**Name: Khalid Abbas**

**Reg#2312236**

**Instructor: Ma’am Nadia Bashir.**

Q1) Write a C++ program that defines a

structure called Person with attributes for name, age, and city. Then,

declare and initialize a Person variable and print its details.

Code:

# include <iostream>

using namespace std;

int main()

{

struct Person

{

int age;

string name;

string city;

};

Person p1;

p1.age = 18;

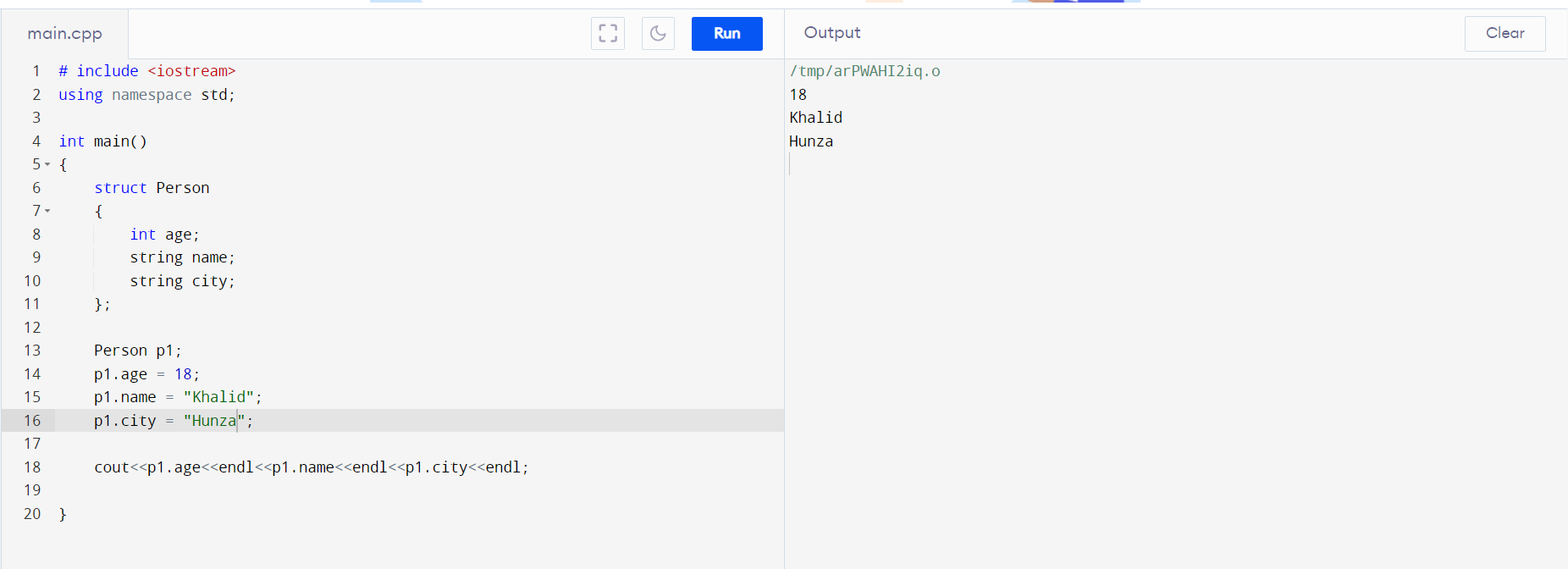
p1.name = "Khalid";

p1.city = "Hunza";

cout<<p1.age<<endl<<p1.name<<endl<<p1.city<<endl;

}

Output:



Q2) Create a C++ program that defines a structure

called Rectangle with attributes for length and width. Ask the user to enter

values for length and width, calculate the area, and display it.

