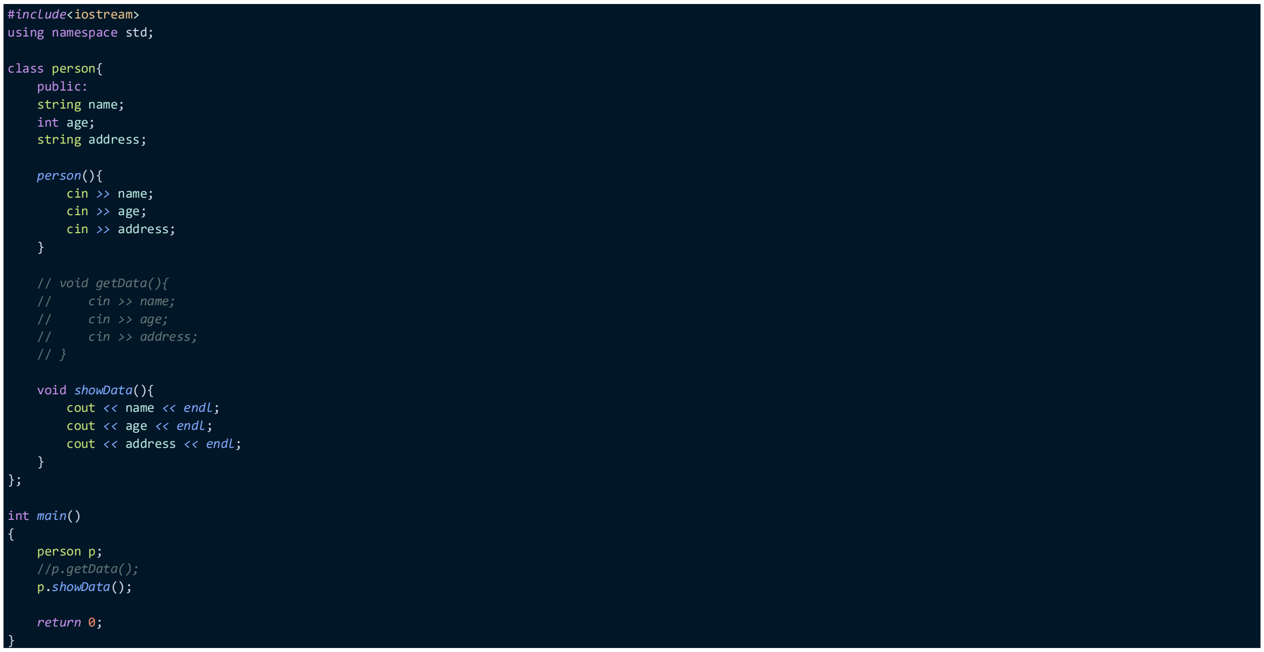
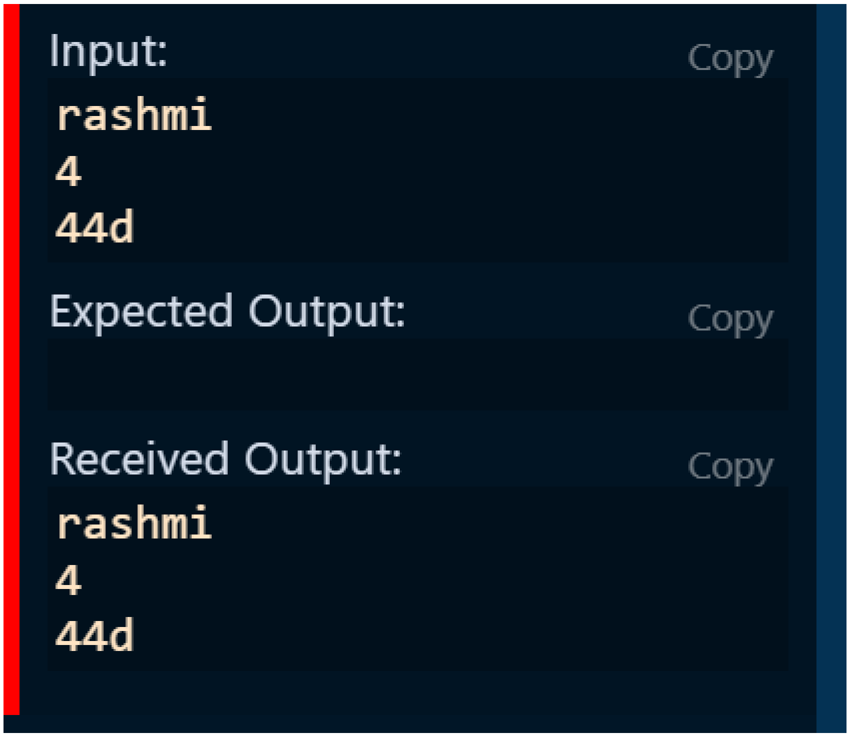
1. Create a class person which has member variables as Name, Age, and Address. There are two member functions in this class showData ( ), getData ( ), though which one can collect the information and show it to user.



