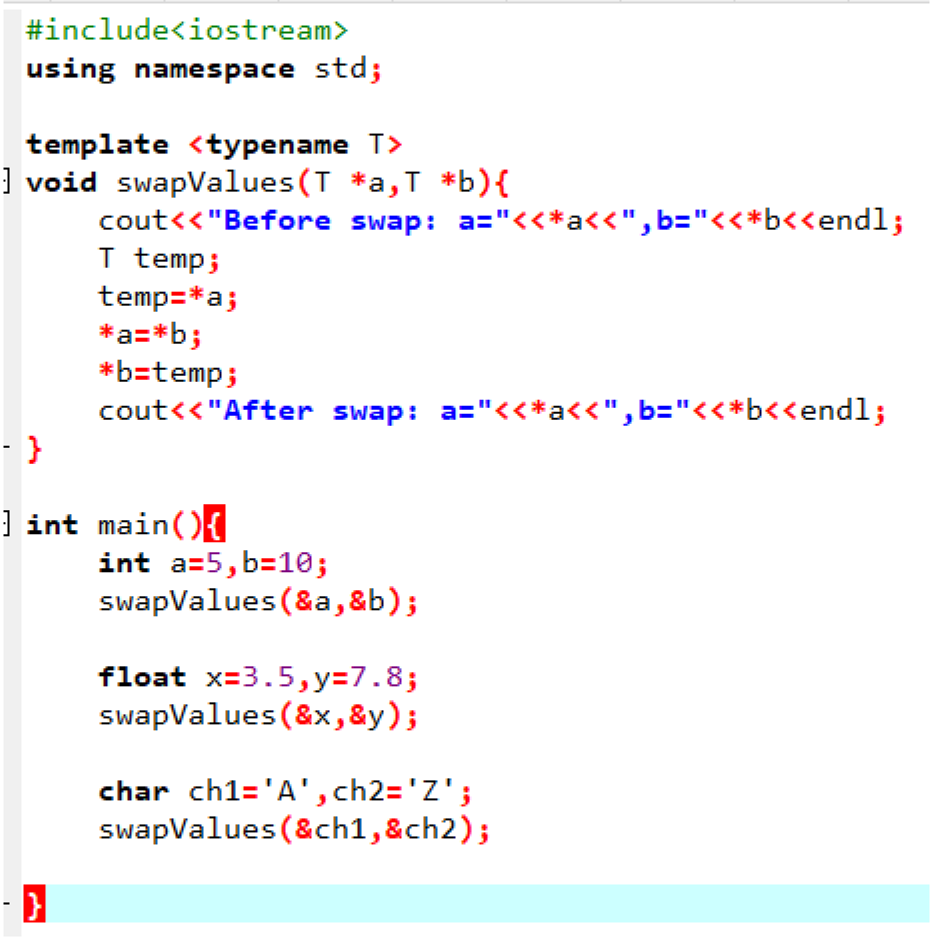
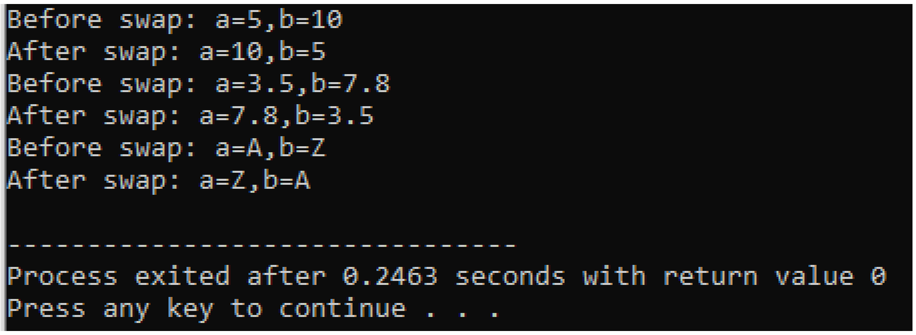
Task 1:

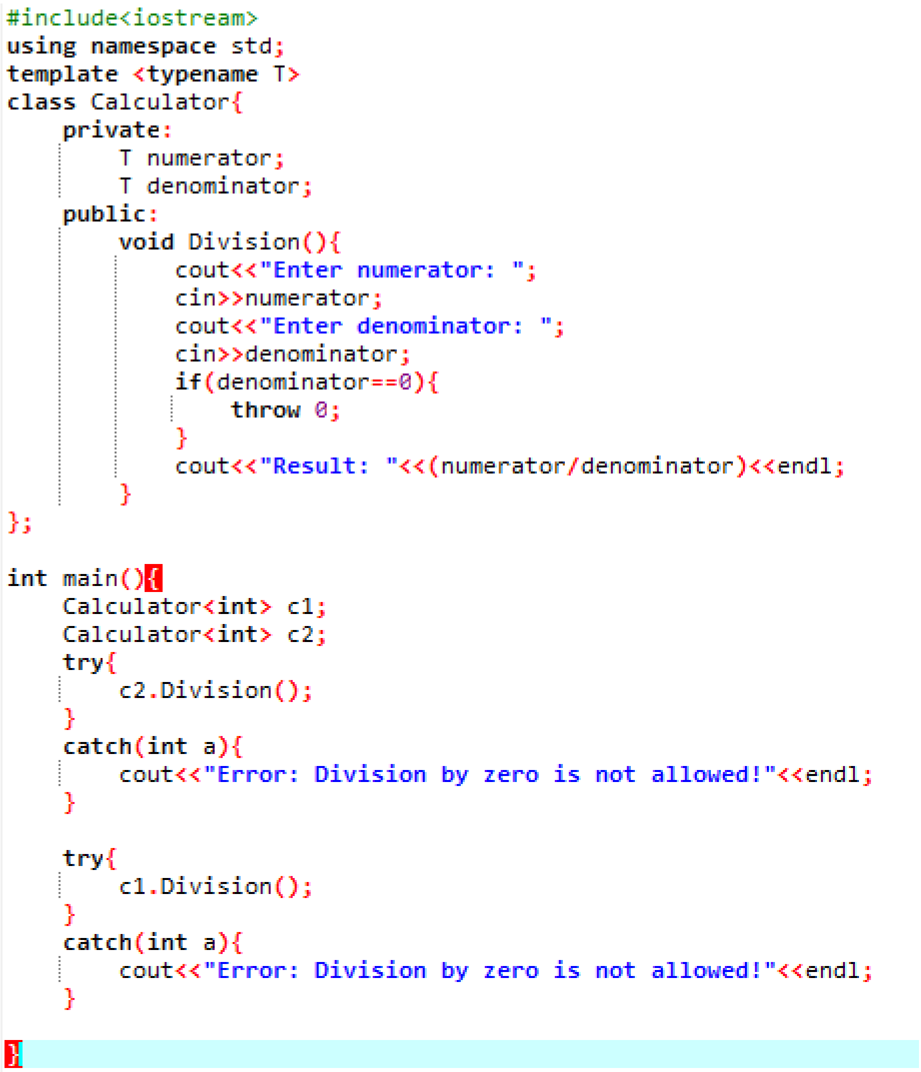


```
#include<iostream>
using namespace std;

template <typename T>
void swapValues(T *a,T *b){
    cout<<"Before swap: a="<<*a<<",b="<<*b<<endl;
    T temp;
    temp=*a;
    *a=*b;
    *b=temp;
    cout<<"After swap: a="<<*a<<",b="<<*b<<endl;
}

int main(){
    int a=5,b=10;
    swapValues(&a,&b);

    float x=3.5,y=7.8;
    swapValues(&x,&y);

    char ch1='A',ch2='Z';
    swapValues(&ch1,&ch2);

}
```

