"     Enter your choice: ";
                     cin>>c;
                     switch(c){
                            case 1:
                            cout<<"   Enter the side of cube: ";
                            cin>>l;
                            volume(l);
                            break;
                            case 2:
                            cout<<"   Enter the length of cuboid: ";
                            cin>>l;
                            cout<<"   Enter the breath of cuboid: ";
                            cin>>b;
                            cout<<"   Enter the height of cuboid: ";
                            cin>>h;
                            volume(l,b,h);
                            break;
                            case 3:
                            cout<<"   Enter the radius of cylinder: ";
                            cin>>r;
                            cout<<"   Enter the height of cylinder: ";
                            cin>>h;
                            volume(r,h);
```

```
                        break;
                        case 4:
                        exit(0);
                }
        }while(true);
        return 0;
}

void volume(float l,float b,float h){
        float volume;
        volume=l*b*h;
        cout<<endl<<".......................................""<<endl;
        cout<<endl<<"     Volume of cuboid is: "<<volume<<endl;
        cout<<endl<<".......................................""<<endl;
}

void volume(float r,float h){
        float volume;
        volume=(3.14*r*r*h);
        cout<<endl<<".......................................""<<endl;
        cout<<endl<<"      Volume of cylinder is: "<<volume<<endl;
        cout<<endl<<".......................................""<<endl;
}

void volume(float l){
        float volume;
        volume=l*l*l;
        cout<<endl<<".......................................""<<endl;
        cout<<endl<<"       Volume of cube: "<<volume<<endl;
        cout<<endl<<".......................................""<<endl;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

.......................................

   1.To calculate volume of cube.
   2.To calculate volume of cuboid.
   3.To calculate volume of cylinder.
   4.Exit.

.......................................
     Enter your choice: 1
   Enter the side of cube: 7

.......................................

       Volume of cube: 343

.......................................

.......................................

   1.To calculate volume of cube.
   2.To calculate volume of cuboid.
   3.To calculate volume of cylinder.
   4.Exit.

.......................................
     Enter your choice: 2
   Enter the length of cuboid: 4
   Enter the breath of cuboid: 5
   Enter the height of cuboid: 9

.......................................

     Volume of cuboid is: 180

.......................................

.......................................
```

```
   1.To calculate volume of cube.
   2.To calculate volume of cuboid.
   3.To calculate volume of cylinder.
   4.Exit.

.......................................
     Enter your choice: 3
   Enter the radius of cylinder: 4
   Enter the height of cylinder: 6

.......................................

      Volume of cylinder is: 301.44

.......................................

.......................................

   1.To calculate volume of cube.
   2.To calculate volume of cuboid.
   3.To calculate volume of cylinder.
   4.Exit.

.......................................
     Enter your choice: 4
```

Q8.
Write a program to copy data from one int array to another. Use the concept of new and delete to allocate and de-allocate memory for the arrays.

Sol.

```
#include <iostream>

 using namespace std;

int main()
{
    cout << "Enter a positive integer: ";
    int length,value,i;
    cin >> length;
    int *array1 = new int[length];
    int *array2 = new int[length];
    cout << "I just allocated an array of integers of length " << length <<endl<<endl;
    for(i=0;i<length;i++){
    cout<<"  array1["<<i<<"]=";
    cin>>value;
    array1[i]=value;
    array2[i]=array1[i];
    }
    cout<<endl<<"..................."<<endl<<endl;
    for(int j=0;j<length;j++){
    cout<<"  array2["<<j<<"]="<<array2[j]<<endl;
    }
    cout<<endl;
       delete[] array1;
       delete[] array2;
    array1 = 0;
    array2 = 0;

    return 0;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out
Enter a positive integer: 5
I just allocated an array of integers of length 5

  array1[0]=12
  array1[1]=13
  array1[2]=14
  array1[3]=15
  array1[4]=16

...................

  array2[0]=12
  array2[1]=13
  array2[2]=14
  array2[3]=15
  array2[4]=16
```

Q9.

An electricity board charges the following rates to domestic users to discourage large consumption of energy:

```
For the first 100 units    -  60P per unit
For next 200 units         -  80P per unit
Beyond 300 units           -  90P per unit
```

All users are charged a minimum of Rs. 50.00. If the total amount is more than Rs. 300.00 then an additional surcharge of 15% is added.
Write a program to read the names of users and number of units consumed and print out the charges with names.

Sol.