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

class person{
    public:
    string name;
    int age;
    string address;

    person(){
        cin >> name;
        cin >> age;
        cin >> address;
    }

    // void getData(){
    //     cin >> name;
    //     cin >> age;
    //     cin >> address;
    // }

    void showData(){
        cout << name << endl;
        cout << age << endl;
        cout << address << endl;
    }
};

int main()
{
    person p;
    //p.getData();
    p.showData();

    return 0;
}
```



Input:

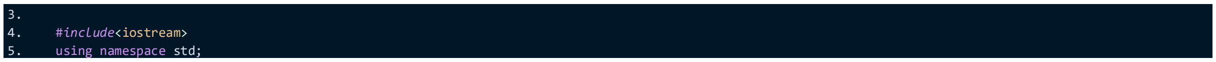
```
rashmi
4
44d
```

Expected Output:

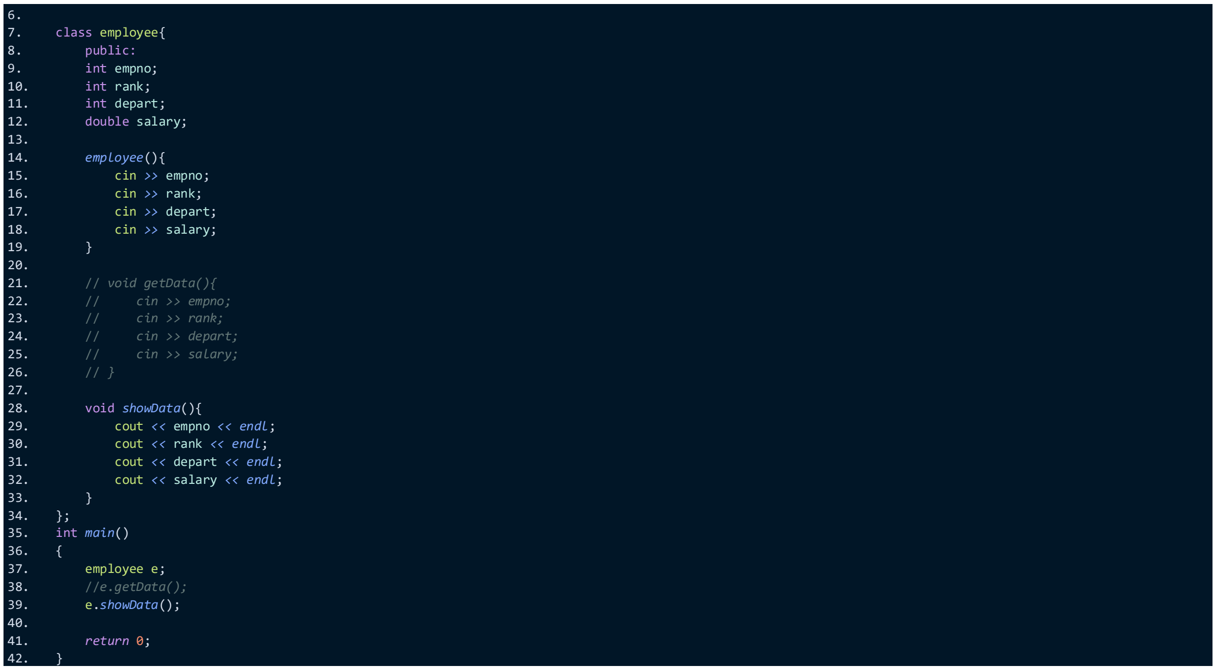
Received Output:

```
rashmi
4
44d
```

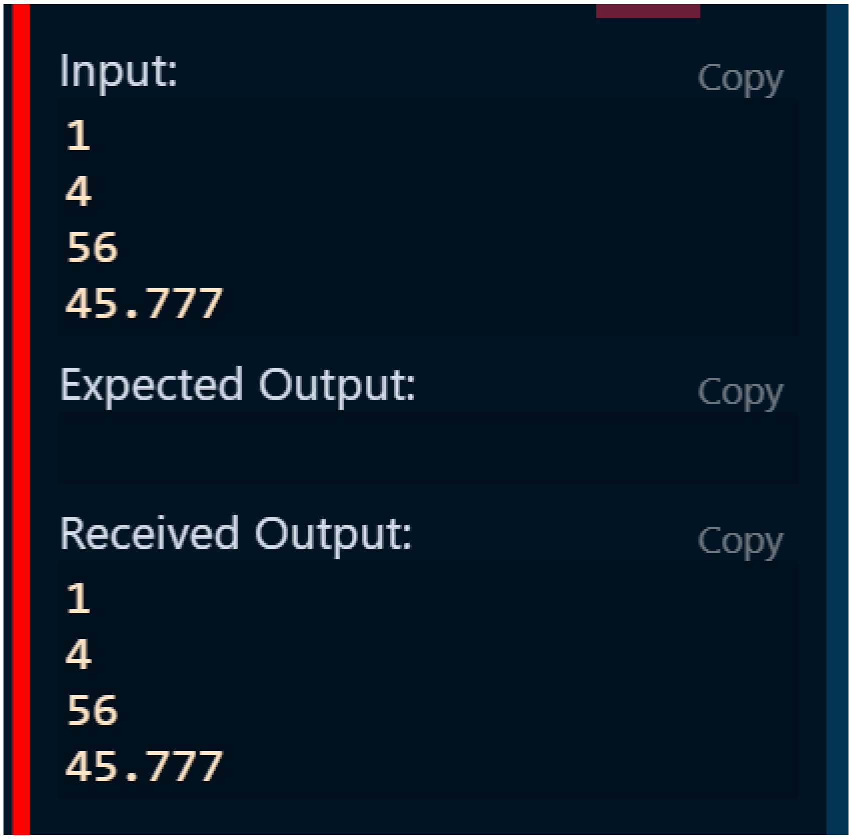
2. Create a class, Employee which has empno, rank, department, salary as member data, and get and show as member functions.



