Name: Brijesh Rameshbhai Rohit

Admission number: U19CS009

PRINCIPLES OF PROGRAMMING LANGUAGES

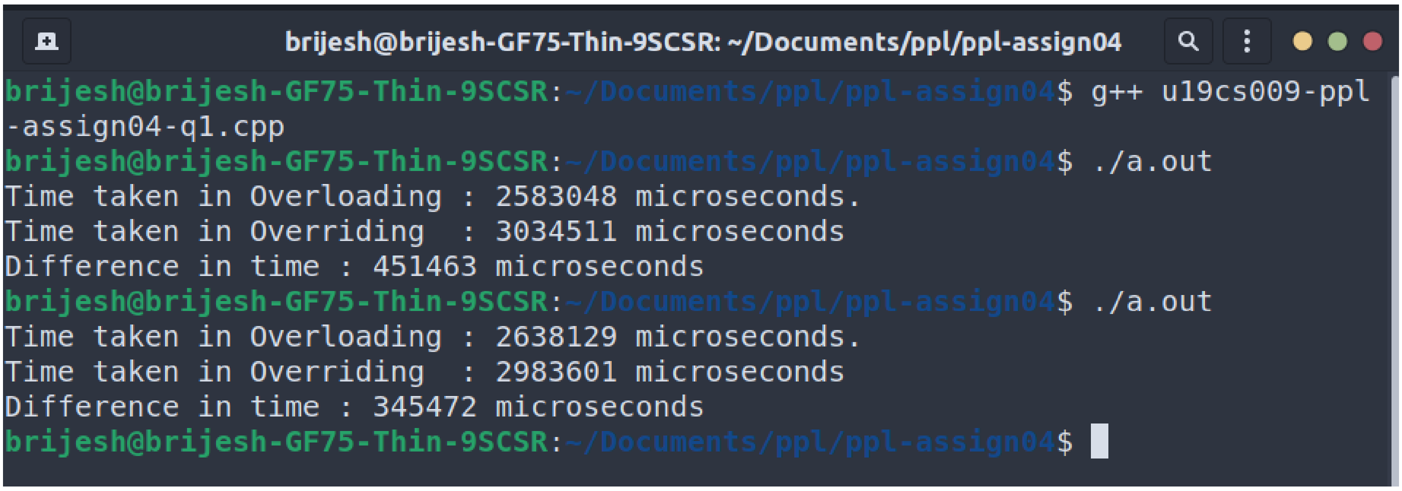
ASSIGNMENT 4

1.Write a program in C++ that calls both a dynamically bound method and a statically bound method a large number of times, timing the calls to both of the two. Compare the timing results and compute the difference of the time required by the two. Explain the results.

CODE=>

```
//U19CS009
//Brijesh Rohit

#include<iostream>
#include<chrono>
#include<unistd.h>
using namespace std::chrono;
using namespace std;

class Base {
public:
    int sum(int x) {
        return x + 2;
    }
    virtual int sum(int x, int y) {
        int a = x;
        int b = y;
        int result = a + b;
        return result;
    }
};

class Derived : public Base {
public:
    int sum(int x, int y) override {
        int a = x;
        int b = y;
        int result = a + b;
        return result;
    }
};

int main(int argc, char const *argv[])
{
    Base B;

    //overloading
    auto start = high_resolution_clock::now();
    for (int i = 0; i < 1000000000; i++)
        int ans = B.sum(2, 4);
    auto end = high_resolution_clock::now();

    auto duration = duration_cast<microseconds>(end - start);

    cout << "Time taken in Overloading : " << duration.count() << "
microseconds." <<  endl;;
    auto time1 = duration.count();
    sleep(2);

    Derived d;
    Base *bsptr = &d;

    //overrinding
    start = high_resolution_clock::now();
    for (int i = 0; i < 1000000000; i++) {
        int ans = bsptr->sum(2, 4);
    }
    end = high_resolution_clock::now();

    duration = duration_cast<microseconds>(end - start);

    cout << "Time taken in Overriding  : " << duration.count() << "
microseconds" << endl;
    auto time2 = duration.count();

    cout << "Difference in time : " << time2 - time1 << " microseconds" << endl;
    return 0;
}
```

