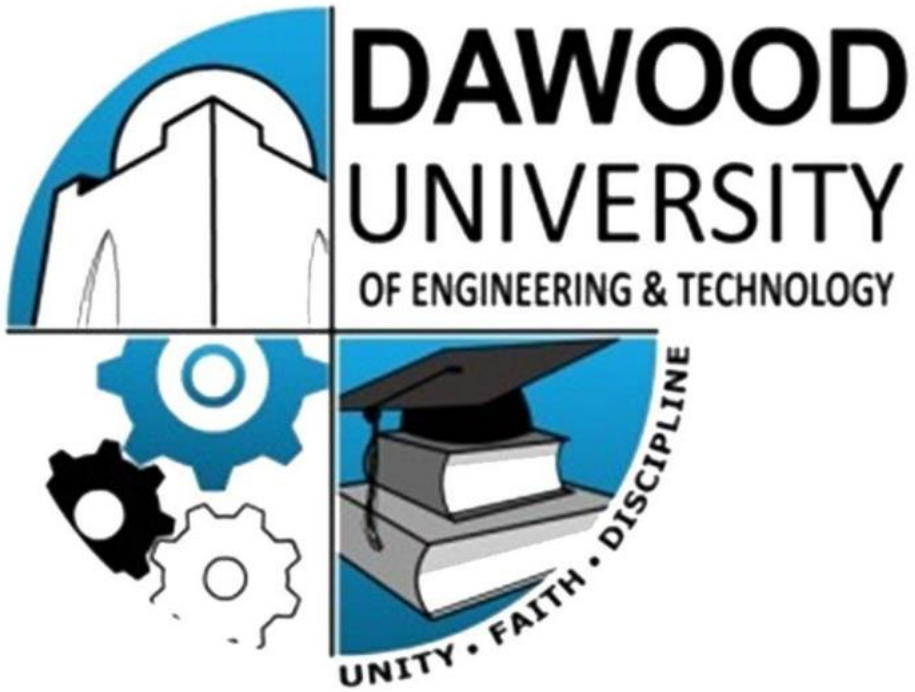


NAME: **SAFI AHMED**

ROLL NO: **22F-BSCS-35**

SUBJECT: **PROGRAMMING FUNDAMENTALS PRACTICAL**

SUBMITTED TO: **ENGR. SOFIA HAJANO**

NAME :	**SAFI AHMED**
ROLL NUMBER :	**35**
DEPARTMENT :	**COMPUTER SCIENCE**
SEMESTER :	**2ND**
YEAR :	**1ST**
SUBJECT :	**PROGRAMMING FUNDAMENTAL (PRACTICAL)**

PF Practical # 1

Question # 1: For each of the following statement write down the flow chart symbol you will use.

STATEMENT	FLOW CHART SYMBOL
Terminate the process	Terminal
Set the variable a = 5	Process
Calculate the square of a number	Process
Getting name of the user	Input/Output
Initiate the process	Process
Continue a flow chart on the next page	Connector
Displaying the marks of student	Process
Executing another process	Process
Prompting the user	Input/Output
Continue a flow chart on the same page	Connector

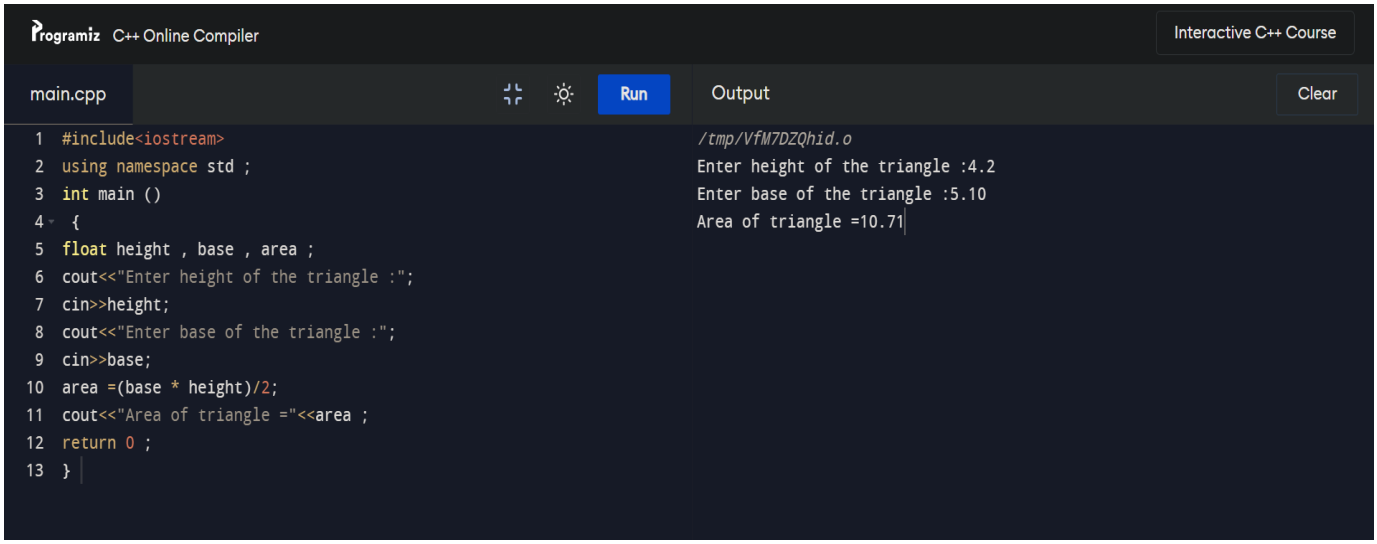
QUESTION # 2: Write down the problem statement for the following flow chart.

Ans. Take a number as input from the User and return it after dividing by 1000.

QUESTION # 3: For each of the following problem statement, create the IPO chart with algorithm and Flow chart.

Problem Statement 1

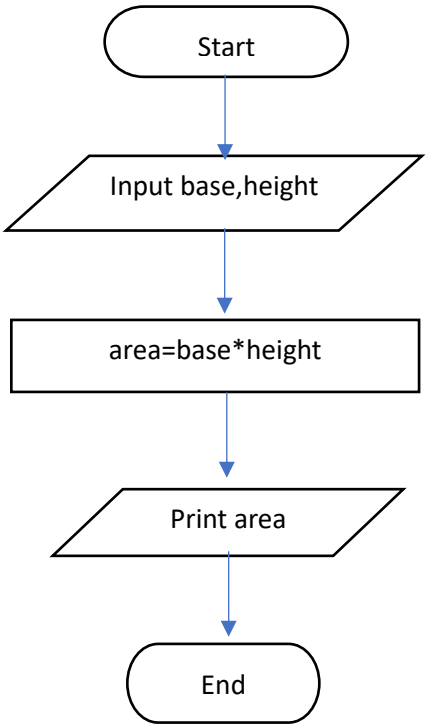
- Write a computer program that accepts the base and height of a right angle triangle from the user and displays the area of the triangle.



➢ IPO Chart & Algorithm :

Input	Processsing	Output
• Base of rectangle • Height of rectangle	**Processing Items :** area = width * height **Algorithm:** Step 01: Start Step 02: Input Base and Height from the user Step 03: Calculate area as: area = Base * Height Step 04: Print area Step 05: End	• Area of triangle

➢ Flow Chart :



Problem Statement # 2

- Write a program that asks the user to enter a four digit integer number and displays the average of all the four digits of a number.
  Sample output:
  Input:
  number = 9425