Code:

# include <iostream>

using namespace std;

int main()

{

struct Rectangle

{

int length;

int width;

};

Rectangle r1;

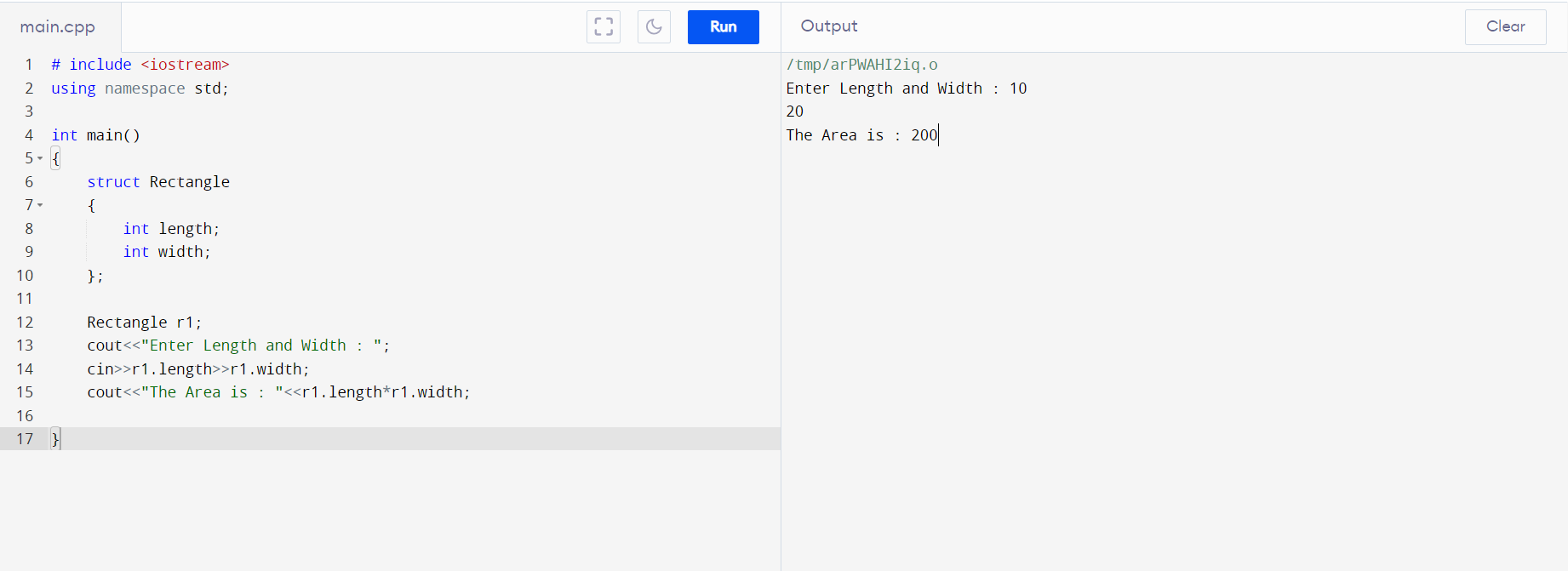
cout<<"Enter Length and Width : ";

cin>>r1.length>>r1.width;

cout<<"The Area is : "<<r1.length\*r1.width;

}

Output:



Q3) Define a structure Employee with

attributes id, name, and salary. Write a program that reads information for

three employees, stores it in an array of structures, and then prints the

details of each employee.

Code:

# include <iostream>

using namespace std;

int main()

{

struct Employee

{

int id[3];

string name[3];

int salary[3];

};

Employee e1;

for (int i = 0 ; i < 3 ; i++)

{

cout<<"Enter ID for Employee "<<i+1<<" : ";

cin>>e1.id[i];

cout<<"Enter NAME for Employee "<<i+1<<" : ";

cin>>e1.name[i];

cout<<"Enter SALARY for Employee "<<i+1<<" : ";

cin>>e1.salary[i];

}

for (int i = 0 ; i < 3 ; i++)