OUTPUT=>



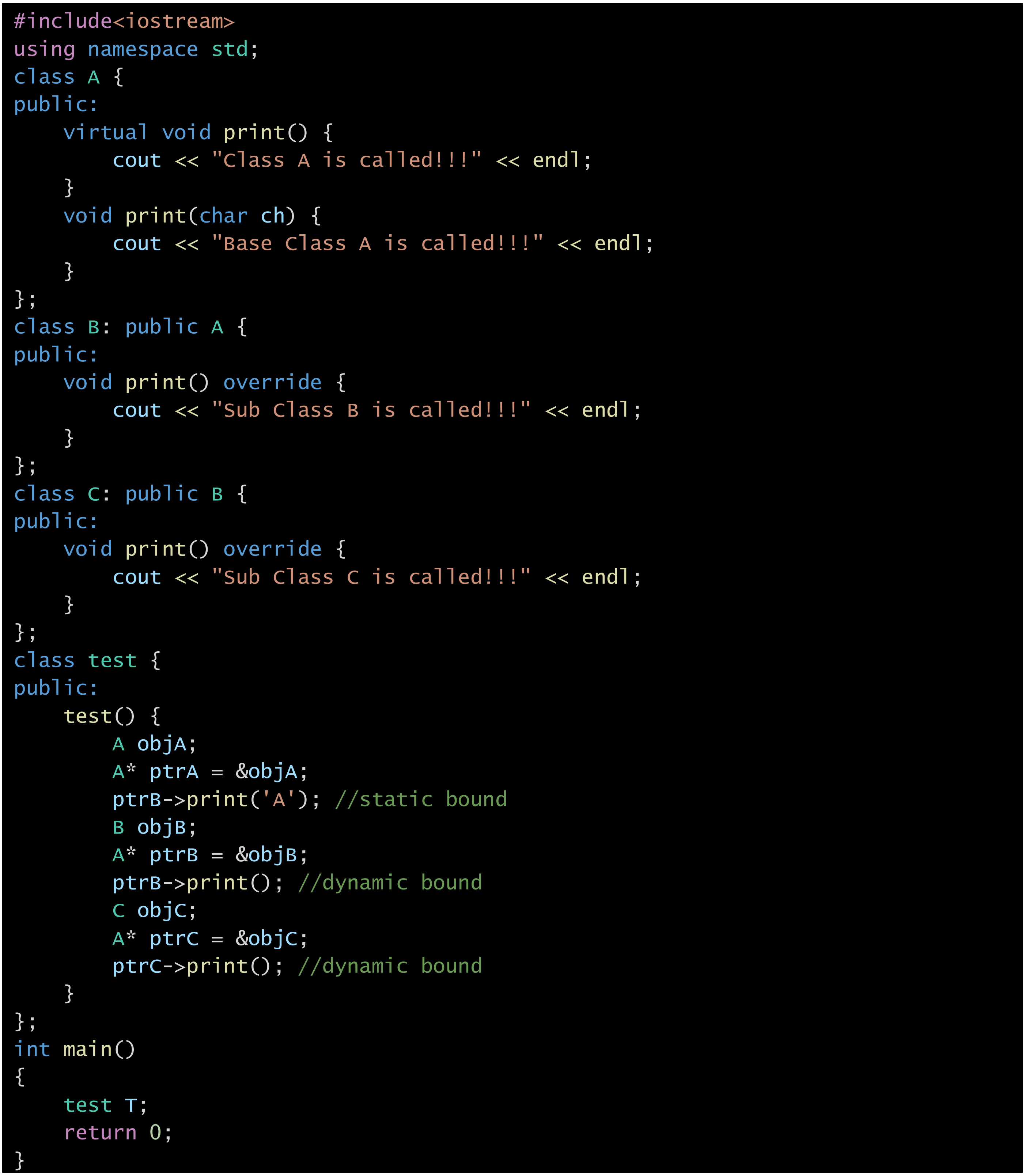
brijesh@brijesh-GF75-Thin-9SCSR: ~/Documents/ppl/ppl-assign04