  Output:
  average = 9+4+2+ 5/ 4 = 5

Programiz C++ Online Compiler — Interactive C++ Course

main.cpp — Run — Output — Clear

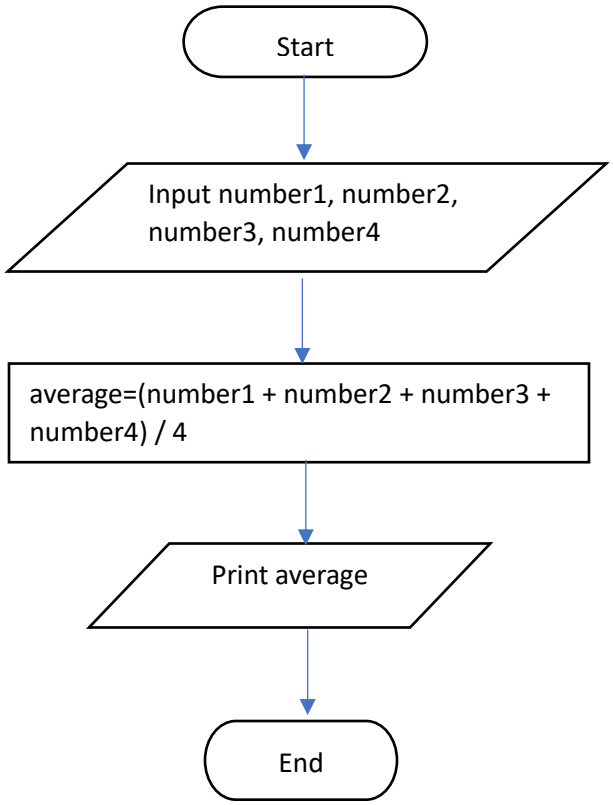
```cpp
#include<iostream>
using namespace std;
int main ()
{
    int number, digit1, digit2, digit3, digit4, sum;
    cout<<"Enter a four digit integer number:" ;
    cin>>number ;
    digit1 = number /1000;
    digit2 = (number %1000) / 100 ;
    digit3 = (number %100) /10 ;
    digit4 = number %10 ;
    sum = digit1 + digit2 + digit3 + digit4 / 4;
    cout<<"The average of all the four digits of a number is:"<<sum ;
    return 0;
}
```

```
/tmp/bwspezthls.o
Enter a four digit integer number:9425
The average of all the four digits of a number is:16
```

➢ IPO Chart & Algorithm :

Input	Processing	Output
• First number • Second number • Third number • Fourth number	**Processing Items:** average = (number1 + number2 + number3 + number4)/4 **Algorithm:** Step 01: Start Step 02: Input number1, number2 ,number3 and number4 from the user Step 03: Calculate average as: average = (number1 + number2 + number3 + number4)/4 Step 04: Print average Step 05: End	• Average of numbers

➢ Flow Chart :



PF Practical # 2

Question # 1 Briefly answer the following questions:

a. How conditional logic is different than sequential logic?

Conditional logic allows the program to make decisions based on conditions, while sequential logic executes instructions in a linear order without branching.

OR

Conditional logic is used to make decisions, while sequential logic is used to control the order of execution. Conditional logic is based on the value of a variable or expression, while sequential logic is based on a sequence of instructions.

Feature	Conditional logic	Sequential Logic
Purpose	Make decisions	Control the order of execution
Based on	Value of a variable or expression	Sequence of instructions
Examples	If-else statement, switch statement	While loop, for loop

b. Can we use multi-way selection if we have two conditions and three statement sets?

No, multi-way selection is typically used for handling multiple conditions and their corresponding statement sets, not specifically for two conditions and three statement sets .

OR

No, we cannot use multi-way selection if we have two conditions and three statement sets. Multi-way selection is used when there are more than three possible outcomes. In this case, there are only two possible outcomes, so we would need to use a different type of logic.

Type of Logic	Number of conditions	Number of statements sets
One-way selection	1	2
Two-way selection	2	3
Multi-way selection	>=3	>=4

c. How can you differentiate multi-way and choice-way selection?

Multi-way selection involves choosing one option among multiple options based on conditions, while choice-way selection is a broader concept that encompasses various decision-making techniques in programming.

OR

Feature	Multi-way selection	Choice-way Selection
Number of possible outcomes	More than three	Fixed number
Example	The user enters a number between 1 and 10, and the program prints the number's corresponding letter grade.	The user enters a number from 1 to 5, and the program prints the corresponding day of the week.

d. How many statement sets are there in multi-way selection if we have 8 conditions?

There can be up to 8 statement sets in a multi-way selection if we have 8 conditions. The number of statement sets is equal to the number of conditions, plus one. This is because there is one statement set for each possible outcome, plus one statement set for the default outcome.

OR

In multi-way selection with eight conditions, there are usually eight corresponding statement sets, each associated with a specific condition.

e. What is advantage of using repetitive logic?

Efficiency: Repetitive logic can help us to write more efficient code by reducing the amount of code that we need to write. For example, if we need to print a list of numbers, we can use a loop to repeat the print statement for each number in the list. This is more efficient than writing a separate print statement for each number.

Reusability: Repetitive logic can help us to write more reusable code by allowing us to encapsulate a repeated task into a function. This function can then be called from other parts of the program, which can help us to reduce code duplication.

Readability: Repetitive logic can help us to write more readable code by grouping related code together. This can make the code easier to understand and maintain.

OR

Repetitive logic, such as loops, automates repetitive tasks, reduces code duplication, and improves code efficiency by executing a block of code multiple times.

f. When will you use counter-controlled repetition logic?

Counter-controlled repetition logic is used when the number of iterations is known beforehand, and a loop is executed a specific number of times using a counter variable

OR

Counter-controlled repetition logic is used when the number of times a block of code needs to be executed is known beforehand. For example, if you need to print the numbers from 1 to 10, you can use a counter-controlled loop to do so.

g. When will you use sentinel-controlled repetition logic?

Sentinel-controlled repetition logic is used when the number of times a block of code needs to be executed is not known beforehand. For example, if you need to read numbers from the user until they enter a negative number, you can use a sentinel-controlled loop to do so.

OR

Sentinel-controlled repetition logic is used when a loop needs to repeat until a specific condition (sentinel value) is encountered, allowing flexibility in the number of iterations

h. Is it possible to use multi-way logic in place on choice-way logic?

Yes, it is possible to use multi-way logic in place of choice-way logic. However, it is not always the best option.

Multi-way logic allows you to have multiple conditions that can be met, while choice-way logic only allows for one condition to be met. This means that if you have a situation where multiple conditions could be met, you would need to use multi-way logic.

OR

Yes, multi-way logic can be used in place of choice-way logic, as it encompasses the concept of making choices based on conditions and supports multiple conditions and options.

Question # 2 For each of the following statement specify which type of conditional or repetition logic you will use.

A. One-Way Selection

B. Two-Way Selection

C. Multi-Way Selection

D. Choice-Way Selection

E. Counter-Controlled Repetition

F. Sentinel-Controlled Repetition

G. None of them

Statement	Logic
Specifying grade on the basis of percentage	Two-way selection
Checking number is even or odd	One-way selection
Getting three numbers as input from the user	Sentinel-controlled repetition
Calculating the area of triangle	None of them
Checking number is positive or not	One-way selection
Checking number is prime or composite	Counter-controlled repetition

Question # 2 For each of the following problem statement, create the IPO chart with algorithm and flow chart.

Problem Statement 1

Write a computer program that asks the user to enter three angles of a triangle. The program displays whether the triangle is right-angle, acute-angle or obtuse-angle.