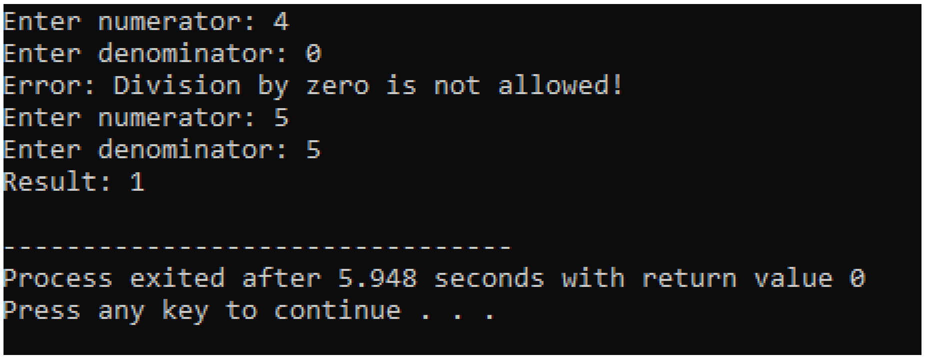


```
Before swap: a=5,b=10
After swap: a=10,b=5
Before swap: a=3.5,b=7.8
After swap: a=7.8,b=3.5
Before swap: a=A,b=Z
After swap: a=Z,b=A

--------------------------------
Process exited after 0.2463 seconds with return value 0
Press any key to continue . . .
```

Task 2:

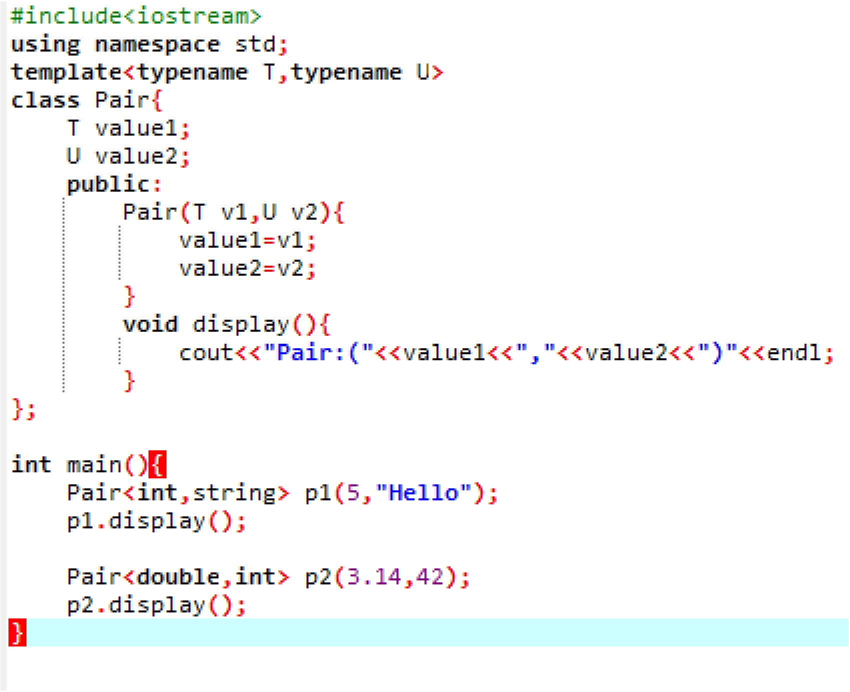


```
#include<iostream>
using namespace std;
template <typename T>
class Calculator{
    private:
        T numerator;
        T denominator;
    public:
        void Division(){
            cout<<"Enter numerator: ";
            cin>>numerator;
            cout<<"Enter denominator: ";
            cin>>denominator;
            if(denominator==0){
                throw 0;
            }
            cout<<"Result: "<<(numerator/denominator)<<endl;
        }
};

int main(){
    Calculator<int> c1;
    Calculator<int> c2;
    try{
        c2.Division();
    }
    catch(int a){
        cout<<"Error: Division by zero is not allowed!"<<endl;
    }

    try{
        c1.Division();
    }
    catch(int a){
        cout<<"Error: Division by zero is not allowed!"<<endl;
    }

}
```