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ g++ u19cs009-ppl
-assign04-q1.cpp
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Time taken in Overloading : 2583048 microseconds.
Time taken in Overriding  : 3034511 microseconds
Difference in time : 451463 microseconds
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Time taken in Overloading : 2638129 microseconds.
Time taken in Overriding  : 2983601 microseconds
Difference in time : 345472 microseconds
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ 
```

Explanation : *Overloading is faster than Overriding because overloading takes place during the compile time whereas overriding takes place during runtime of the program execution.*

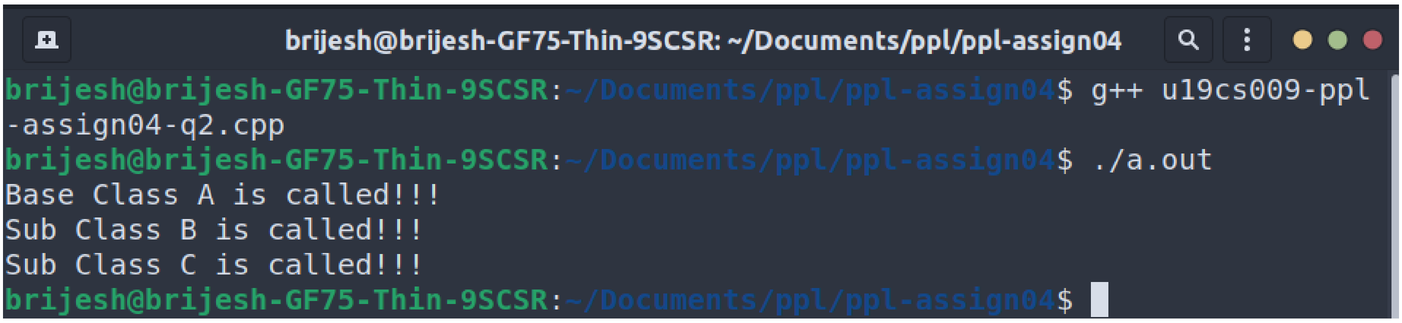
2. Design and implement a C++ program that defines a base class A, which has a subclass B, which itself has a subclass C. The A class must implement a method, which is overridden in both B and C. You must also write a test class that instantiates A, B, and C and includes three calls to the method. One of the calls must be statically bound to A's method. One call must be dynamically bound to B's method, and one must be dynamically bound to C's method. All of the method calls must be through a pointer to class A.

CODE=>



```
#include<iostream>
using namespace std;
class A {
public:
    virtual void print() {
        cout << "Class A is called!!!" << endl;
    }
    void print(char ch) {
        cout << "Base Class A is called!!!" << endl;
    }
};
class B: public A {
public:
    void print() override {
        cout << "Sub Class B is called!!!" << endl;
    }
};
class C: public B {
public:
    void print() override {
        cout << "Sub Class C is called!!!" << endl;
    }
};
class test {
public:
    test() {
        A objA;
        A* ptrA = &objA;
        ptrB->print('A'); //static bound
        B objB;
        A* ptrB = &objB;
        ptrB->print(); //dynamic bound
        C objC;
        A* ptrC = &objC;
        ptrC->print(); //dynamic bound
    }
};
int main()
{
    test T;
    return 0;
}
```

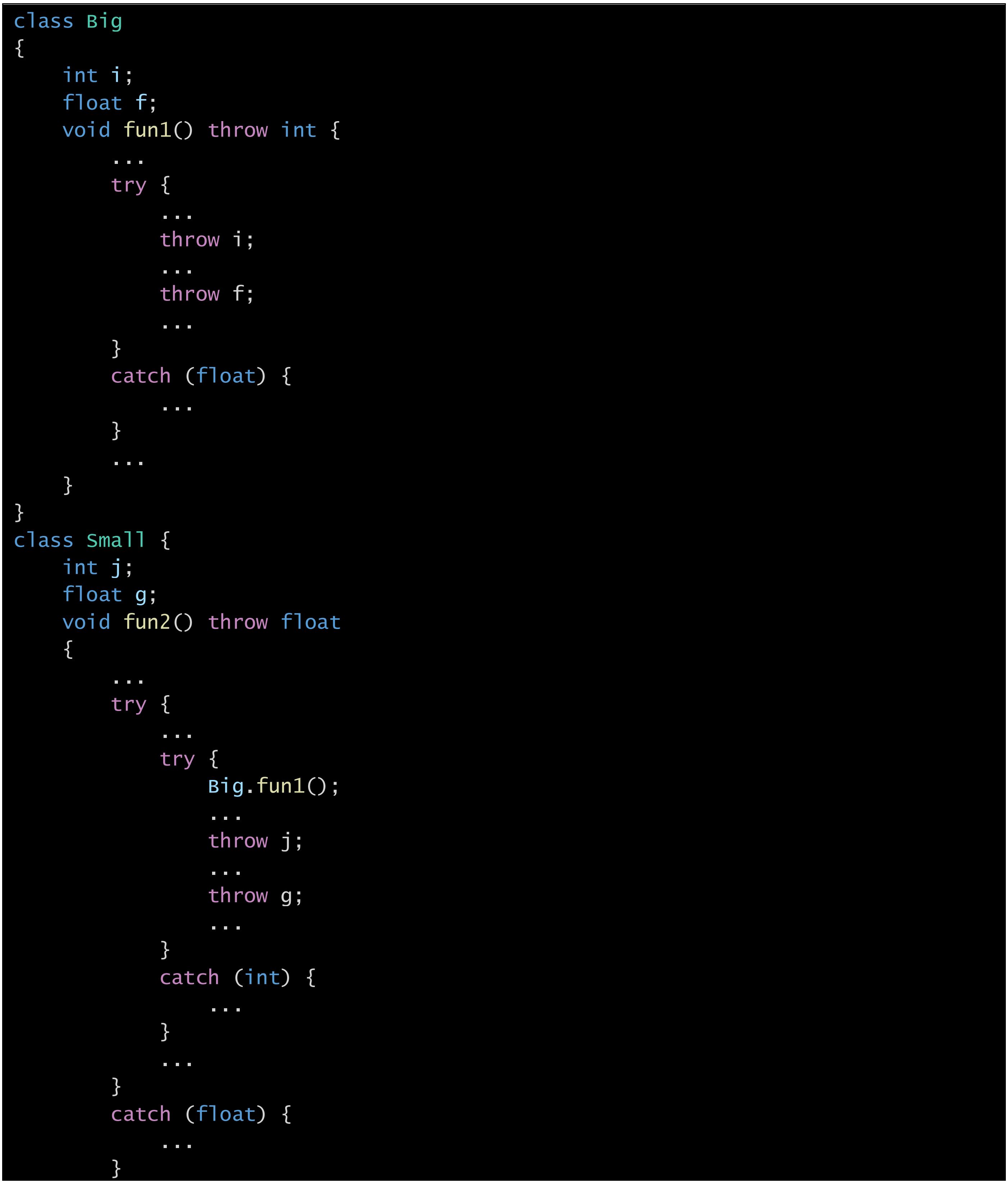
OUTPUT=>



brijesh@brijesh-GF75-Thin-9SCSR: ~/Documents/ppl/ppl-assign04

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ g++ u19cs009-ppl
-assign04-q2.cpp
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Base Class A is called!!!
Sub Class B is called!!!
Sub Class C is called!!!
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ 
```

3. Consider the following C++ skeletal program:



```
class Big
{
    int i;
    float f;
    void fun1() throw int {
        ...
        try {
            ...
            throw i;
            ...
            throw f;
            ...
        }
        catch (float) {
            ...
        }
        ...
    }
}
class Small {
    int j;
    float g;
    void fun2() throw float
    {
        ...
        try {
            ...
            try {
                Big.fun1();
                ...
                throw j;
                ...
                throw g;
                ...
            }
            catch (int) {
                ...
            }
            ...
        }
        catch (float) {
            ...
        }
    }
}
```

In each of the four throw statements, where is the exception handled?

Note that fun1 is called from fun2 in class Small.

Solution:-

Throw i is caught by the catch (int) of the inner try of fun2.

Throw f is caught by the catch (float) of fun1.

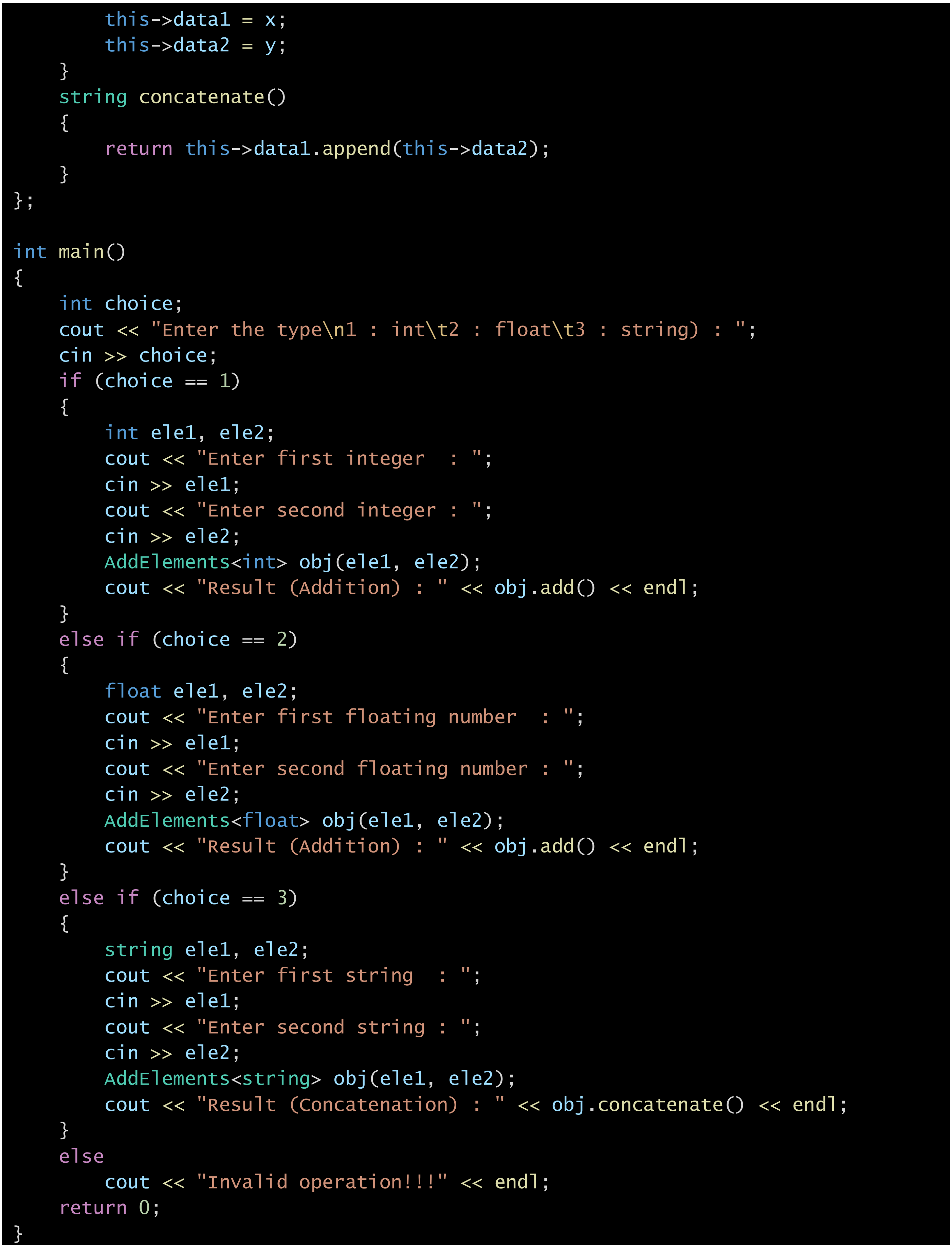
Throw j is caught by catch (int) of the inner try of fun2.

Throw g is caught by the catch (float) of the outer try of fun2.

4. Write a C++ program that takes a set of inputs. The type of input governs the kind of operation to be performed, i.e. concatenation for strings and addition for int or float. You need to write the class template AddElements which has a function add() for giving the sum of int or float elements. You also need to write a template specialization for the typed string with a function concatenate() to concatenate the second string to the first string.

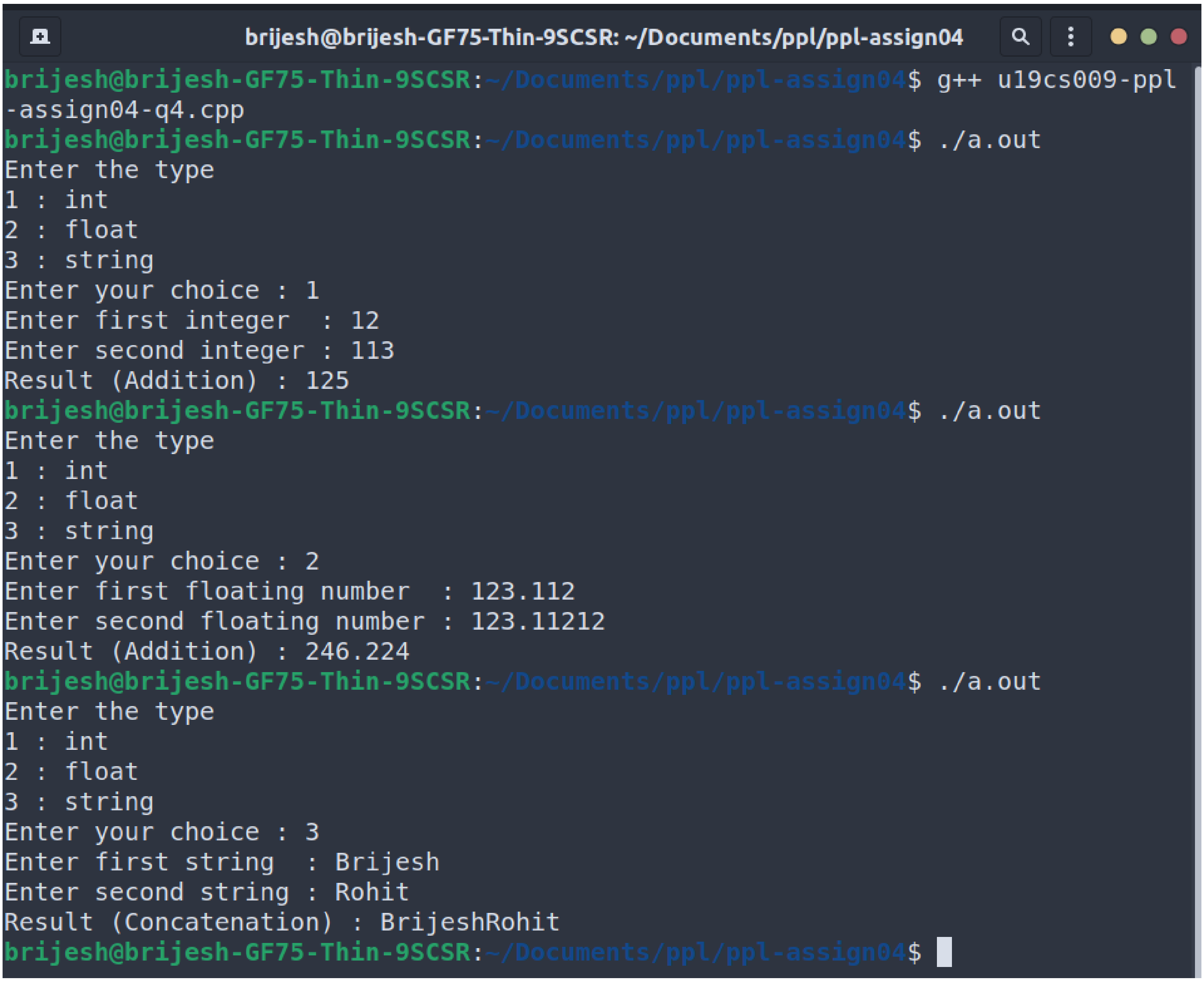
CODE=>

```
//U19CS009
//Brijesh Rohit

#include<iostream>
using namespace std;

template <class T>
class AddElements {
    T data1;
    T data2;
public:
    AddElements(T x, T y)
    {
        this->data1 = x;
        this->data2 = y;
    }
    T add()
    {
        return this->data1 + this->data2;
    }
};

template<>
class AddElements <string> {
    string data1;
    string data2;
public:
    AddElements(string x, string y)
    {
```



```
        this->data1 = x;
        this->data2 = y;
    }
    string concatenate()
    {
        return this->data1.append(this->data2);
    }
};

int main()
{
    int choice;
    cout << "Enter the type\n1 : int\t2 : float\t3 : string) : ";
    cin >> choice;
    if (choice == 1)
    {
        int ele1, ele2;
        cout << "Enter first integer  : ";
        cin >> ele1;
        cout << "Enter second integer : ";
        cin >> ele2;
        AddElements<int> obj(ele1, ele2);
        cout << "Result (Addition) : " << obj.add() << endl;
    }
    else if (choice == 2)