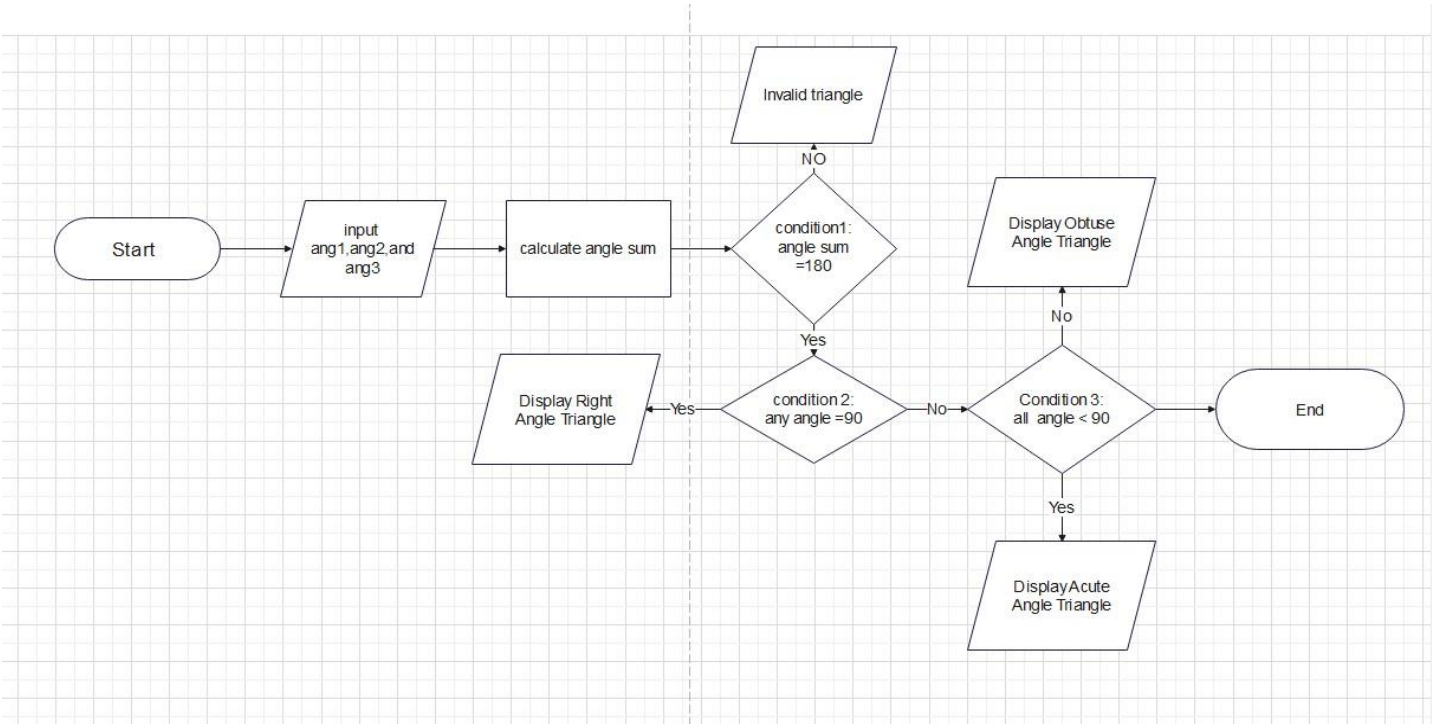
```cpp
#include <iostream>

using namespace std;

int main() {
  // Read three angles of a triangle from the user.
  double angle1, angle2, angle3;
  cout << "Enter three angles of a triangle: ";
  cin >> angle1 >> angle2 >> angle3;

  // Calculate the sum of the three angles.
  double sum = angle1 + angle2 + angle3;

  // Determine the type of triangle.
  if (sum == 180.0) {
    cout << "The triangle is a right-angle triangle.\n";
  } else if (sum < 180.0) {
    cout << "The triangle is an acute-angle triangle.\n";
  } else {
    cout << "The triangle is an obtuse-angle triangle.\n";
  }

  return 0;
}
```

```
Enter three angles of a triangle: 90
40
65
The triangle is an obtuse-angle triangle.

--------------------------------
Process exited after 4.895 seconds with return value 0
Press any key to continue . . .
```

IPO AND ALGORITHMS

INPUT	PROCESSING	OUTPUT
Three angles (angle1, angle2, angle3)	Processing Item : Calculate the sum of the angles (angleSum) Algorithms: Step 1: Start Step 2: Create three variables to store the angles of the triangle (angle1, angle2, angle3) Step 3: Prompt the user to enter the values for angle1, angle2, and angle3 Step 4: Calculate the sum of the three angles and store it in a variable (angleSum) Step 5: Check the following conditions: - If angleSum equals 180, then it is a valid triangle - If any angle is equal to 90, then it is a right-angle triangle - If all angles are less than 90, then it is an acuteangle triangle - If any angle is greater than 90, then it is an obtuseangle triangle Step 6: Display the result based on the conditions checked Step 7: End	Type of the triangle (Rightangle, Acuteangle, Obtuseangle)

FLOW Chart



Problem Statement 2

Write a computer program that displays the sum of first 10 odd multiples of 3.

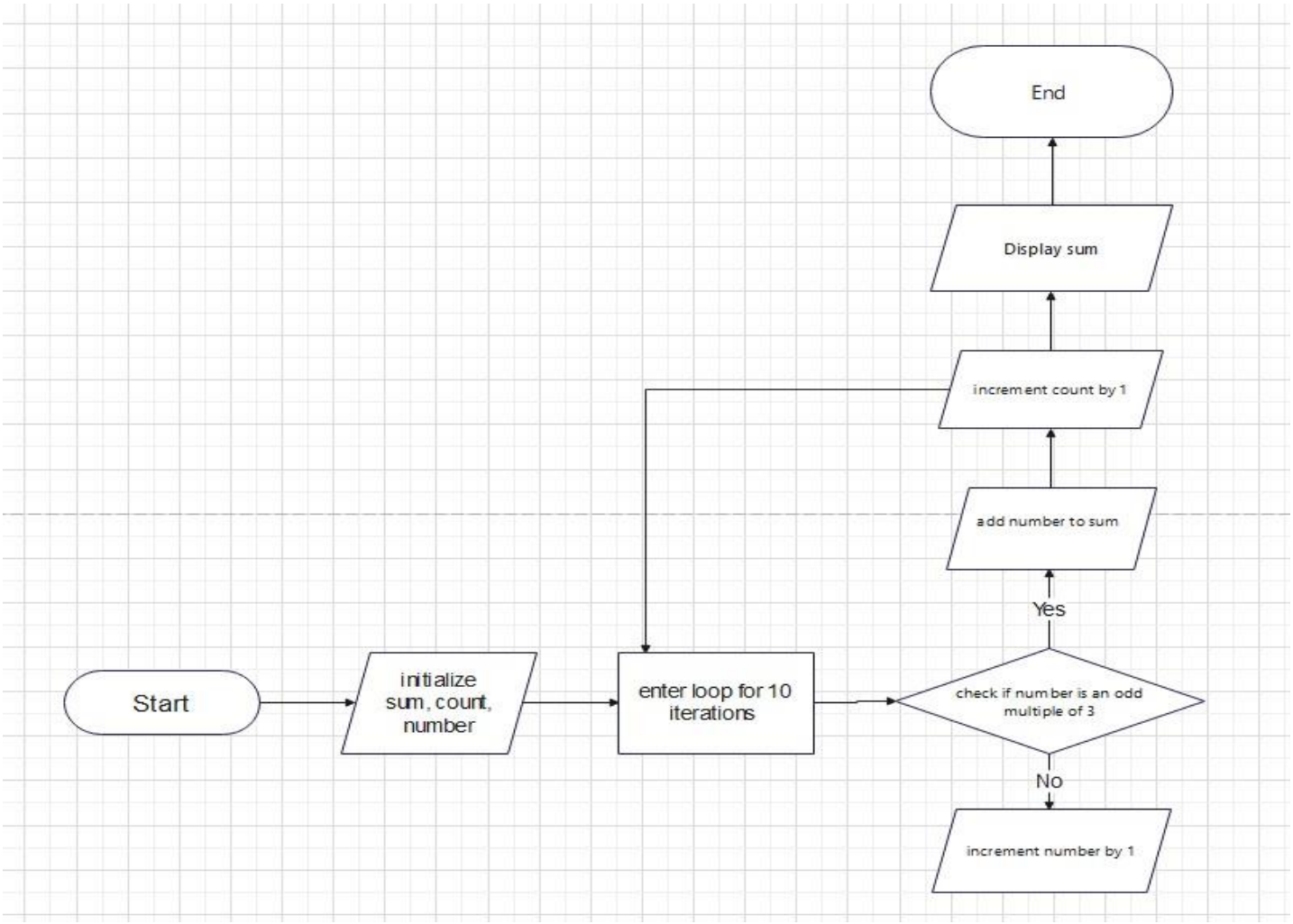
```cpp
#include <iostream>
using namespace std;

int main() {
  int sum = 0, i = 3;


  while (i <= 30) {
    sum += i;
    i += 6;
  }


  cout << "The sum of first 10 odd multiples of 3 is: " << sum << endl;

  return 0;
}
```

```
he sum of first 10 odd multiples of 3 is: 75
```

IPO AND ALGORITHMS

INPUT	PROCESSING	OUTPUT
None	Processing Item : Calculate the sum of the first 10 odd multiples of 3 Algorithms: Step 1: Start Step 2: Create variables to store the sum (sum = 0), count (count = 0), and number (number = 1) Step 3: Enter a loop to iterate 10 times - Check if the number is an odd multiple of 3 (number % 3 == 0 and number % 2 != 0) - If true, add the number to the sum - Increment the count by 1 - Increment the number by 1 Step 4: Display the sum Step 5: End	Sum

FLOWCHART



Problem Statement 3

Write a computer program that displays the sum of last 5 four digit multiples of 5.