{

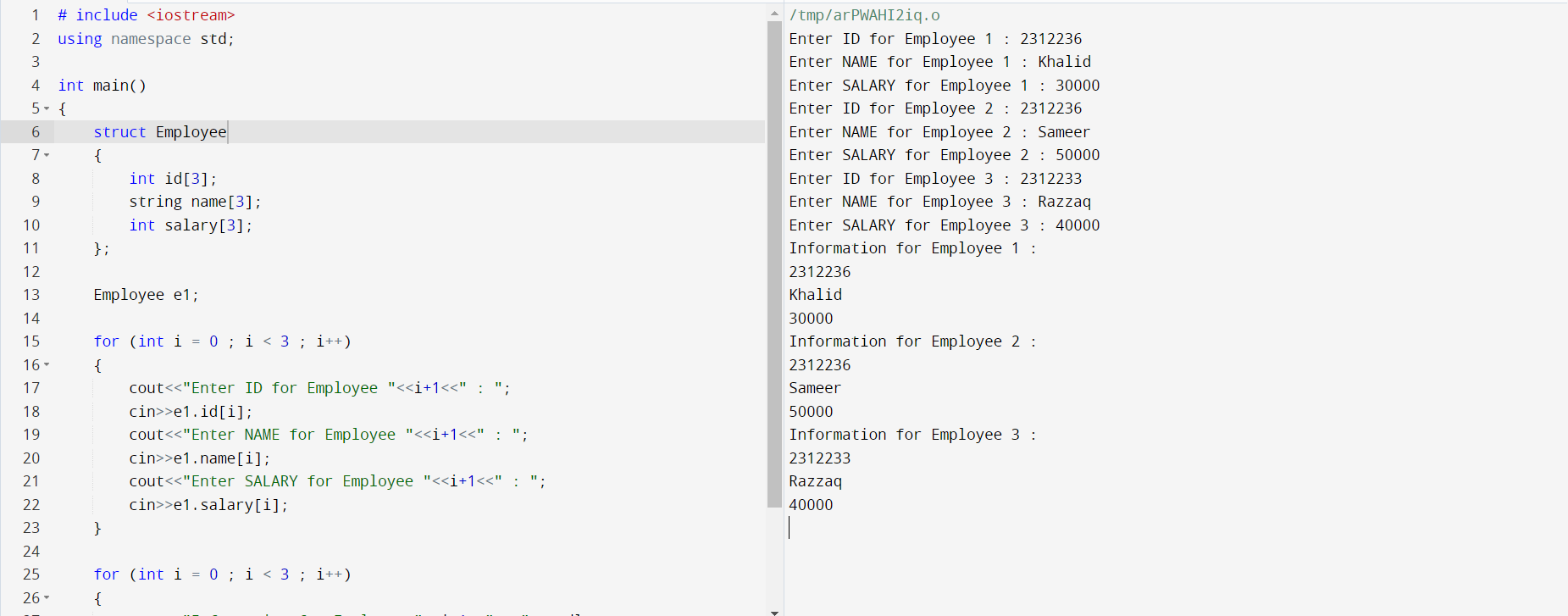
cout<<"Information for Employee "<<i+1<<" : "<<endl;

cout<<e1.id[i]<<endl<<e1.name[i]<<endl<<e1.salary[i]<<endl;

}

}

Output:



Q3) Create a C++ program that defines a structure

BankAccount with attributes for accountNumber, balance, and

accountHolderName. Implement basic banking operations like deposit and

Withdrawal.

Code:

# include <iostream>

using namespace std;

int main()

{

struct BankAccount

{

int account\_number;

// Let balance = 10,000

int balance = 10000;

string name;

};

BankAccount b1;

int choice;

while (choice != 3)

{

cout<<endl<<"1 - Withdraw \n2 - Deposit \n3 - Exit \nEnter Your Choice : ";

cin>>choice;

if (choice == 1)

{

cout<<"How much do u want to withdraw : ";

int withdraw\_amount;

cin>>withdraw\_amount;

if (withdraw\_amount <= b1.balance)

{

cout<<"Anount Withdrawn: "<<withdraw\_amount<<endl;

cout<<"Left amount in account : "<<b1.balance-withdraw\_amount;

}

else if (withdraw\_amount > b1.balance)

{

cout<<"You dont have enough money in ur account. ";

}

}

else if (choice == 2)

{

cout<<"How much do u want to Deposit : ";

int deposit\_amount;

cin>>deposit\_amount;

cout<<"Anount Deposited: "<<deposit\_amount<<endl;

cout<<"Updated account balance : "<<b1.balance+deposit\_amount;