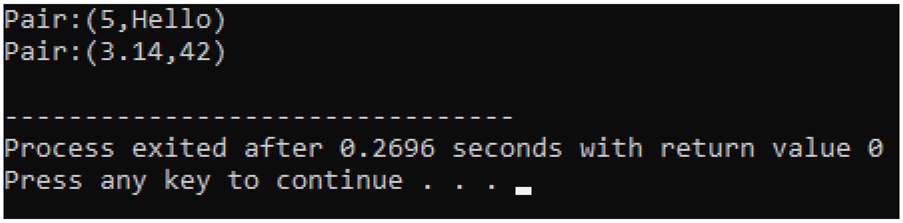


```
Enter numerator: 4
Enter denominator: 0
Error: Division by zero is not allowed!
Enter numerator: 5
Enter denominator: 5
Result: 1

--------------------------------
Process exited after 5.948 seconds with return value 0
Press any key to continue . . .
```

Task 3:

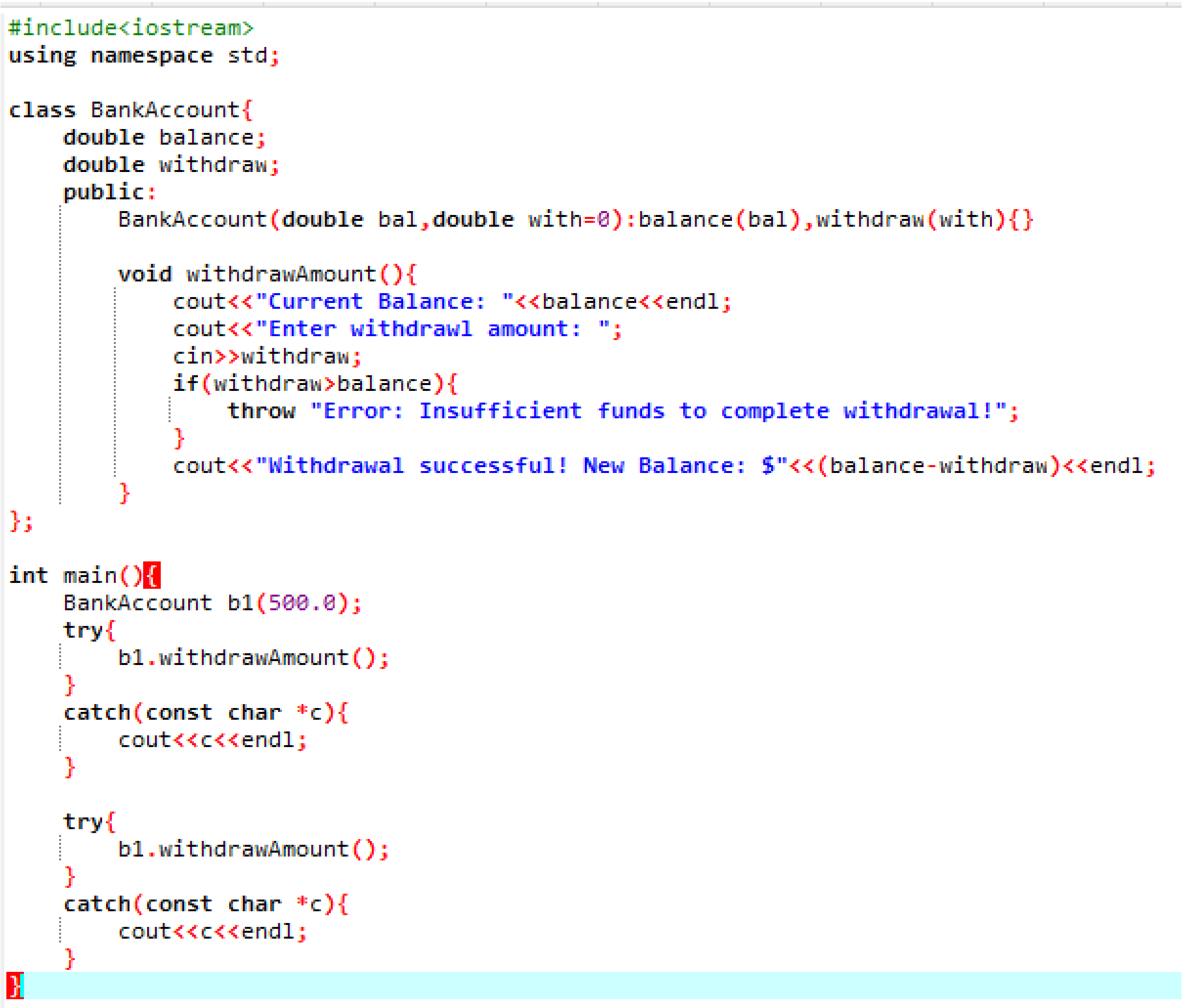


```cpp
#include<iostream>
using namespace std;
template<typename T,typename U>
class Pair{
    T value1;
    U value2;
    public:
        Pair(T v1,U v2){
            value1=v1;
            value2=v2;
        }
        void display(){
            cout<<"Pair:("<<value1<<","<<value2<<")"<<endl;
        }
};

int main(){
    Pair<int,string> p1(5,"Hello");
    p1.display();

    Pair<double,int> p2(3.14,42);
    p2.display();
}
```

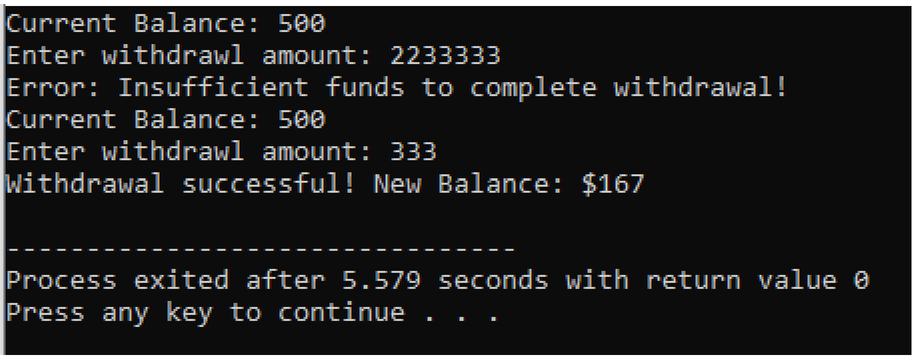


```
Pair:(5,Hello)
Pair:(3.14,42)

--------------------------------
Process exited after 0.2696 seconds with return value 0
Press any key to continue . . .
```

Task 4:

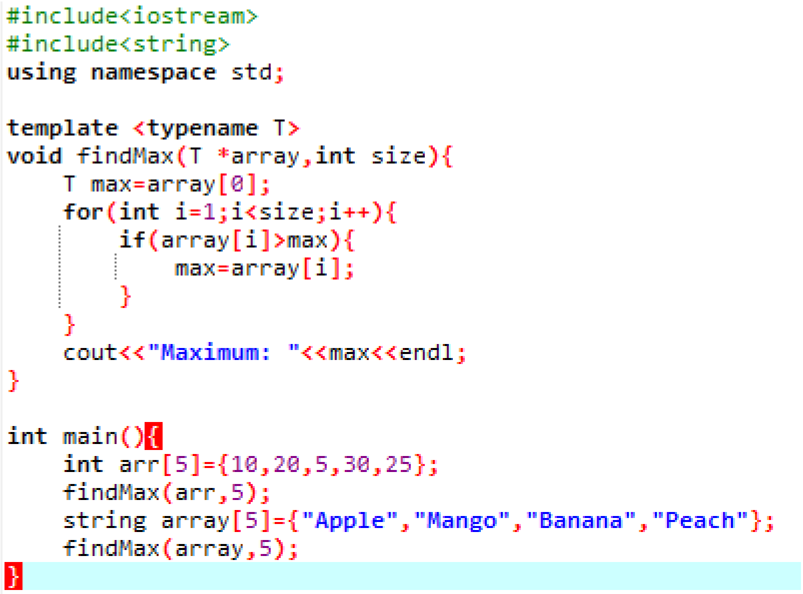


```
#include<iostream>
using namespace std;

class BankAccount{
    double balance;
    double withdraw;
    public:
        BankAccount(double bal,double with=0):balance(bal),withdraw(with){}

        void withdrawAmount(){
            cout<<"Current Balance: "<<balance<<endl;
            cout<<"Enter withdrawl amount: ";
            cin>>withdraw;
            if(withdraw>balance){
                throw "Error: Insufficient funds to complete withdrawal!";
            }
            cout<<"Withdrawal successful! New Balance: $"<<(balance-withdraw)<<endl;
        }
};

int main(){
    BankAccount b1(500.0);
    try{
        b1.withdrawAmount();
    }
    catch(const char *c){
        cout<<c<<endl;
    }

    try{
        b1.withdrawAmount();
    }
    catch(const char *c){
        cout<<c<<endl;
    }
}
```

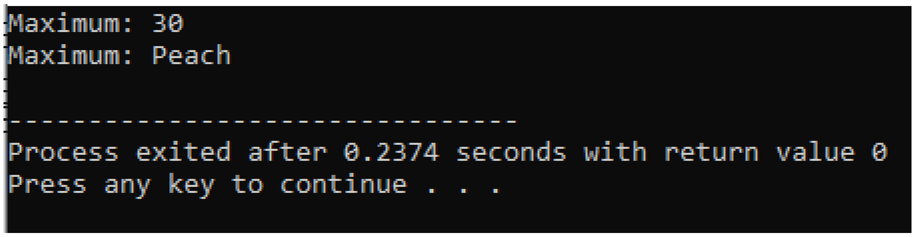


```
Current Balance: 500
Enter withdrawl amount: 2233333
Error: Insufficient funds to complete withdrawal!
Current Balance: 500
Enter withdrawl amount: 333
Withdrawal successful! New Balance: $167


--------------------------------
Process exited after 5.579 seconds with return value 0
Press any key to continue . . .
```

Task 5:

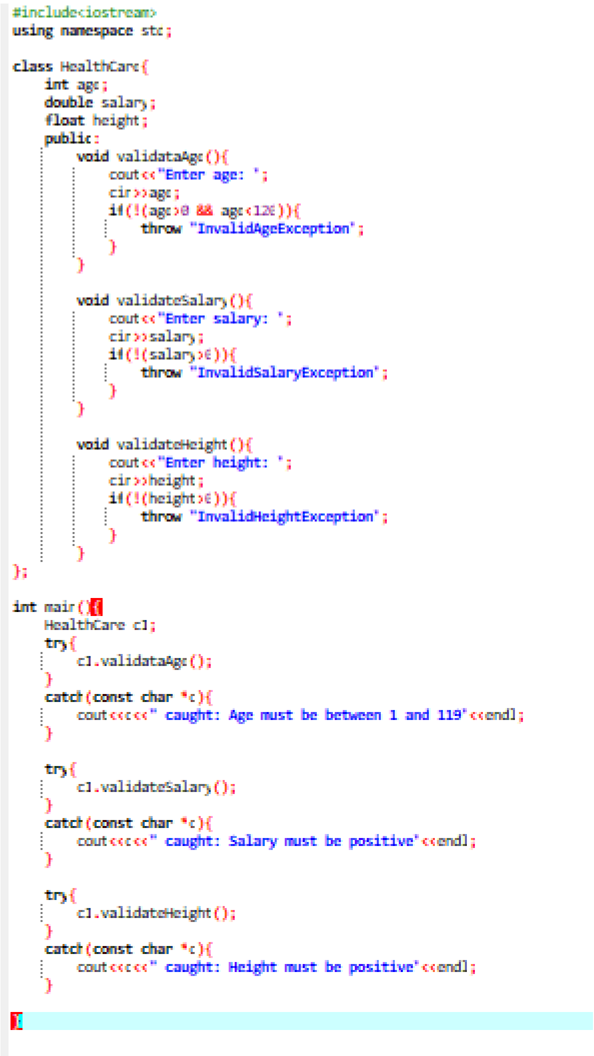


```
#include<iostream>
#include<string>
using namespace std;

template <typename T>
void findMax(T *array,int size){
    T max=array[0];
    for(int i=1;i<size;i++){
        if(array[i]>max){
            max=array[i];
        }
    }
    cout<<"Maximum: "<<max<<endl;
}

int main(){
    int arr[5]={10,20,5,30,25};
    findMax(arr,5);
    string array[5]={"Apple","Mango","Banana","Peach"};
    findMax(array,5);
}
```

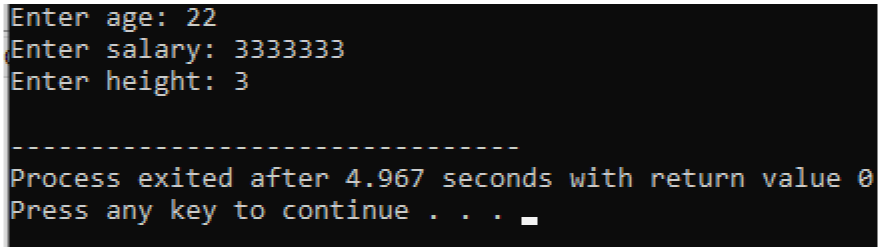


```
Maximum: 30
Maximum: Peach

--------------------------------
Process exited after 0.2374 seconds with return value 0
Press any key to continue . . .
```

Task 6:

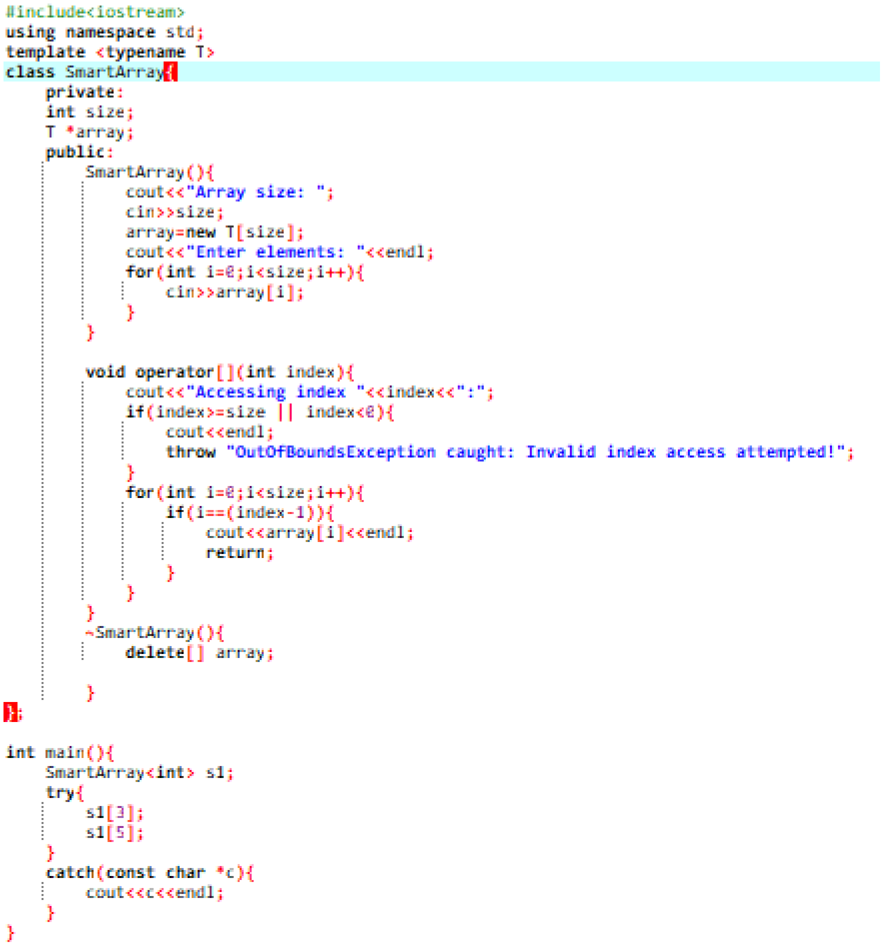


```
#include<iostream>
using namespace std;

class HealthCare{
    int age;
    double salary;
    float height;
    public:
        void validataAge(){
            cout<<"Enter age: ";
            cin>>age;
            if(!(age>0 && age<120)){
                throw "InvalidAgeException";
            }
        }

        void validateSalary(){
            cout<<"Enter salary: ";
            cin>>salary;
            if(!(salary>0)){
                throw "InvalidSalaryException";
            }
        }

        void validateHeight(){
            cout<<"Enter height: ";
            cin>>height;
            if(!(height>0)){
                throw "InvalidHeightException";
            }
        }
};

int main(){
    HealthCare c1;
    try{
        c1.validataAge();
    }
    catch(const char *c){
        cout<<c<<" caught: Age must be between 1 and 119"<<endl;
    }

    try{
        c1.validateSalary();
    }
    catch(const char *c){
        cout<<c<<" caught: Salary must be positive"<<endl;
    }

    try{
        c1.validateHeight();
    }
    catch(const char *c){
        cout<<c<<" caught: Height must be positive"<<endl;
    }

}
```

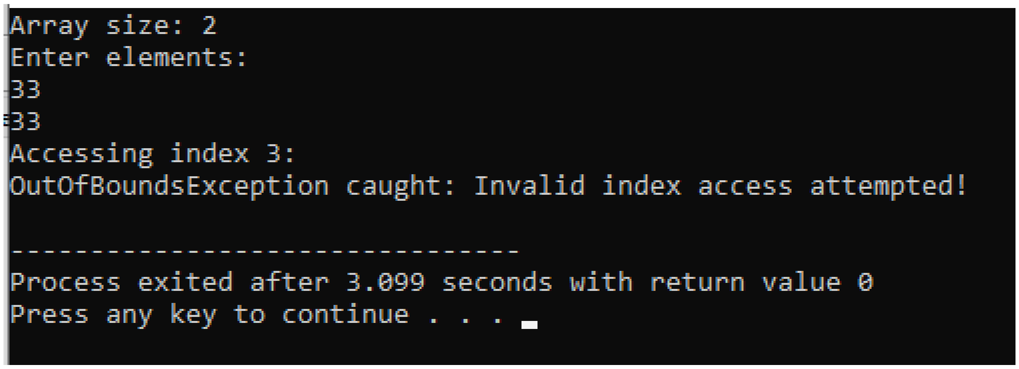


```
Enter age: 22
Enter salary: 3333333
Enter height: 3

--------------------------------
Process exited after 4.967 seconds with return value 0
Press any key to continue . . .
```

Task 7:



```
#include<iostream>
using namespace std;
template <typename T>
class SmartArray{
    private:
    int size;
    T *array;
    public:
        SmartArray(){
            cout<<"Array size: ";
            cin>>size;
            array=new T[size];
            cout<<"Enter elements: "<<endl;
            for(int i=0;i<size;i++){
                cin>>array[i];
            }
        }

        void operator[](int index){
            cout<<"Accessing index "<<index<<":";
            if(index>=size || index<0){
                cout<<endl;
                throw "OutOfBoundsException caught: Invalid index access attempted!";
            }
            for(int i=0;i<size;i++){
                if(i==(index-1)){
                    cout<<array[i]<<endl;
                    return;
                }
            }
        }
        ~SmartArray(){
            delete[] array;

        }
};

int main(){
    SmartArray<int> s1;
    try{
        s1[3];
        s1[5];
    }
    catch(const char *c){
        cout<<c<<endl;
    }
}
```



```
Array size: 2
Enter elements:
33
33
Accessing index 3:
OutOfBoundsException caught: Invalid index access attempted!

--------------------------------
Process exited after 3.099 seconds with return value 0
Press any key to continue . . .
```