```
#include<iostream>

using namespace std;
	struct ele_board{
		char name[30];
		double unit;
		double charges;
		};
double bill(double);

int main(void){
	struct ele_board record[5];
	for(int i=0;i<5;i++){
	cout<<i+1<<" Enter name: ";
	cin>>record[i].name;
	cout<<i+1<<" Enter unit ";
	cin>>record[i].unit;
	record[i].charges=bill(record[i].unit);
	}
	cout<<"  S.No"<<"  Name     "<<"  Unit  "<<"  Amount Charges   "<<endl;
	for(int j=0;j<5;j++){
	cout<<"   "<<j+1<<"  "<<record[j].name<<"       "<<record[j].unit<<"    "<<record
[j].charges<<endl;
	}
}

double bill(double unit){
	double amount=0;
	for(int i=1;i<=unit;i++){
	if(i<=100)
	{
	amount+=0.6;
	}
	else if(i<=300)
	{
	amount+=0.8;
	}
```

```
       else if(i>300)
       {
       amount+=0.9;
       }
       }
       if(amount<50)
       {
       return 50;
       }
       if (amount>300)
       {
       amount+=(amount*0.15);
       }
       return amount;
}
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out
1 Enter name: Pankaj
1 Enter unit 500
2 Enter name: Gagan
2 Enter unit 400
3 Enter name: Yogi
3 Enter unit 600
4 Enter name: Amit
4 Enter unit 300
5 Enter name: Hitesh
5 Enter unit 80
  S.No  Name       Unit    Amount Charges
   1  Pankaj       500    460
   2  Gagan        400    356.5
   3  Yogi         600    563.5
   4  Amit         300    220
   5  Hitesh       80     50
```

Q10.

Sol.

```
#include<iostream>

using namespace std;

  class DB;

  class DM{
      int metre;
      int cm;
  public:
    void getdata();
    void display();
  friend DM add ( DM & , DB & );
  };

  class DB{
      int feet;
      int inch;
  public :
    void getdata();
  friend DM add ( DM & , DB &);
  };

  DM add (DM &x ,DB &y){
    DM p;
            p.cm  = 100*x.metre + x.cm;
            p.cm+= (30*y.feet)+(3*y.inch);
            p.metre = p.cm /100;
            p.cm = p.cm % 100;
    return p;
  }


int main(){
  DB k;
  DM p;
  k.getdata();
  p.getdata();
  DM total;
  total = add(p,k);
  total.display();
}

void DM::getdata(){
                      cout<<endl<<"........................................";
                      cout<<endl<<"     Enter the values in Metre: ";
                      cin>>metre;
                      cout<<endl<<"     Enter the value of Centimetre: ";
                      cin>> cm;
                      cout<<"........................................"<<endl;
                  }

void DB::getdata (){
                      cout<<endl<<"........................................";
                      cout<<endl<<"     Enter the value in Feet: ";
                      cin>>feet;
                      cout<<endl<<"     Enter the value in Inch: ";
                      cin>>inch;
                      cout<<"........................................";
                    }
void DM::display(){
                      cout<<endl<<"........................................"<<endl;
                      cout<<"   Total in Metre: "<<metre<<" and Centimetre: "<<cm;
                      cout<<endl<<"........................................"<<endl;
                  }
```

OUTPUT :

```
PANKAJs-iMac:oops pankaj_kumar$ ./a.out

........................................
     Enter the value in Feet: 31

     Enter the value in Inch: 9
........................................
........................................
     Enter the values in Metre: 15

     Enter the value of Centimetre: 90
........................................

........................................
   Total in Metre: 25 and Centimetre: 47
...............................
```