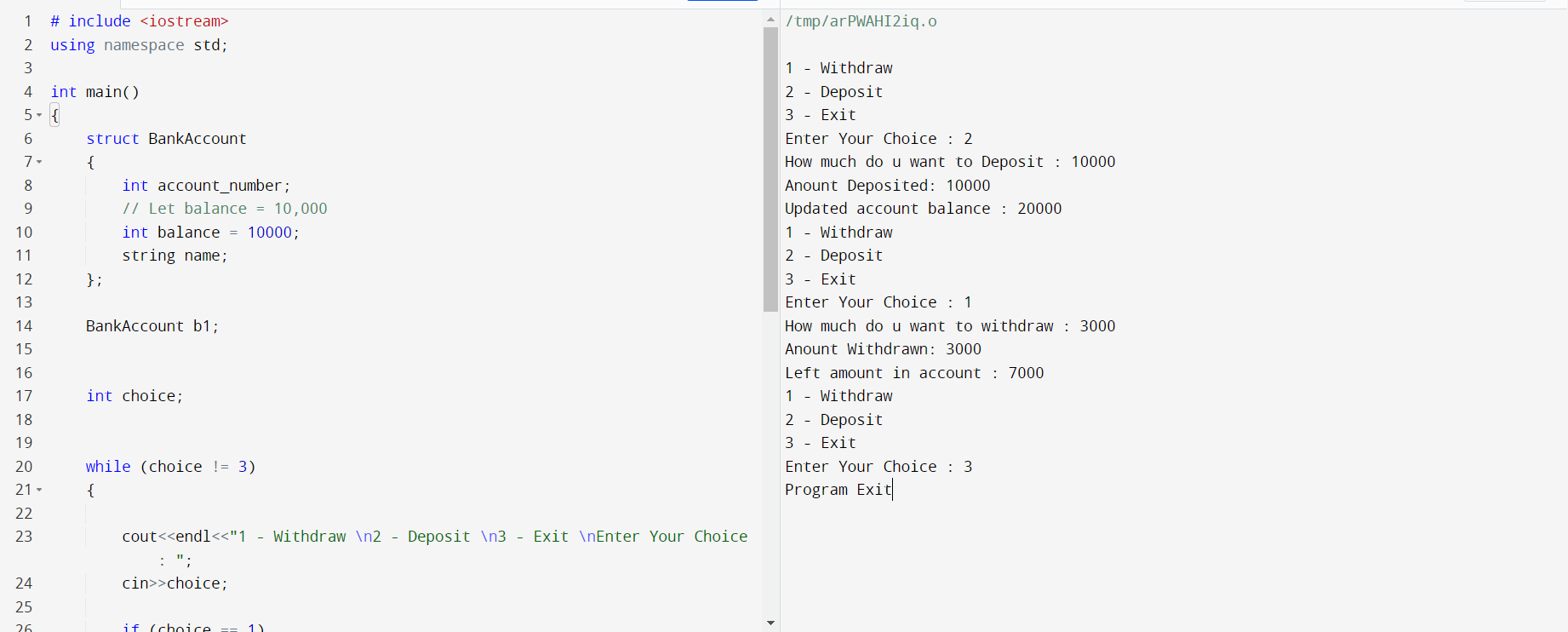
}

}

cout<<"Program Exit";

}

Output:



Q5) Define a structure Temperature with

attributes for temperature in Celsius and Fahrenheit. Write a program that

converts a given temperature from Celsius to Fahrenheit and vice versa.

Code:

# include <iostream>

using namespace std;

int main()

{

struct temp

{

float c , f;

};

temp t;

cout<<"Enter temp in celsuis : ";

cin>>t.c;

cout<<"Temp in Farenheit : "<<1.8\*t.c + 32<<endl;