    {
        float ele1, ele2;
        cout << "Enter first floating number  : ";
        cin >> ele1;
        cout << "Enter second floating number : ";
        cin >> ele2;
        AddElements<float> obj(ele1, ele2);
        cout << "Result (Addition) : " << obj.add() << endl;
    }
    else if (choice == 3)
    {
        string ele1, ele2;
        cout << "Enter first string  : ";
        cin >> ele1;
        cout << "Enter second string : ";
        cin >> ele2;
        AddElements<string> obj(ele1, ele2);
        cout << "Result (Concatenation) : " << obj.concatenate() << endl;
    }
    else
        cout << "Invalid operation!!!" << endl;
    return 0;
}
```

OUTPUT=>



brijesh@brijesh-GF75-Thin-9SCSR: ~/Documents/ppl/ppl-assign04

```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ g++ u19cs009-ppl
-assign04-q4.cpp
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Enter the type
1 : int
2 : float
3 : string
Enter your choice : 1
Enter first integer  : 12
Enter second integer : 113
Result (Addition) : 125
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Enter the type
1 : int
2 : float
3 : string
Enter your choice : 2
Enter first floating number  : 123.112
Enter second floating number : 123.11212
Result (Addition) : 246.224
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ ./a.out
Enter the type
1 : int
2 : float
3 : string
Enter your choice : 3
Enter first string  : Brijesh
Enter second string : Rohit
Result (Concatenation) : BrijeshRohit
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ppl/ppl-assign04$ 
```