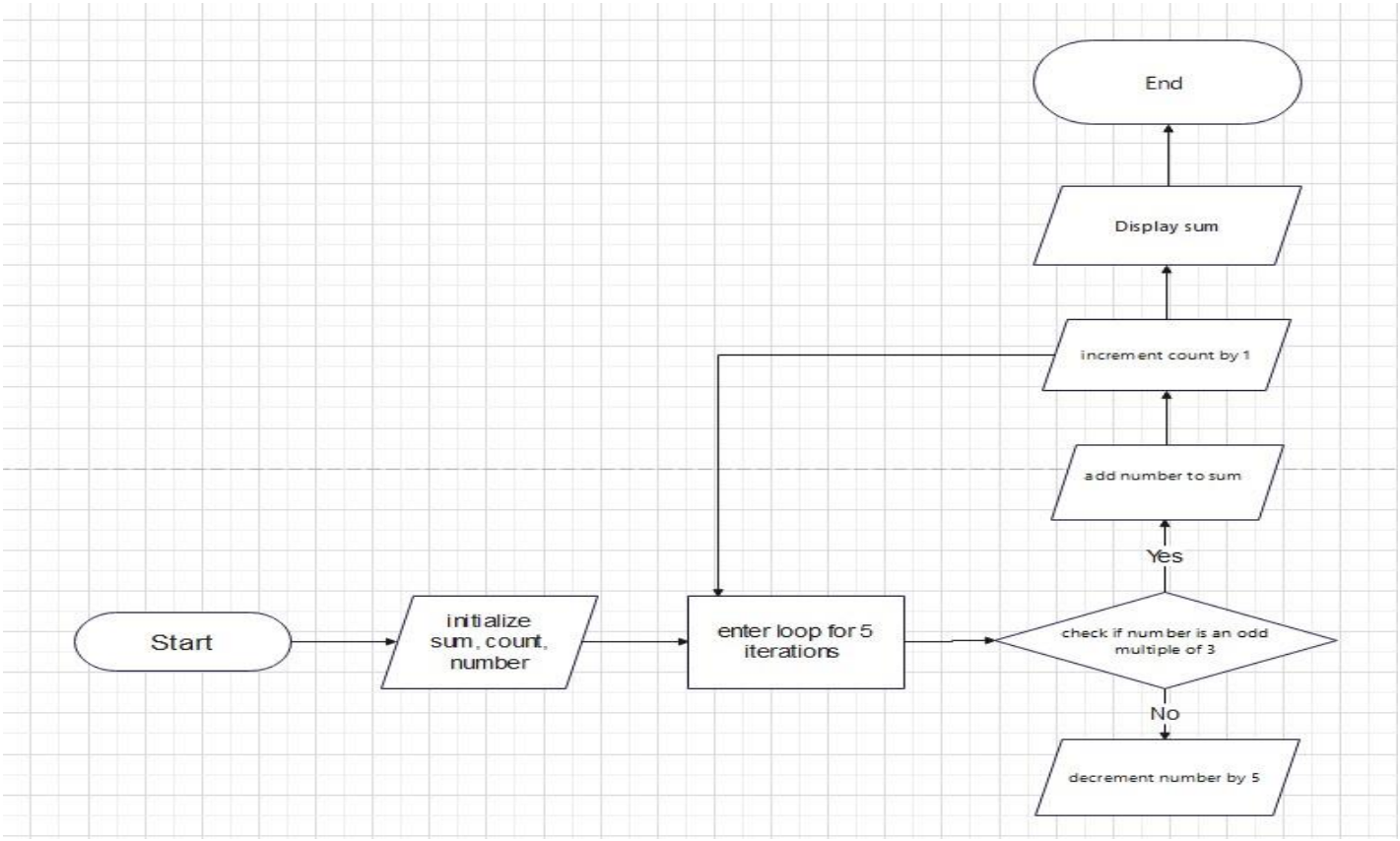
```cpp
#include <iostream>
using namespace std;

int main() {
  int sum = 0, i = 9995;

  while (i >= 9990) {
    sum += i;
    i -= 5;
  }

  cout << "The sum of last 5 four digit multiples of 5 is: " << sum << endl;

  return 0;
}
```

```
The sum of last 5 four digit multiples of 5 is: 19985

--------------------------------
Process exited after 0.3876 seconds with return value 0
Press any key to continue . . .
```

IPO AND ALGORITHMS:

INPUT	PROCESSING	OUTPUT
None	Processing Item : Calculate the sum of the last 5 four-digit multiples of 5 Algorithms: Step 1: Start Step 2: Create variables to store the sum (sum = 0), count (count = 0), and number (number = 10000) Step 3: Enter a loop to iterate 5 times - Check if the number is a four-digit multiple of 5 (number % 5 == 0 and number >= 10000 and number <= 9999) - If true, add the number to the sum - Increment the count by 1 - Decrement the number by 5 Step 4: Display the sum Step 5: End	Sum

FLOWCHART:



PF Practical # 3

Question # 1 Briefly answer the following questions:

a. What is Integrated Development Environment (IDE)?

An IDE is a special software that comes with all the tools to help write code, Edit it and run. Examples are VS Code, Code blocks etc . An Integrated Development Environment (IDE) is a software application that provides comprehensive tools and features for software development. It serves as a centralized platform that combines various components and functionalities to streamline the development process. IDEs typically include a source code editor, a compiler or interpreter, a debugger, and build automation tools. They often offer features like code completion, syntax highlighting, version control integration, and debugging capabilities, among others. IDEs aim to enhance productivity and facilitate efficient coding by offeringa unified environment for writing, editing, compiling, testing, and debugging code, all within a single application. They are widely used by developers across different programming languages and help streamline the software developmentworkflow.

b. What is the role of main function in a C++ program?

The main code is stored inside this function. It basically calls to execute the body. The main function in a C++ program serves as the entry point for program execution. It is a mandatory function that must be present in every C++ program. When the program is executed, the operating system starts executing the

instructions within the main function. It is responsible for coordinating and controlling the program's flow by calling other functions, performing calculations, handling user input and output, and making decisions based on conditions. The main function typically returns an integer value that indicatesthe program's termination status, which can be used by the operating system or calling program to determine if the execution was successful or encountered an error.

c. Will the program execute if there is no any main function in it?

No, a C++ program will not execute if there is no `main` function in it. The `main`function is the entry point for programexecution, and its presence is mandatory in every C++ program. The compiler looks for the `main` function as the starting point to begin executing the program. If there is no `main` function, the compiler will generate an error, indicatingthat the program cannot be executed. Therefore, the `main` function is crucial and required for the program to run successfully.

d. What will happen if there are more than one main functions in the program?

No, It will not run a C++ program will not execute if there is no `main`function in it. The `main` function is the entry point for program execution, and its presence is mandatory in every C++ program. The compiler looks for the `main` function as the starting point to begin executing the program. If there is no `main` function, the compiler will generate an error, indicating that the program cannot be executed. Therefore, the `main` function is crucial and required for the program to run successfully.

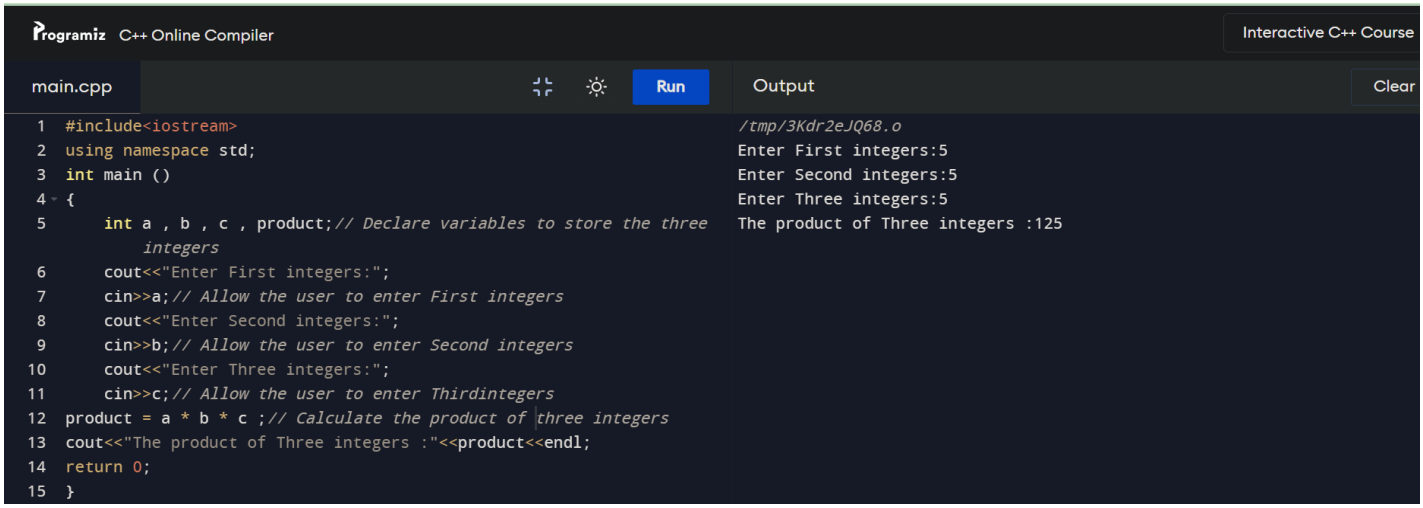
e. Can we write the preprocessor directives after the main function?