```

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



```cpp

class employee{
    public:
    int empno;
    int rank;
    int depart;
    double salary;

    employee(){
        cin >> empno;
        cin >> rank;
        cin >> depart;
        cin >> salary;
    }

    // void getData(){
    //     cin >> empno;
    //     cin >> rank;
    //     cin >> depart;
    //     cin >> salary;
    // }

    void showData(){
        cout << empno << endl;
        cout << rank << endl;
        cout << depart << endl;
        cout << salary << endl;
    }
};
int main()
{
    employee e;
    //e.getData();
    e.showData();

    return 0;
}
```



Input:

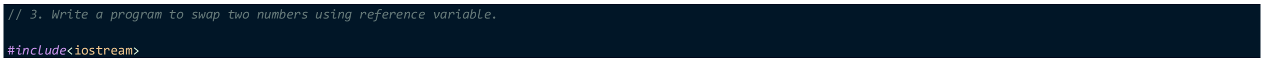
```
1
4
56
45.777
```

Expected Output:

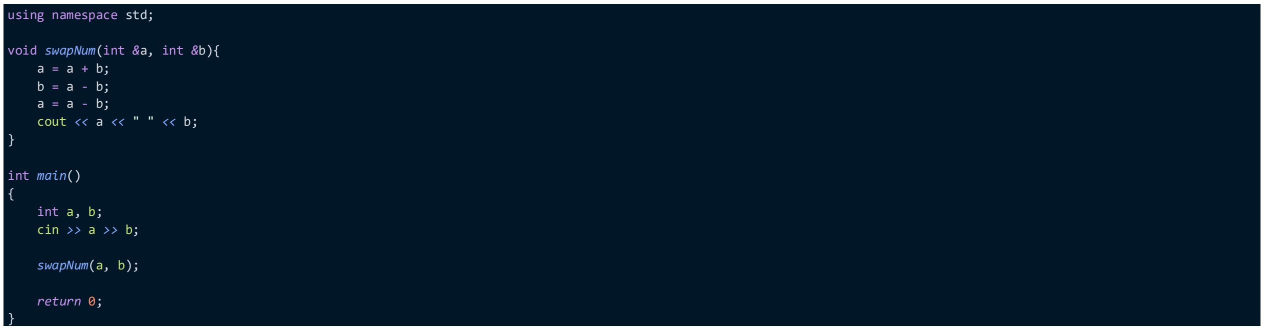
Received Output:

```
1
4
56
45.777
```

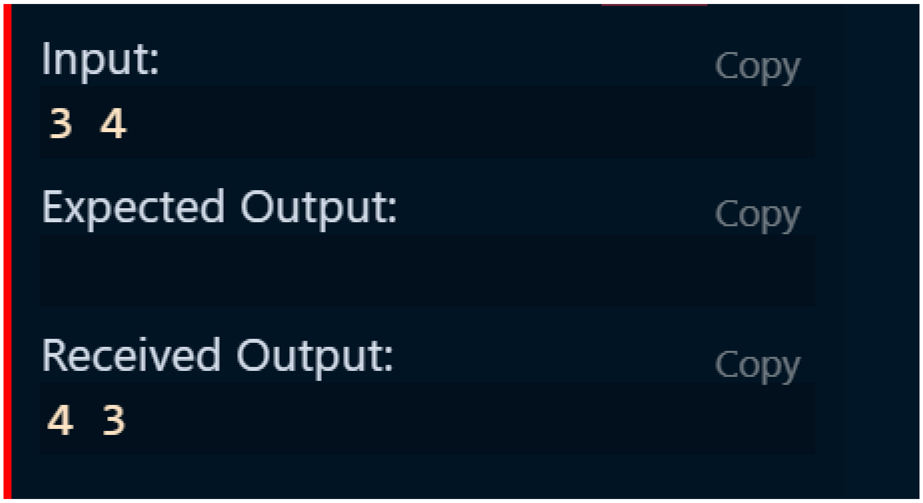
3. Write a program to swap two numbers using reference variable.



```cpp
// 3. Write a program to swap two numbers using reference variable.

#include<iostream>
```



```
using namespace std;

void swapNum(int &a, int &b){
    a = a + b;
    b = a - b;
    a = a - b;
    cout << a << " " << b;
}

int main()
{
    int a, b;
    cin >> a >> b;

    swapNum(a, b);

    return 0;
}
```