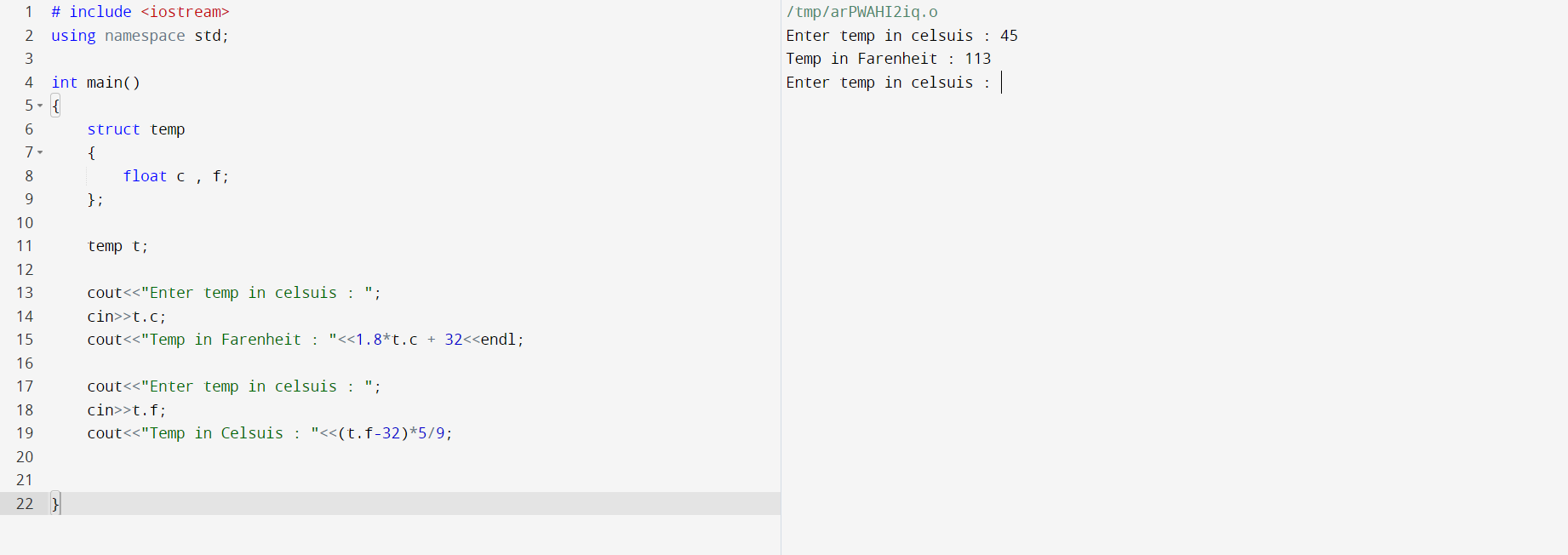
cout<<"Enter temp in celsuis : ";

cin>>t.f;

cout<<"Temp in Celsuis : "<<(t.f-32)\*5/9;

}

Output:



Q6) Create a C++ program that defines a

structure Course with attributes name and score. Ask the user to

enter the names and scores of three courses, calculate the average

score, and display it.

Code:

# include <iostream>

using namespace std;

int main()

{

struct course

{

string name[3];

int score[3];

};

int avrg = 0;

course c;

for (int i = 0 ; i < 3 ; i++)

{

cout<<"Enter Course "<<i+1<<" Name : ";

cin>>c.name[i];

cout<<"Enter Course "<<i+1<<" Score : ";

cin>>c.score[i];

}

for(int i = 0 ; i < 3 ; i++)