No, preprocessor directives cannot be written after the main function in C++. Preprocessor directives, such as #include or #define, are processed by the preprocessorbefore the actual compilation of the code begins. They are used to manipulate the source code before it is compiled. The preprocessor runs as a separate step andperforms tasks like including header files, defining macros, and conditional compilation. Since the main function is the entry point for program execution, it needs to be defined before any code that depends on orinteracts with it. Therefore, preprocessor directives should be placed before the main function, typically at the beginning of the file, to ensure that they are processed correctly before the program execution starts.

f. What is syntax error and logical error?

A syntax error occurs when one makes an error in the Grammar of the language. The later occurs when a logical error is a bug in a program that causes it to operate incorrectly, but not to terminate abnormally (or crash). A syntax error is a type of error that occurs when the code violates the grammar rules of the programming language. Itmeans the code does not conform to the syntax specified by the language, resulting in a compilation error. Syntax errorsare detected by the compiler or interpreter during the compilation or parsing phase and prevent the program from running .

Question # 2Write a complete program that calculates and displays the product of three integers. Add comments to the code where appropriate.



Question # 3 What, if anything, prints when each of the following C++ statements is performed? If nothing prints, then answer "nothing." Assume x = 2 and y = 3.

a. `cout<<x;`

Output: 2

b. `cout<<x + x;`

Output: 4

c. `cout<<"x=";`

Output: x=

d. `cout<<"x ="<< x;`

Output: x = 2

e. `cout<<x + y<<" = "<< y + x;`

Output: 5 = 5

f. `z =x + y;`

This is an assignment statement and not a cout statement, so it does not produce any output.

Question # 4 Write a program that asks the user to enter two numbers, obtains the two numbers from the user and prints the sum, product, difference, and quotient of the two numbers.

Programiz C++ Online Compiler — Interactive C++ Course

main.cpp — Run — Output — Clear

```cpp
#include<iostream>
using namespace std;
int main ()
{
    int num1 , num2 ;
    cout<<"Enter the First number :";
    cin>>num1;
    cout<<"Enter the Second number :";
    cin>>num2;
    int sum =num1 + num2;
    int difference =num1 - num2;
    int product =num1 * num2;
    int qoutient=num1 / num2;
cout<<"The sum of the two numbers is :"<<sum<<endl;
cout<<"The difference of the two numbers is :"<<difference<<endl;
cout<<"The product of the two numbers is :"<<product<<endl;
cout<<"The qoutient of the two numbers is :"<<qoutient<<endl;
    return 0;
}
```

```
/tmp/3Kdr2eJQ68.o
Enter the First number :2
Enter the Second number :4
The sum of the two numbers is :6
The difference of the two numbers is :-2
The product of the two numbers is :8
The qoutient of the two numbers is :0
```

Question # 5 Consider the following C++ program that calculates the area of triangle;

```
#include <iostream>
#include <conio>
 using namespace std;
 int MAIN ( )
 }
 Char height , base , area;
 cout<<”Enter height of triangle:  ”;
 cin>>height;
 cout<<”Enter base of triangle: ”;
 cin>>BASE;
 area = (height / base ) * 2;
 cout<<”Area of triangle =”<<Area;
 getch( )
 return ;
 {
```

a. Find all syntax errors.

b. Find all logical errors.

c. Correct and rewrite the program.

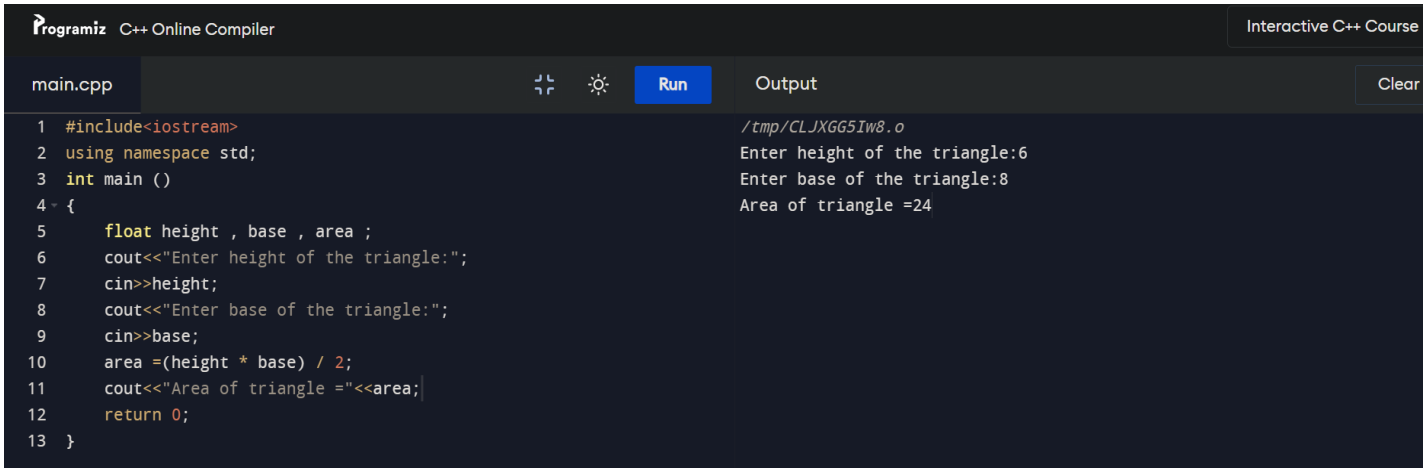
Syntax Errors

1. The `include` statement is misspelled. It should be `**#include**<conio.h>`.
2. Incorrect capitalization of the main function declaration. The function `MAIN()` is not declared in the `int` scope. It should be declared as `int main()`.
3. Missing opening and closing braces for the main function.
4. Incorrect capitalization of variables `base and area` .The variable BASE and Area is misspelled. It should be base and area .
5. The expression (height / base) * 2 is incorrect. It should be `(height * base) / 2`.

Logical Errors

1. The program does not check if the user enters valid input for the height and base of the triangle.
2. The program does not calculate the area of the triangle correctly.
3. Declaring `height`, `base`, and `area`as characters insteadof numeric types (int or double) to store numerical values.
4. Incorrect calculation of the area using integer divisioninstead of floating-point division.

Corrected Program



Question # 6 For each of the following, specify whether itis a valid or invalid identifier. In case of invalid identifier give the reason i.e. why it is invalid?

Identifier	Valid / Invalid	Remarks
second_number	**Valid**	
second number	**Invalid**	**Because No spaces are allowed**
2nd_number	**Invalid**	**Because the Variable doesn't start with a number**
number_2nd	**Valid**	
2nd_#	**Invalid**	**because no special characters are allowed, Also the Variable doesn't start with a number.**
#_second	**Invalid**	**because no special characters are allowed**
include_second_number	**valid**	
2nd_number_include	**Invalid**	**because no special characters are allowed**
2nd_#_include	**Invalid**	**because no special characters are allowed, Also the Variable doesn't start with a number.**
1#_include_sec	**Invalid**	**because no special characters are allowed, Also the Variable doesn't start with a number.**

Question # 7 For each of the data item given below, write down the example value and the data type.

S.No	Data Item	Example Value	Data type
1	CNIC number		
2	Phone number		
3	Roll number		
4	Option in MCQ		
5	Percentage of marks		

S.No	Data Item	Example Value	Data type
1	CNIC number	44201-1234567-8	string
2	Phone number	+92-333-4567890	string
3	Roll number	1234	integer
4	Option in MCQ	A, B, C, D	logical
5	Percentage of marks	98.5	float

PF Practical # 4

Question # 1 Briefly answer the following questions:

a. What does flow chart represent?

A flowchart represents a visual diagram that illustrates the sequence of steps or decisions in a process or algorithm. **OR** A Pictorial Flow of the Program

b. What is sequential control flow?

Sequential control flow means executing instructions in a straightforward, systematic manner without branching or decision-making. **OR** Execution of the code line by line.