Input:

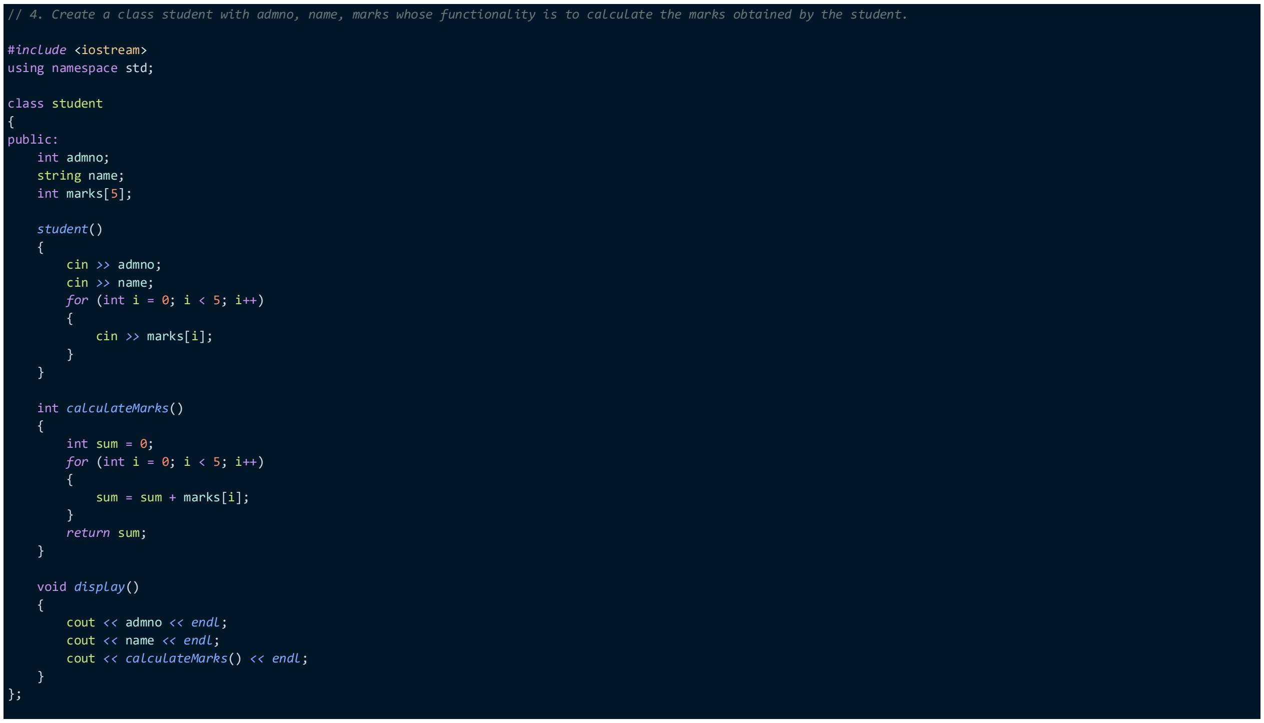
```
3 4
```

Expected Output:

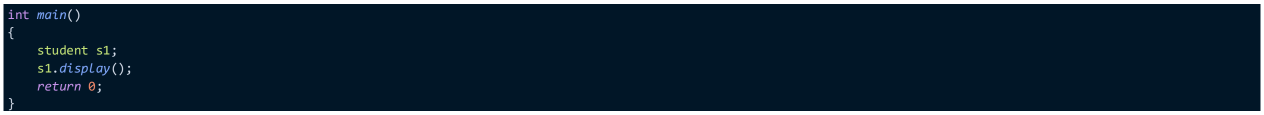
Received Output:

```
4 3
```

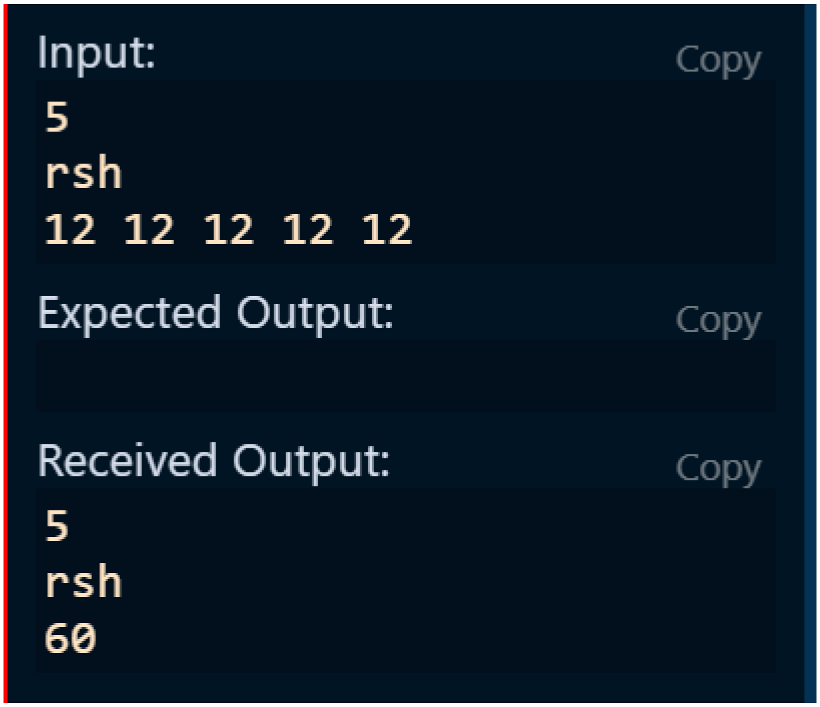
4. Create a class student with admno, name, marks whose functionality is to calculate the marks obtained by the student.



```
// 4. Create a class student with admno, name, marks whose functionality is to calculate the marks obtained by the student.

#include <iostream>
using namespace std;

class student
{
public:
    int admno;
    string name;
    int marks[5];

    student()
    {
        cin >> admno;
        cin >> name;
        for (int i = 0; i < 5; i++)
        {
            cin >> marks[i];
        }
    }

    int calculateMarks()
    {
        int sum = 0;
        for (int i = 0; i < 5; i++)
        {
            sum = sum + marks[i];
        }
        return sum;
    }

    void display()
    {
        cout << admno << endl;
        cout << name << endl;
        cout << calculateMarks() << endl;
    }
};
```



```
int main()
{
    student s1;
    s1.display();
    return 0;
}
```



Input:

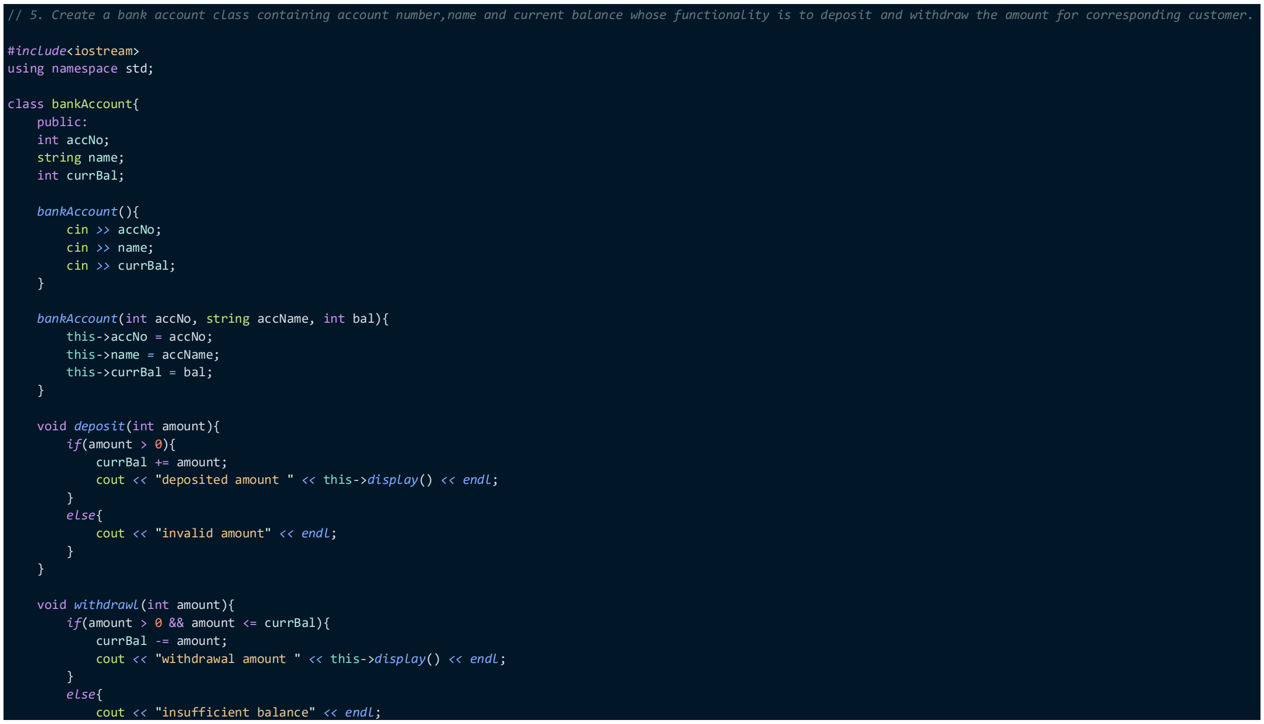
```
5
rsh
12 12 12 12 12
```

Expected Output:

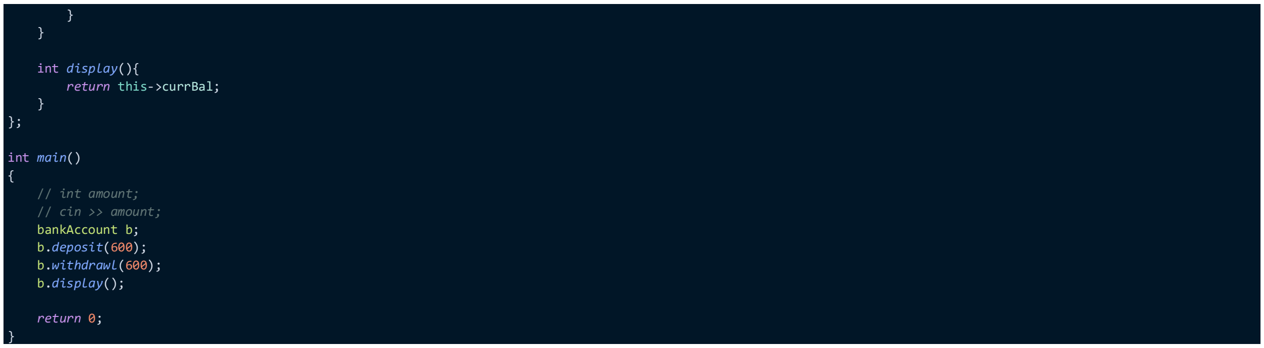
Received Output:

```
5
rsh
60
```

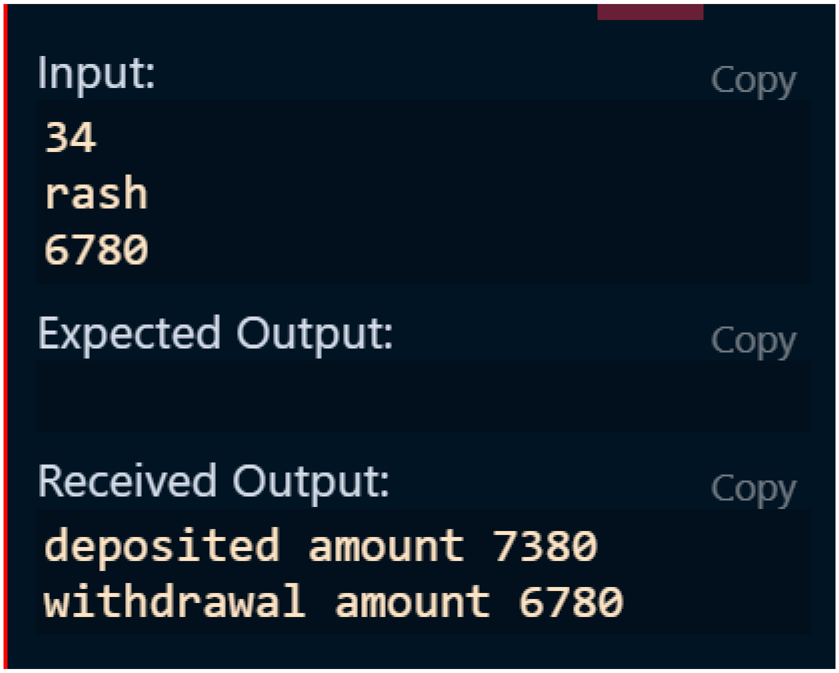
5. Create a bank account class containing account number,name and current balance whose functionality is to deposit and withdraw the amount for corresponding customer.



```
// 5. Create a bank account class containing account number,name and current balance whose functionality is to deposit and withdraw the amount for corresponding customer.

#include<iostream>
using namespace std;

class bankAccount{
    public:
    int accNo;
    string name;
    int currBal;

    bankAccount(){
        cin >> accNo;
        cin >> name;
        cin >> currBal;
    }

    bankAccount(int accNo, string accName, int bal){
        this->accNo = accNo;
        this->name = accName;
        this->currBal = bal;
    }

    void deposit(int amount){
        if(amount > 0){
            currBal += amount;
            cout << "deposited amount " << this->display() << endl;
        }
        else{
            cout << "invalid amount" << endl;
        }
    }

    void withdrawl(int amount){
        if(amount > 0 && amount <= currBal){
            currBal -= amount;
            cout << "withdrawal amount " << this->display() << endl;
        }
        else{
            cout << "insufficient balance" << endl;
```

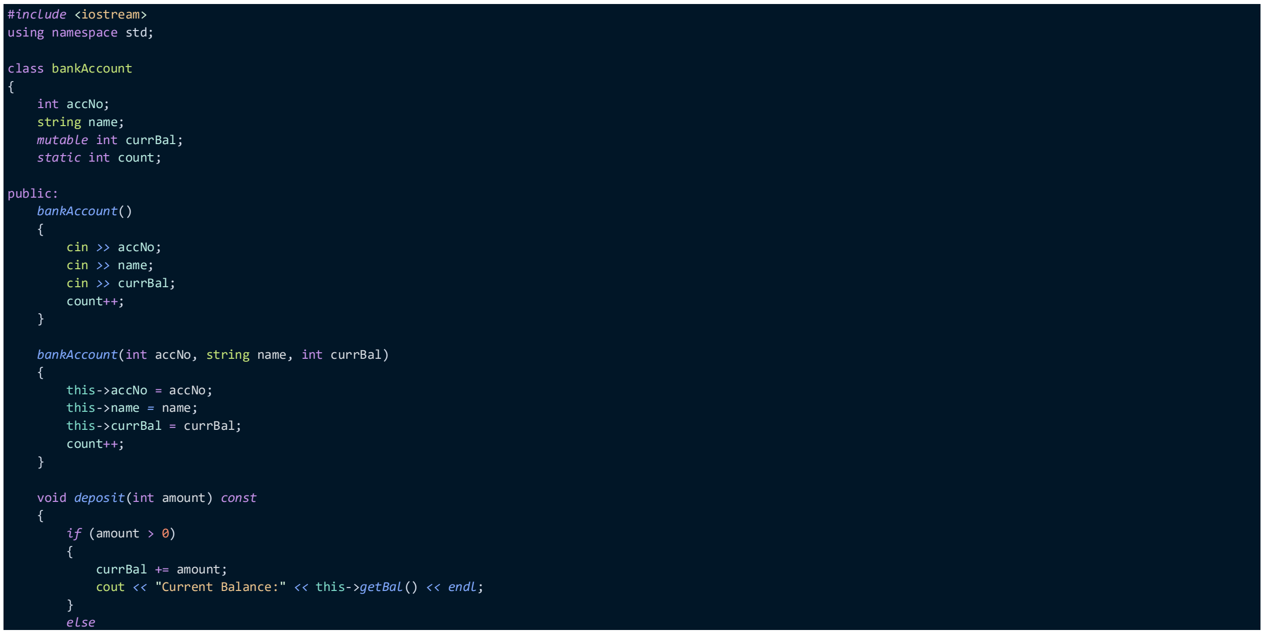


```
        }
    }

    int display(){
        return this->currBal;
    }
};

int main()
{
    // int amount;
    // cin >> amount;
    bankAccount b;
    b.deposit(600);
    b.withdrawl(600);
    b.display();

    return 0;
}
```

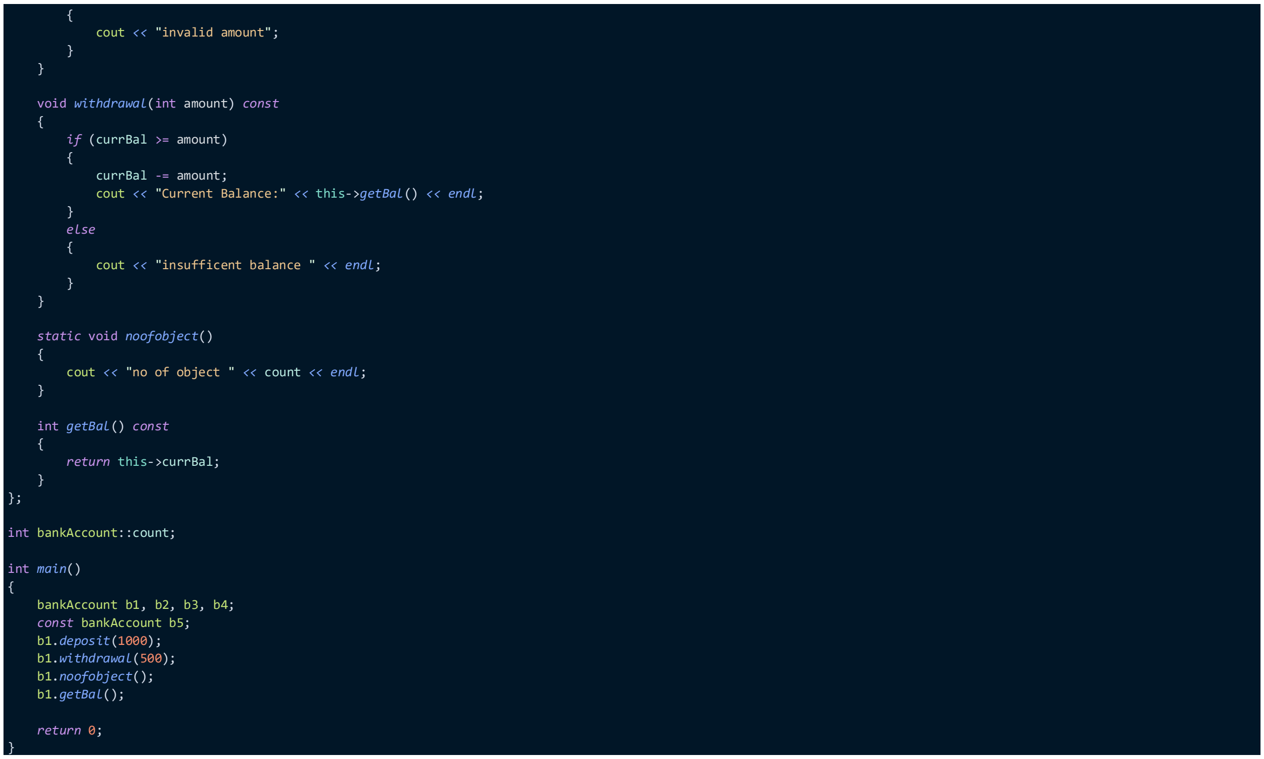


```
34
rash
6780
```

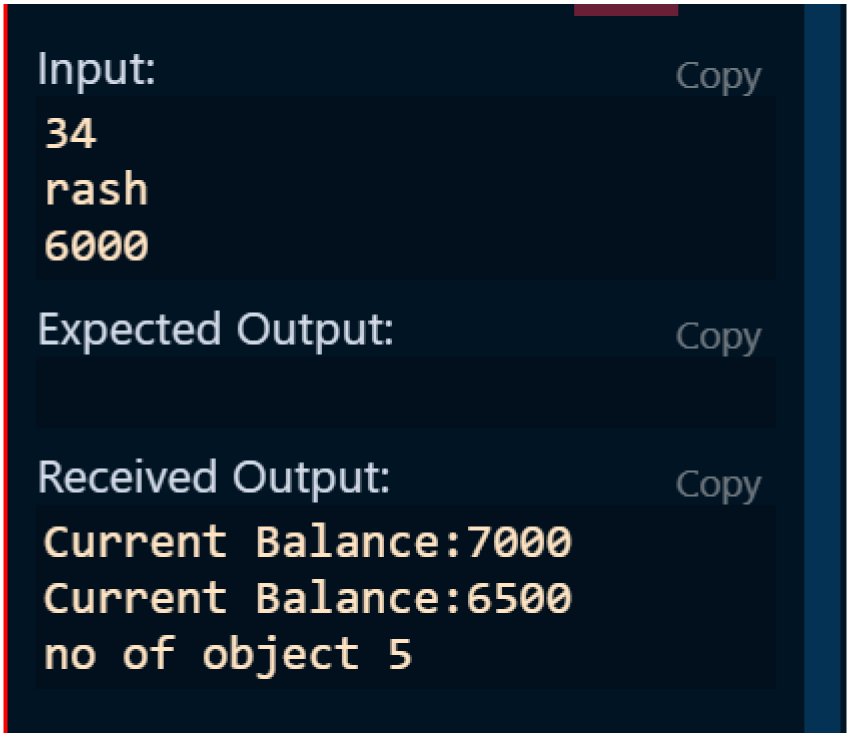
```
deposited amount 7380
withdrawal amount 6780
```



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

class bankAccount
{
    int accNo;
    string name;
    mutable int currBal;
    static int count;

public:
    bankAccount()
    {
        cin >> accNo;
        cin >> name;
        cin >> currBal;
        count++;
    }

    bankAccount(int accNo, string name, int currBal)
    {
        this->accNo = accNo;
        this->name = name;
        this->currBal = currBal;
        count++;
    }

    void deposit(int amount) const
    {
        if (amount > 0)
        {
            currBal += amount;
            cout << "Current Balance:" << this->getBal() << endl;
        }
        else
```



```
        {
            cout << "invalid amount";
        }
    }

    void withdrawal(int amount) const
    {
        if (currBal >= amount)
        {
            currBal -= amount;
            cout << "Current Balance:" << this->getBal() << endl;
        }
        else
        {
            cout << "insufficent balance " << endl;
        }
    }

    static void noofobject()
    {
        cout << "no of object " << count << endl;
    }

    int getBal() const
    {
        return this->currBal;
    }
};

int bankAccount::count;

int main()
{
    bankAccount b1, b2, b3, b4;
    const bankAccount b5;
    b1.deposit(1000);
    b1.withdrawal(500);
    b1.noofobject();
    b1.getBal();

    return 0;
}
```



Input:

```
34
rash
6000
```

Expected Output:

Received Output:

```
Current Balance:7000
Current Balance:6500
no of object 5
```