{

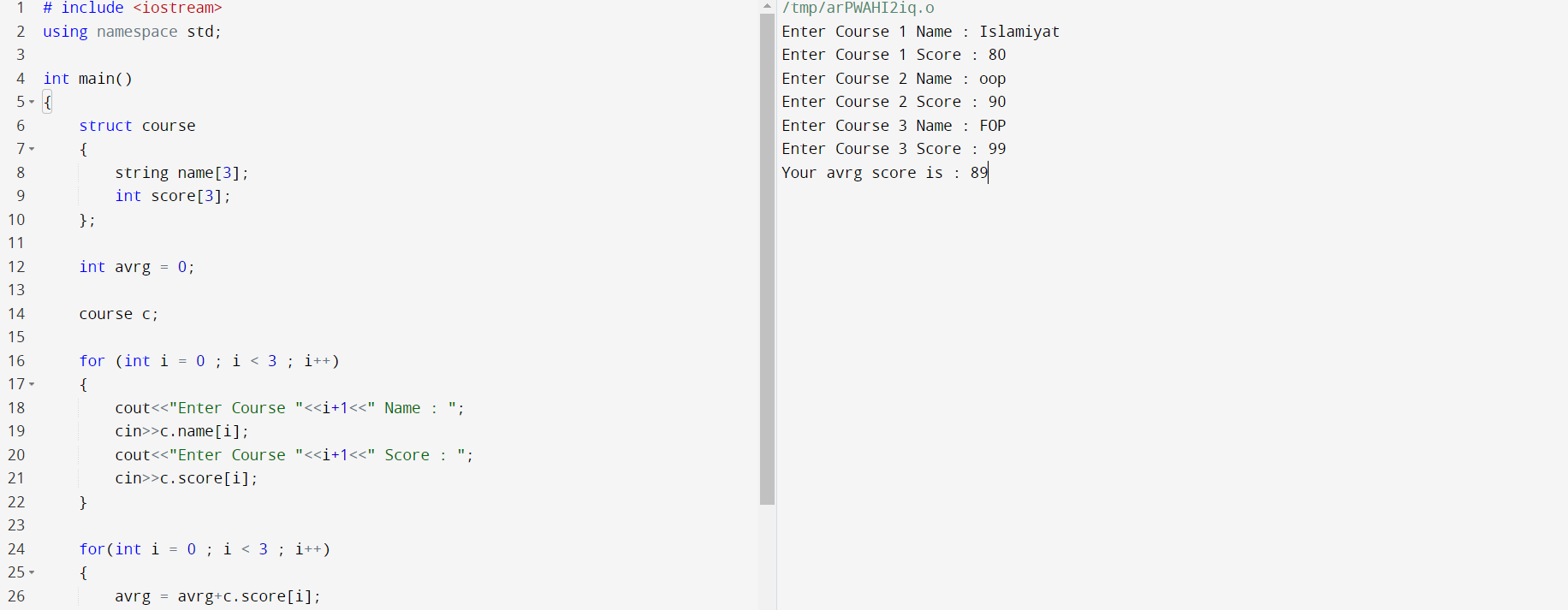
avrg = avrg+c.score[i];

}

cout<<"Your avrg score is : "<<avrg/3;

}

Output:



Q7) Write a C++ program that defines a structure

called Point with attributes x and y. Create two Point variables, swap their

values, and print the swapped values.

Code:

# include <iostream>

using namespace std;

int main()

{

struct point

{

int x , y;

};

point p;

int temp;

cout<<"Enter Value for first and second variable : ";

cin>>p.x>>p.y;

cout<<"Original \nFirst Variable : "<<p.x<<"\nSecond Variable : "<<p.y;

temp = p.x;

p.x = p.y;

p.y = temp;

cout<<"\nSwapped \nFirst Variable : "<<p.x<<"\nSecond Variable : "<<p.y;

}

Output:



Q7) Define a structure Complex to

represent complex numbers with real and imaginary parts. Write a

program that adds two complex numbers and displays the result.

Code:

# include <iostream>

using namespace std;

int main()

{

struct complex

{

int imaginary1;

int real1;

int imaginary2;

int real2;

};

complex c;

cout<<"Enter Real part of first number : ";

cin>>c.real1;

cout<<"Enter Imaginary part of first number : ";

cin>>c.imaginary1;

cout<<"Enter Real part of second number : ";

cin>>c.real2;

cout<<"Enter Imaginary part of first number : ";

cin>>c.imaginary2;