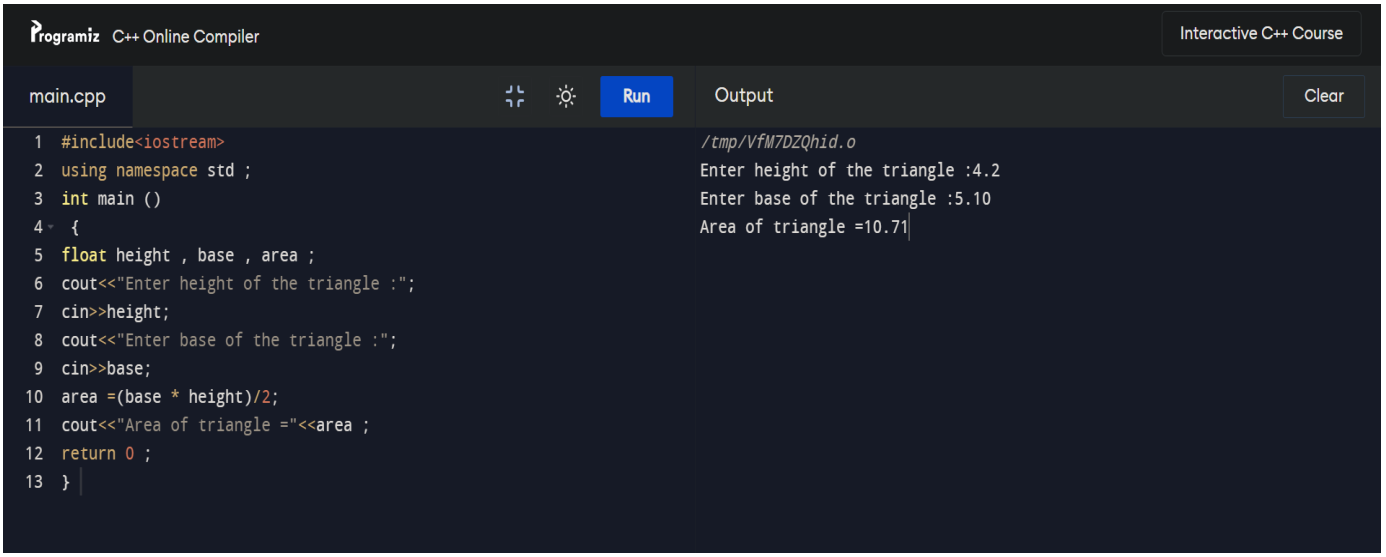
c. What do arrows represent in a flowchart?

Arrows in a flowchart show the direction of control flow, indicating the order or sequence in which steps or instructions are executed.

Question # 2 For each of the following problem statement, create C++ program .

Problem Statement 01

Write a computer program in C++ that accepts the base and height of a right angle triangle from the user and displays the area of triangle.



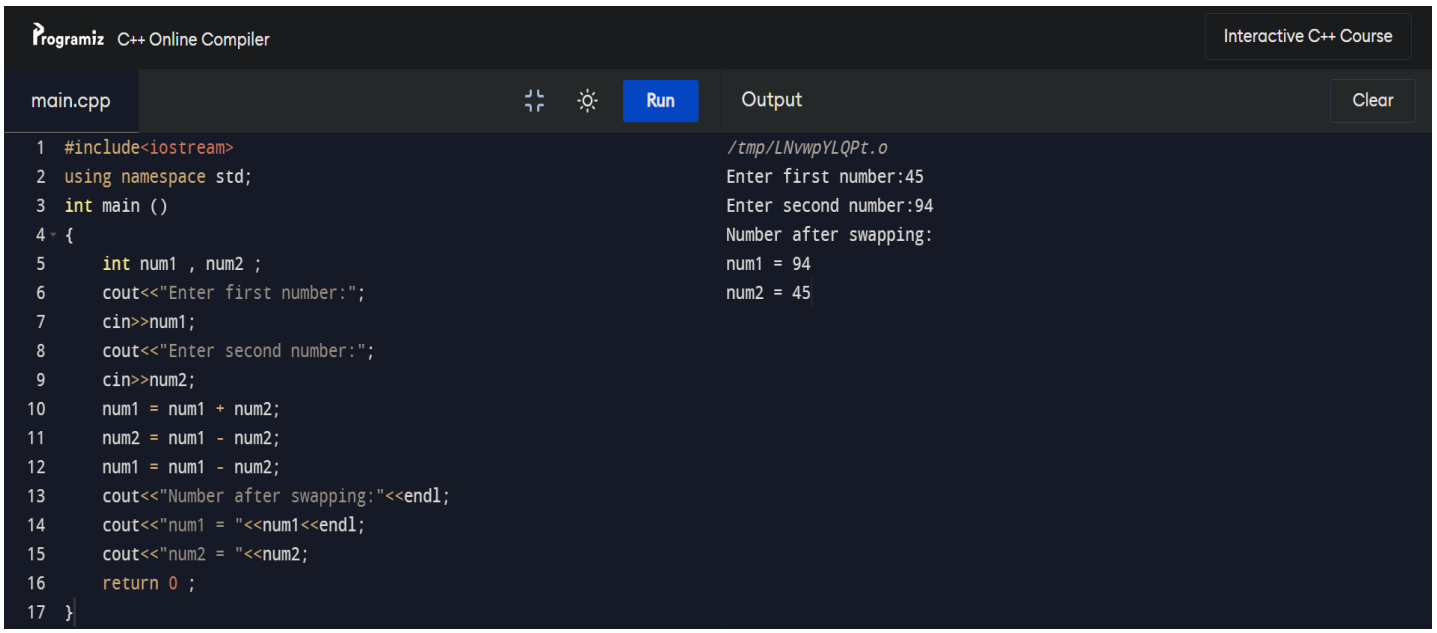
Problem Statement 03

A person is running in a circular ground. Write a program in C++ that asks the user to input the radius of ground in meters and number of rounds the person completes. The program should display amount of distance travelled by the person in meters.

Problem Statement 04

Write a program in C++ that asks the user to enter two integer numbers, stores them in variables num1 and num2 respectively. The program swaps the values of two variables with each other without using a third variable and displays the values of both the variables after swapping.

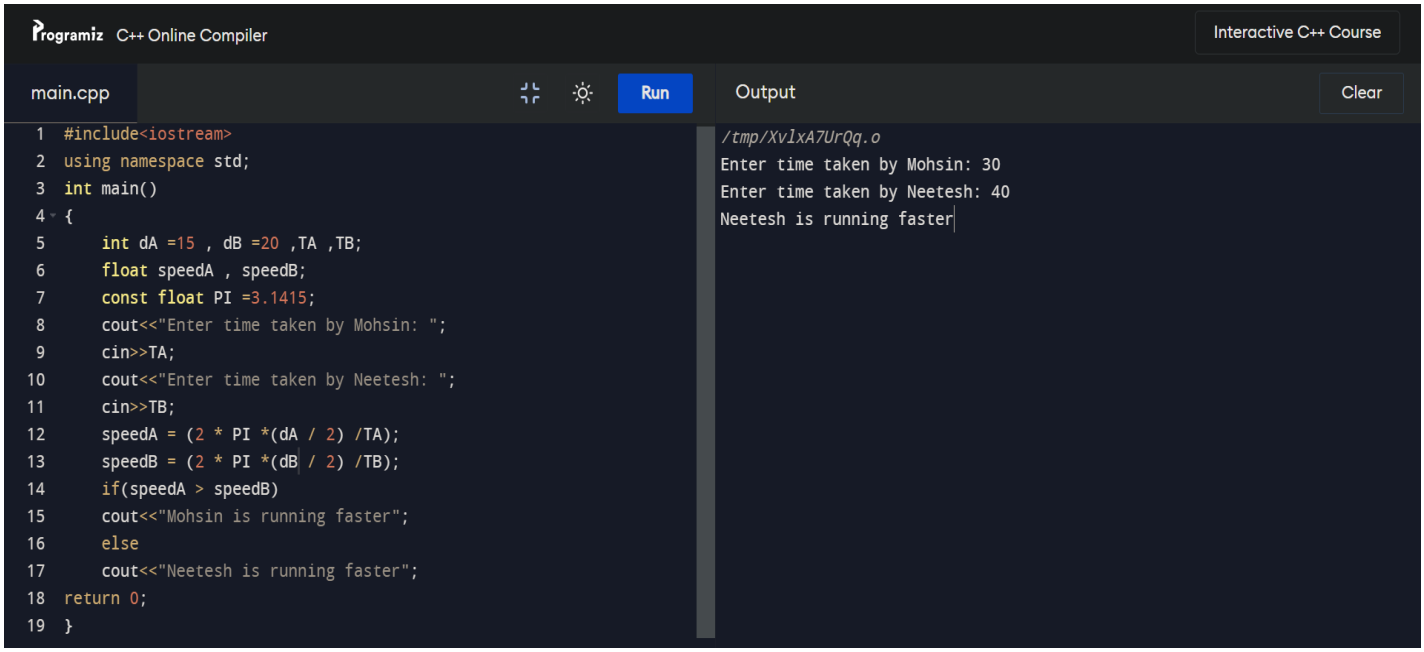
Sample output: (Input: num1 = 45 num1 = 94) (Output: num2 = 94 num2 = 45)



PF Practical # 5

Problem Statement 01

There are two circular grounds Ground-A and Ground-B. Ground-A is having diameter of 15 meters and Ground-B is having diameter of 20 meters. Mohsin is running in Ground-A and Neetesh is running in Ground-B. Write a program in C++ that asks the user to input the time taken, in seconds, to complete one complete round of the ground by both the friends and displays who is running faster.



Problem Statement 02

Write a program in C++ that asks the user to enter three angles of a triangle. The program displays whether the triangle is right-angle, acute-angle or obtuse-angle.

```cpp
#include <iostream>

using namespace std;

int main() {
  // Read three angles of a triangle from the user.
  double angle1, angle2, angle3;
  cout << "Enter three angles of a triangle: ";
  cin >> angle1 >> angle2 >> angle3;

  // Calculate the sum of the three angles.
  double sum = angle1 + angle2 + angle3;

  // Determine the type of triangle.
  if (sum == 180.0) {
    cout << "The triangle is a right-angle triangle.\n";
  } else if (sum < 180.0) {
    cout << "The triangle is an acute-angle triangle.\n";
  } else {
    cout << "The triangle is an obtuse-angle triangle.\n";
  }

  return 0;
}
```

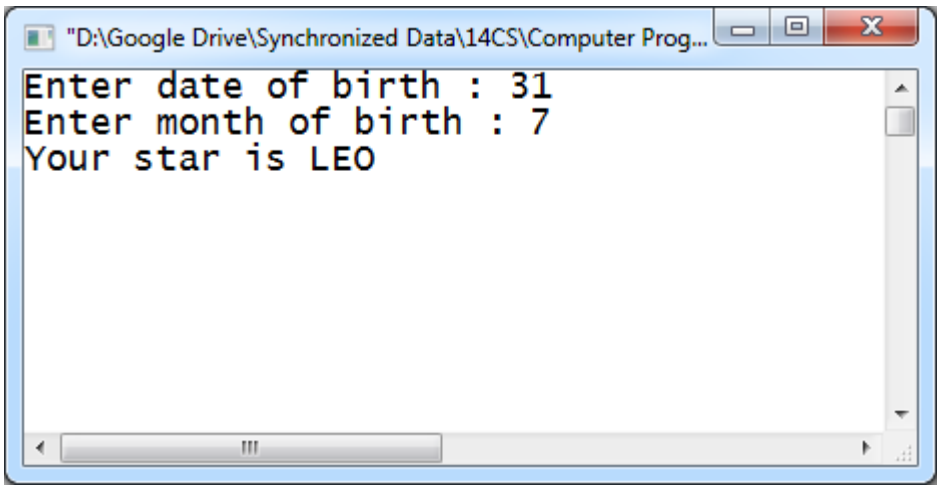
```
Enter three angles of a triangle: 90
40
65
The triangle is an obtuse-angle triangle.

--------------------------------
Process exited after 4.895 seconds with return value 0
Press any key to continue . . .
```

Problem Statement 03

Write a program in C++ that asks the user to enter date of birth and month of birth. The program should display the zodiac star.

Capricorn  22 Dec - 20 Jan	Aquarius  21 Jan -19 Feb	Pisces  20 Feb - 20 Mar	Aries  21 Mar- 19 Apr
Taurus  20 Apr - 20 May	Gemini  21 May - 21 Jun	Cancer  22 Jun - 23 Jul	Leo  24 Jul - 23 Aug
Virgo  24 Aug - 22 Sept	Libra  23 Sept - 22 Oct	Scorpio  23 Oct - 22 Nov	Sagitarius  23 Nov - 20 Dec

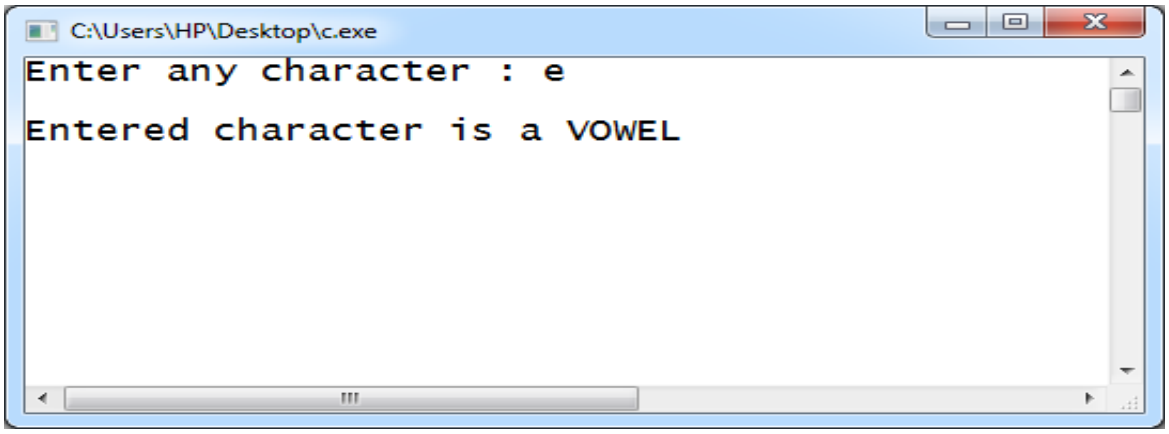
```
#include<iostream>
#include<conio.h>

using namespace std;

int main()
{
    int date, month;

    cout<<"Enter date of birth : ";
    cin>>date;
    cout<<"Enter month of birth : ";
    cin>>month;

    if( (date>=22 && date<=31 && month== 12) || (date>=1 && date<=20 && month==1) )
        cout<<"Your star is CAPRICORN";
    else if( (date>=21 && date<=31 && month==1) || (date>=1 && date<=19 && month==2) )
        cout<<"Your star is AQUARIUS";
    else if( (date>=20 && date<=31 && month==2) || (date>=1 && date<=20 && month==3) )
        cout<<"Your star is PISCES";
    else if( (date>=21 && date<=31 && month==3) || (date>=1 && date<=19 && month==4) )
        cout<<"Your star is ARIES";
    else if( (date>=20 && date<=31 && month==4) || (date>=1 && date<=20 && month==5) )
        cout<<"Your star is TAURUS";

   else if( (date>=21 && date<=31 && month==5) || (date>=1 && date<=21 && month==6) )
       cout<<"Your star is GEMINI";
   else if( (date>=22 && date<=31 && month==6) || (date>=1 && date<=23 && month==7) )
       cout<<"Your star is CANCER";
   else if( (date>=24 && date<=31 && month==7) || (date>=1 && date<=23 && month==8) )
       cout<<"Your star is LEO";
   else if( (date>=24 && date<=31 && month==8) || (date>=1 && date<=22 && month==9) )
       cout<<"Your star is VIRGO";
   else if( (date>=23 && date<=31 && month==9) || (date>=1 && date<=22 && month==10) )
       cout<<"Your star is LIBRA";
   else if( (date>=23 && date<=31 && month==10) || (date>=1 && date<=22 && month==11) )
       cout<<"Your star is SCORPIO";
   else
       cout<<"Your star is SAGITARIUS";

   getch();
   return 0;
}
```

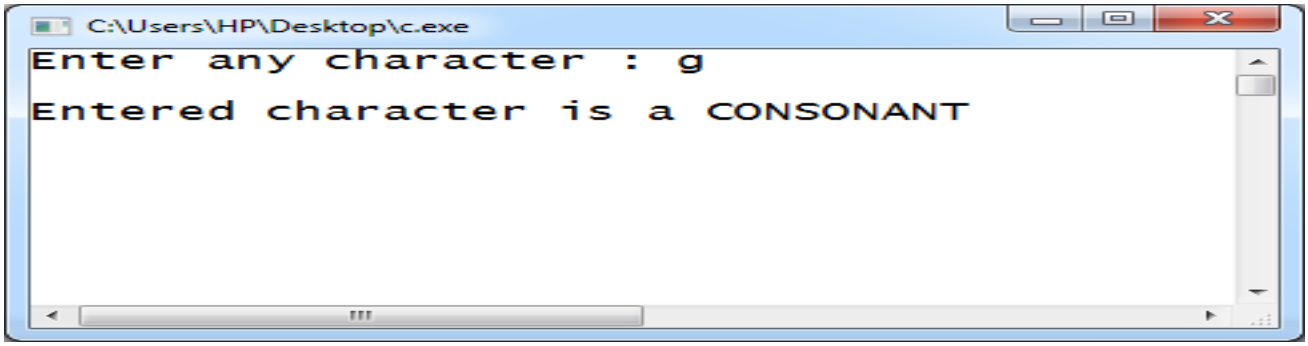


Problem Statement 04

Write a program in C++ that asks the user to enter any character. The program should whether the entered character is a vowel or a consonant.

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

using namespace std;

int main()
{
    char ch;

    cout<<"Enter any character : ";
    ch = getche();

    switch(ch)
    {
        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u':
            cout<<endl<<endl<<"Entered character is a VOWEL";
            break;


        default:
            cout<<endl<<endl<<"Entered character is a CONSONANT";
            break;
    }

    getch();
    return 0;
}
```





Problem Statement 5

Write a program that allows the user to convert a temperature given in degrees from either Celsius to Fahrenheit or Fahrenheit to Celsius. Use the following formulas:

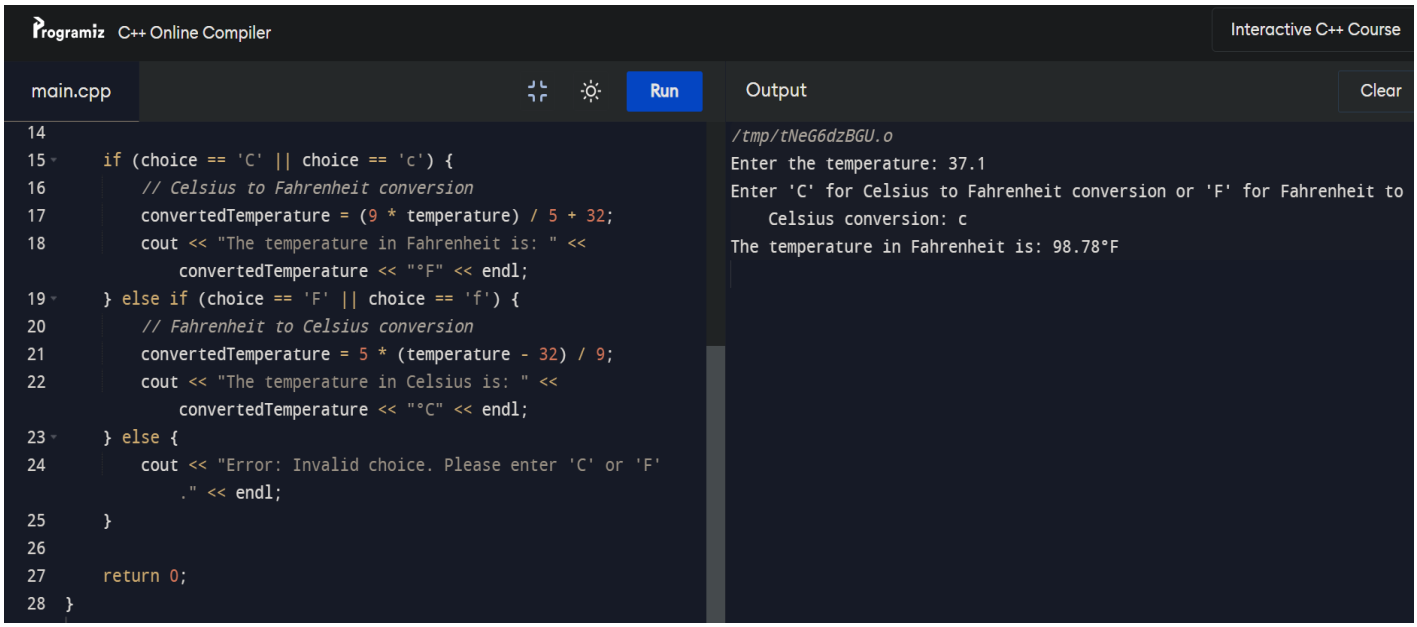
Degrees_C= 5(Degrees_F−32)/9

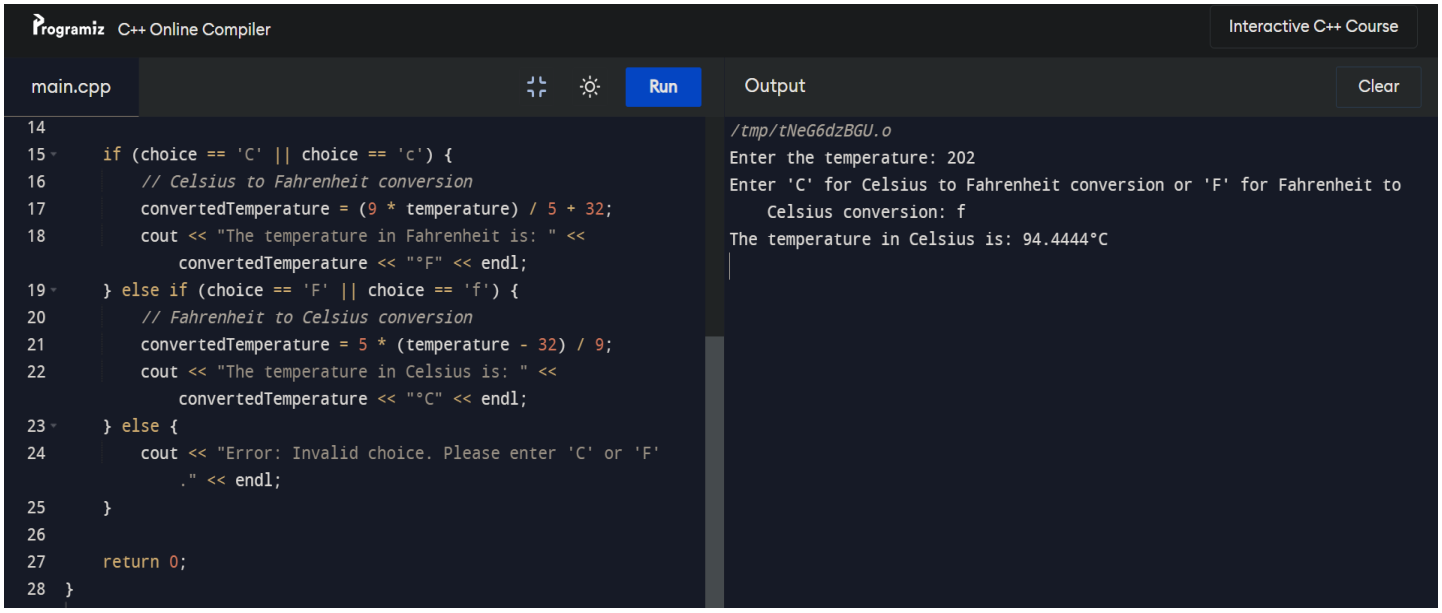
Degrees_F= (9(Degrees_C)/5) + 32)

Prompt the user to enter a temperature and either a C or c for Celsius or an F or f for Fahrenheit.

Convert the temperature to Fahrenheit if Celsius is entered or to Celsius if Fahrenheit is entered.

Display the result in a readable format. If anything other than C, c, F, or f is entered, print error message and stop.




Programiz C++ Online Compiler
Interactive C++ Course
main.cpp
Run
Output
Clear
14
15 if (choice == 'C' || choice == 'c') {
16 // Celsius to Fahrenheit conversion
17 convertedTemperature = (9 * temperature) / 5 + 32;
18 cout << "The temperature in Fahrenheit is: " <<
convertedTemperature << "°F" << endl;
19 } else if (choice == 'F' || choice == 'f') {
20 // Fahrenheit to Celsius conversion
21 convertedTemperature = 5 * (temperature - 32) / 9;
22 cout << "The temperature in Celsius is: " <<
convertedTemperature << "°C" << endl;
23 } else {
24 cout << "Error: Invalid choice. Please enter 'C' or 'F'
." << endl;
25 }
26
27 return 0;
28 }
/tmp/tNeG6dzBGU.o
Enter the temperature: 202
Enter 'C' for Celsius to Fahrenheit conversion or 'F' for Fahrenheit to
Celsius conversion: f
The temperature in Celsius is: 94.4444°C