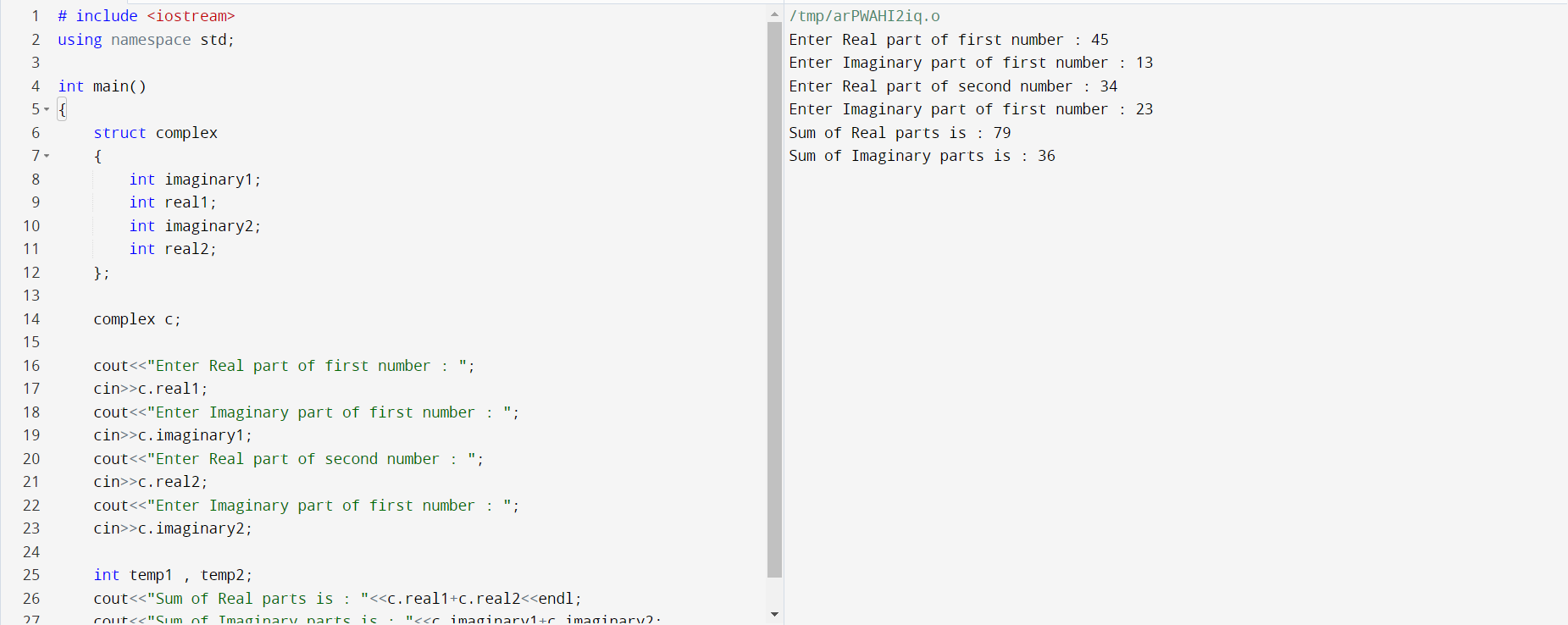
int temp1 , temp2;

cout<<"Sum of Real parts is : "<<c.real1+c.real2<<endl;

cout<<"Sum of Imaginary parts is : "<<c.imaginary1+c.imaginary2;

}

Output:



Q8) Create a C++ program that defines a structure

Student with attributes rollNumber, name, and marks. Initialize an array of

Student structures and print the details of the student with the highest

Marks.

Code:

# include <iostream>

using namespace std;

int main()

{

struct student

{

int roll\_no;

string name;

int marks;

};

student s[3];

for (int i = 0 ; i < 3 ; i ++)

{

cout<<"Enter roll no for "<<i+1<<" : ";

cin>>s[i].roll\_no;

cout<<"Enter name for "<<i+1<<" : ";

cin>>s[i].name;

cout<<"Enter marks for "<<i+1<<" : ";

cin>>s[i].marks;

}

if (s[0].marks >= s[1].marks && s[0].marks >= s[2].marks)

{

cout<<"Roll no "<<s[0].roll\_no<<"\nName "<<s[0].name<<"\nMarks "<<s[0].marks;

}

if (s[1].marks >= s[0].marks && s[1].marks >= s[2].marks)

{

cout<<"Roll no "<<s[1].roll\_no<<"\nName "<<s[1].name<<"\nMarks "<<s[1].marks;

}

if (s[2].marks >= s[0].marks && s[2].marks >= s[1].marks)

{

cout<<"Roll no "<<s[2].roll\_no<<"\nName "<<s[2].name<<"\nMarks "<<s[2].marks;

}

}

Output:

