

NAME: SAFI AHMED

ROLL NO: 22F-BSCS-35

**SUBJECT: PROGRAMMING FUNDAMENTALS
PRACTICAL**

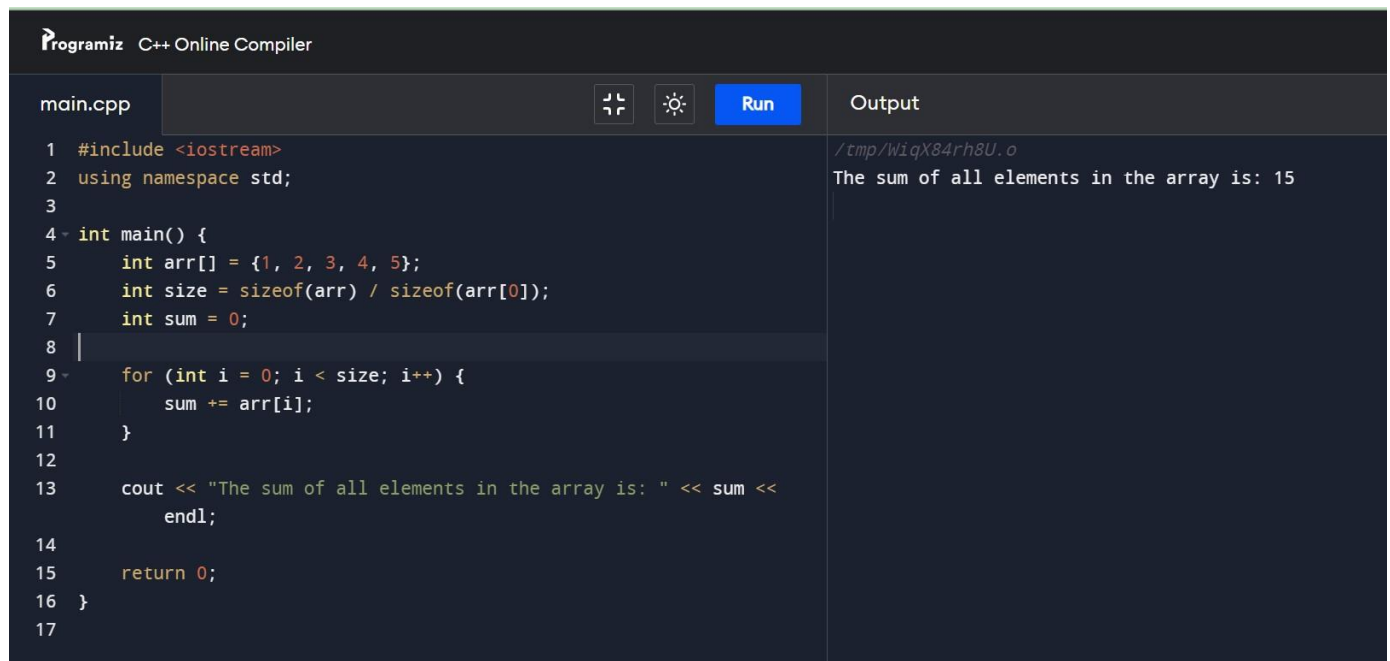
SUBMITTED TO: ENGR. SOFIA HAJANO



Practical 08

PROBLEM STATEMENT 01

Write a C++ program that displays the Sum of all the elements of a one dimensional array.



The screenshot shows a C++ Online Compiler interface. The code in main.cpp is as follows:

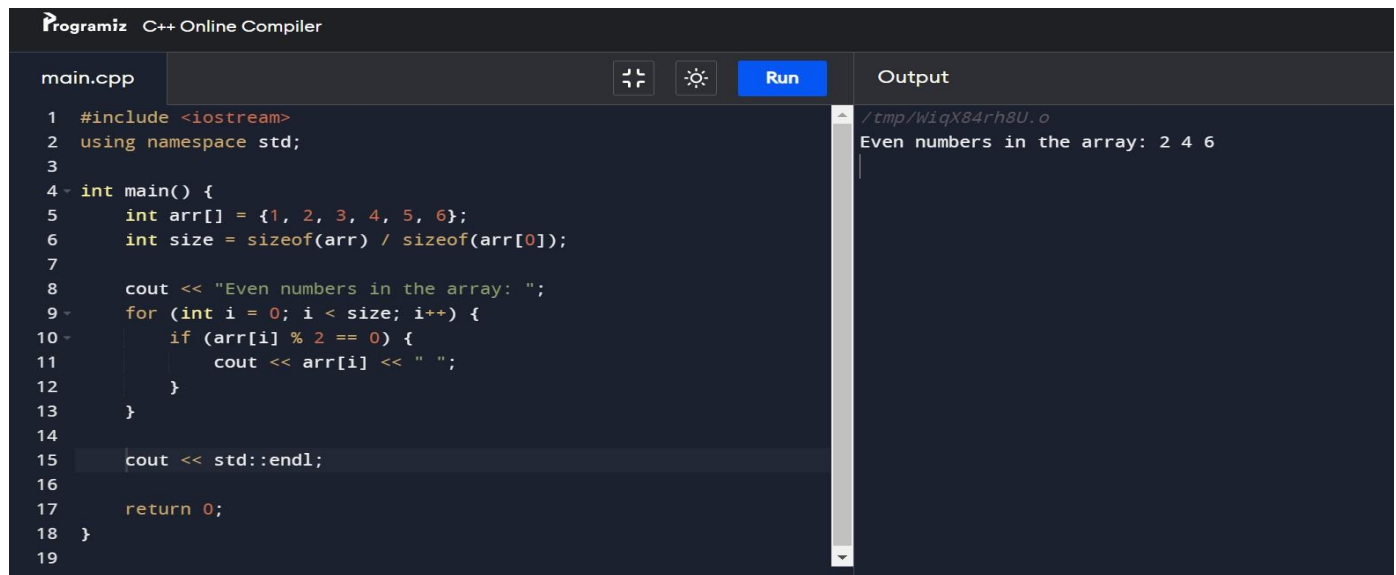
```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int arr[] = {1, 2, 3, 4, 5};
6     int size = sizeof(arr) / sizeof(arr[0]);
7     int sum = 0;
8
9     for (int i = 0; i < size; i++) {
10         sum += arr[i];
11     }
12
13     cout << "The sum of all elements in the array is: " << sum << endl;
14
15     return 0;
16 }
17
```

The output of the program is:

```
/tmp/WiqX84rh8U.o
The sum of all elements in the array is: 15
```

PROBLEM STATEMENT 02

Write a C++ program that displays only the even numbers if present in the one dimensional array.



The screenshot shows a C++ Online Compiler interface. The code in main.cpp is as follows:

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int arr[] = {1, 2, 3, 4, 5, 6};
6     int size = sizeof(arr) / sizeof(arr[0]);
7
8     cout << "Even numbers in the array: ";
9     for (int i = 0; i < size; i++) {
10         if (arr[i] % 2 == 0) {
11             cout << arr[i] << " ";
12         }
13     }
14
15     cout << std::endl;
16
17     return 0;
18 }
19
```

The output of the program is:

```
/tmp/WiqX84rh8U.o
Even numbers in the array: 2 4 6
```

Practical 09

PROBLEM STATEMENT 01

Three sessional test are conducted in the subject of Computer Programming. Each of which having 05 maximum marks. The total sessional marks out of 10, for every student, are calculated by adding two best test marks. Write a program in C++ that inputs the roll number (Just the number not string) and marks of three sessional tests for five students and displays the total marks for each of the student.

	Roll No	Test 01	Test 02	Test 03	Total
Student 1	01	4	5	3	9
Student 2	03	5	5	4	10
Student 3	05	2	4	3	7
Student 4	07	0	4	1	5
Student 5	09	3	3	2	6

INPUT

```
#include <iostream> using namespace std; int main() {          const int
Rows=5;      const int Cols=5;      int marks[ Rows] [ Cols] =
{{1,4,5,3,0} , {3,5,5,4,0} ,{5,2,4,3,0} ,
{7,0,4,1,0} , {9,3,3,2,0}};
    for (size_t i = 0; i < Rows; i++)
    {
        int sumOneTwo  = marks[i][1] + marks[i][2];
int sumOneThree= marks[i][1] + marks[i][3];      int sumTwoThree=
marks[i][2] + marks[i][3];      if (sumOneTwo >= sumOneThree &&
sumOneTwo >=sumTwoThree)      marks[i][4] = sumOneTwo;
else if (sumOneThree >= sumOneTwo && sumOneThree >=sumTwoThree)
marks[i][4] = sumOneThree;      else marks[i][4] = sumTwoThree;
    }      for (size_t j = 0; j < Rows; j++)

    {
```

```

        cout << " Sessional marks of Roll number " << marks [j][0] << " = " <<
marks [j][4] << endl;
    }    return 0;
}

```

OUTPUT

```

PS E:\YouTube\codewithhrrynotes> cd "e:\YouTube\codewithhrrynotes\" ; if ($?)
{ g++ assignment.cpp -o assignment } ; if ($?) { .\assignment }
Sessional marks of Roll number 1 = 9
Sessional marks of Roll number 3 = 10
Sessional marks of Roll number 5 = 7
Sessional marks of Roll number 7 = 5
Sessional marks of Roll number 9 = 6

```

PROBLEM STATEMENT 02

Write a program in C++ that initializes matrix A and B. The program should generate the matrix C which is given as:

$$C = (A + B^T)^T$$

Whereas,

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad B = \begin{bmatrix} 3 & 1 & 9 \\ 7 & 8 & 0 \\ 5 & 3 & 6 \end{bmatrix}$$

Programiz C++ Online Compiler

main.cpp

Run

```
1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5
6 // Function to initialize a matrix
7 void initializeMatrix(vector<vector<int>>& matrix, const vector
  <vector<int>>& values) {
8     for (int i = 0; i < matrix.size(); i++) {
9         for (int j = 0; j < matrix[0].size(); j++) {
10             matrix[i][j] = values[i][j];
11         }
12     }
13 }
14
15 // Function to print a matrix
16 void printMatrix(const vector<vector<int>>& matrix) {
17     for (int i = 0; i < matrix.size(); i++) {
18         for (int j = 0; j < matrix[0].size(); j++) {
19             cout << matrix[i][j] << " ";
20         }
21         cout << endl;
22     }
23 }
24
```

/tmp/2REa38gVLZ.o

4 9 8
5 13 9
16 8 15

Programiz C++ Online Compiler

main.cpp

Run

```
25 int main() {
26     // Initialize matrix A
27     vector<vector<int>> A = {{1, 2, 3},
28                             {4, 5, 6},
29                             {7, 8, 9}};
30
31     // Initialize matrix B
32     vector<vector<int>> B = {{3, 1, 9},
33                             {7, 8, 0},
34                             {5, 3, 6}};
35
36     // Calculate matrix C
37     vector<vector<int>> C(A.size(), vector<int>(A[0].size()));
38     for (int i = 0; i < A.size(); i++) {
39         for (int j = 0; j < A[0].size(); j++) {
40             C[i][j] = A[i][j] + B[j][i]; // B^T
41         }
42     }
43
44     // Print matrix C
45     printMatrix(C);
46
47     return 0;
48 }
49
```

/tmp/2REa38gVLZ.o

4 9 8
5 13 9
16 8 15

Practical 10

PROBLEM STATEMENT 01

Write a program in C++ that creates the following functions:

Function	Description
<code>int vowelCount(string str)</code>	Receives string object and returns number of vowels in it.
<code>int consonantCount(string str)</code>	Receives string object and returns number of consonants in it.
<code>int upperCount(string str)</code>	Receives string object and returns number of uppercase letters in it.
<code>int lowerCount(string str)</code>	Receives string object and returns number of lowercase letters in it.

Use all of the functions in main function.

INPUT

```

#include <iostream> #include <string> using
namespace std; int countVowels(const string& str)
{
    int count
= 0;    string vowels = "aeiouAEIOU";    for
(char c : str) {        if (vowels.find(c) !=
string::npos)            count++;
    }    return count;
} int countConsonants(const string& str) {    int count = 0;
string vowels = "aeiouAEIOU";    for (char c : str) {        if
(isalpha(c) && vowels.find(c) == string::npos)
count++;
    }    return count;
} int countUppercase(const string& str) {
int count = 0;    for (char c : str) {
if (isupper(c))        count++;
    }    return count;
} int countLowercase(const string& str) {
int count = 0;    for (char c : str) {
if (islower(c))        count++;
    }    return count;
}

int main() {    string input;    cout << "Enter a string: ";
getline(cin, input);    cout << "Number of vowels: " <<
countVowels(input)<<endl;    cout << "Number of consonants: " <<
countConsonants(input)<<endl;    cout << "Number of uppercase letters:
" << countUppercase(input)<<endl;    cout << "Number of lowercase
letters: " << countLowercase(input)<<endl;    return 0;
}

```

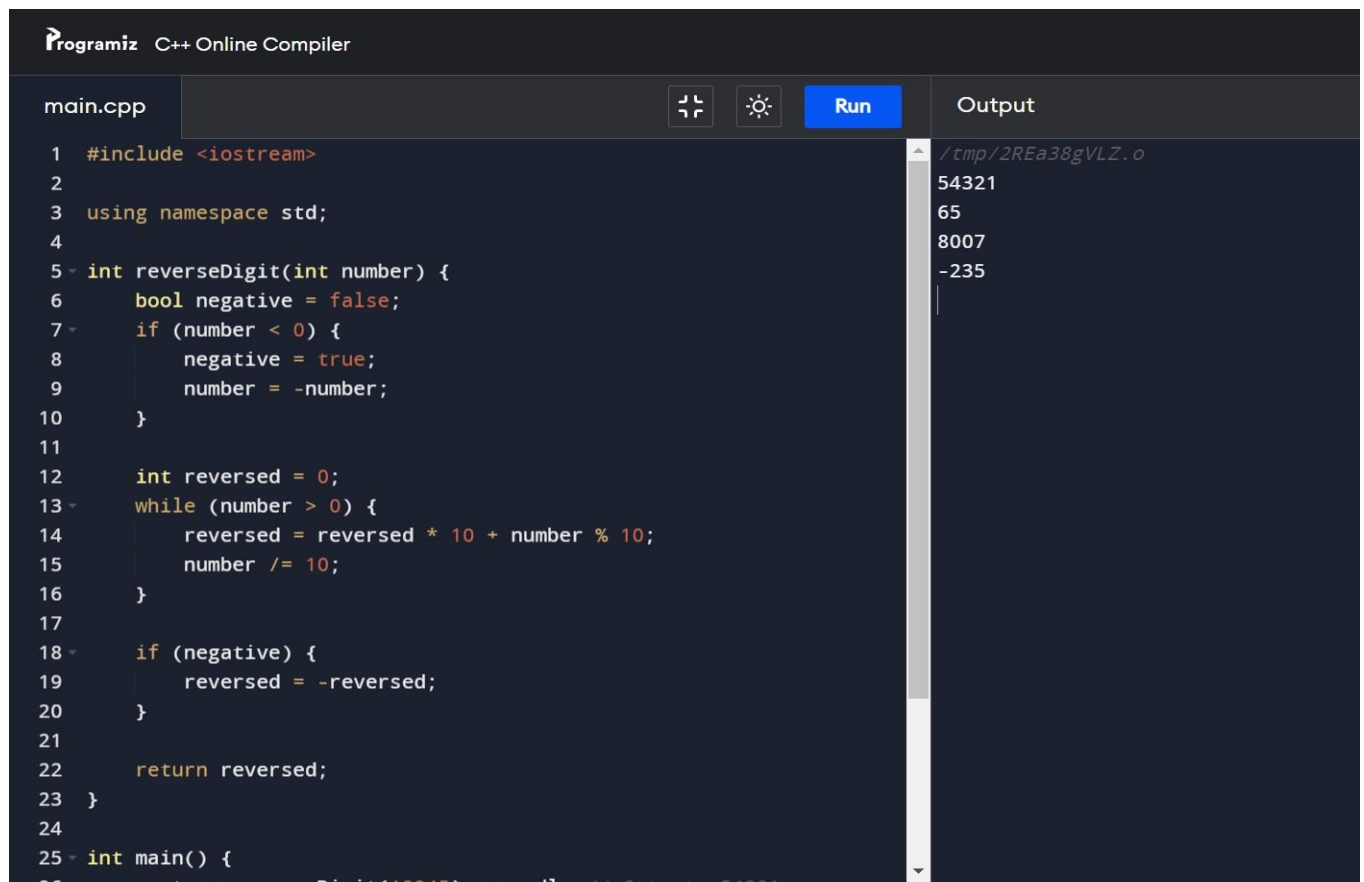
OUTPUT

```
PS E:\YouTube\codewithhrrynotes> cd "e:\YouTube\codewithhrrynotes\" ; if ($?) { g++ assignment.cpp -o assignment } ; if ($?) { .\assignment }
Enter a string: Safi Ahmed
Number of vowels: 4
Number of consonants: 5
Number of uppercase letters: 2
Number of lowercase letters: 7
```

PROBLEM STATEMENT 02

Write a function, reverse Digit that takes an integer as a parameter and returns the number with its digits reversed.

For example, the value of reverseDigit(12345) is 54321; the value of reverseDigit(5600) is 65; the value of reverseDigit(7008) is 8007; and the value of reverseDigit(-532) is -235.



The screenshot shows a C++ Online Compiler interface with the following components:

- Header:** Programiz C++ Online Compiler
- File Tab:** main.cpp
- Code Editor:** Contains the implementation of the reverseDigit function and a main function for testing.
- Buttons:** Run (blue), and icons for full screen and settings.
- Output Panel:** Displays the results of the program execution.

Code Implementation:

```
1 #include <iostream>
2
3 using namespace std;
4
5 int reverseDigit(int number) {
6     bool negative = false;
7     if (number < 0) {
8         negative = true;
9         number = -number;
10    }
11
12    int reversed = 0;
13    while (number > 0) {
14        reversed = reversed * 10 + number % 10;
15        number /= 10;
16    }
17
18    if (negative) {
19        reversed = -reversed;
20    }
21
22    return reversed;
23 }
24
25 int main() {
```

Output:

```
/tmp/2REa38gVLZ.o
54321
65
8007
-235
```



```
Programiz C++ Online Compiler
main.cpp
~
9      negative = true;
10     number = -number;
11 }
12
13 int reversed = 0;
14 while (number > 0) {
15     reversed = reversed * 10 + number % 10;
16     number /= 10;
17 }
18
19 if (negative) {
20     reversed = -reversed;
21 }
22
23 return reversed;
24 }
25
26 int main() {
27     cout << reverseDigit(12345) << endl; // Output: 54321
28     cout << reverseDigit(5600) << endl; // Output: 65
29     cout << reverseDigit(7008) << endl; // Output: 8007
30     cout << reverseDigit(-532) << endl; // Output: -235
31 }
32
33 }
```

Output

```
/tmp/2REa38gVLZ.o
54321
65
8007
-235
```

Practical 11

PROBLEM STATEMENT 01

Write a recursive function `power (base, exponent)` that, when invoked, returns

$base^{exponent}$

For example, $power(3, 4) = 3 * 3 * 3 * 3$. Assume that exponent is an integer greater than or equal to 1. Hint: The recursion step would use the relationship.

$base^{exponent} = base * base^{exponent - 1}$

And the terminating condition occurs when exponent is equal to 1, because $base$

$^1 = base$.

```
Programiz C++ Online Compiler

main.cpp Run Output

1 #include <iostream>
2
3 using namespace std;
4
5 int power(int base, int exponent) {
6     // Base case: exponent is 1
7     if (exponent == 1) {
8         return base;
9     }
10
11     // Recursive case: exponent > 1
12     return base * power(base, exponent - 1);
13 }
14
15 int main() {
16     cout << power(3, 4) << endl; // Output: 81
17     cout << power(2, 5) << endl; // Output: 32
18     cout << power(5, 3) << endl; // Output: 125
19
20     return 0;
21 }
22
```

PROBLEM STATEMENT 02

The Fibonacci series

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

Begins with the terms 0 and 1 and has the property that each succeeding term is the sum of the two preceding terms. Write a recursive function that generates N series numbers. Where N is the integer inputted by the user at runtime.

Programiz C++ Online Compiler

C++ C

main.cpp

Run

```
1 #include <iostream>
2
3 using namespace std;
4
5 void generateFibonacci(int N, int first = 0, int second = 1) {
6     if (N <= 0) {
7         return;
8     }
9
10    cout << first << " ";
11    generateFibonacci(N - 1, second, first + second);
12 }
13
14 int main() {
15     int N;
16     cout << "Enter the number of Fibonacci series numbers to generate
17         : ";
18     cin >> N;
19     generateFibonacci(N);
20
21     return 0;
22 }
```

Output

/tmp/2REa38gVLZ.o

Enter the number of Fibonacci series numbers to generate: 10

0 1 1 2 3 5 8 13 21 34

The Fibonacci series is 0-indexed, so the first number is considered to be 0, the second number is 1, and so on.

PROBLEM STATEMENT 03

Write a recursive algorithm to multiply two positive integers m and n using repeated addition. Specify the base case and the recursive case.

```
Programiz C++ Online Compiler

main.cpp Run Output

1 #include <iostream>
2
3 using namespace std;
4
5 int multiply(int m, int n) {
6     // Base case: n is 0
7     if (n == 0) {
8         return 0;
9     }
10
11     // Recursive case: n > 0
12     return m + multiply(m, n - 1);
13 }
14
15 int main() {
16     int m, n;
17     cout << "Enter two positive integers: ";
18     cin >> m >> n;
19
20     cout << "Product: " << multiply(m, n) << endl;
21
22     return 0;
23 }
24
```

/tmp/2REa38gVLZ.o
Enter two positive integers: 4 2
Product: 8

Practical 12

PROBLEM STATEMENT 01

Create a structure called volume that uses three variables of type Distance (structure) to model the volume of a room. Initialize a variable of type volume to specific dimensions, and then calculate the volume it represent and print out the result.

main.cpp



Run

Output

```
1  #include <iostream>
2
3  using namespace std;
4
5  // Structure to represent distance
6  struct Distance {
7      double feet;
8      double inches;
9  };
10
11 // Structure to represent volume
12 struct Volume {
13     Distance length;
14     Distance width;
15     Distance height;
16 };
17
18 // Function to calculate the volume
19 double calculateVolume(const Volume& room) {
20     double lengthInInches = room.length.feet * 12 + room.length
        .inches;
21     double widthInInches = room.width.feet * 12 + room.width.inches
        ;
22     double heightInInches = room.height.feet * 12 + room.height
        .inches;
23
24 }
```

/tmp/2REa38gVLZ.o

The volume of the room is: 1125.47 cubic feet

main.cpp



Run

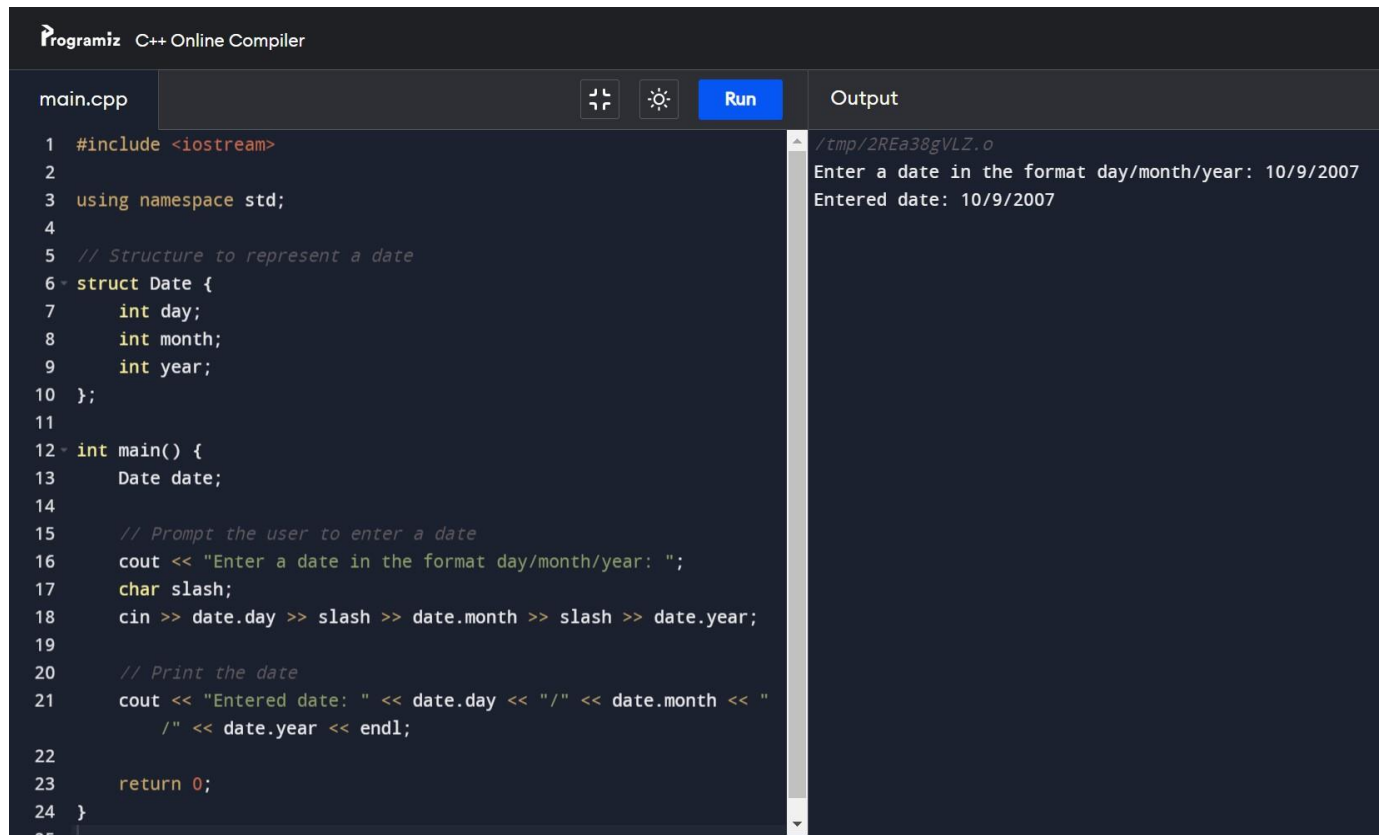
Output

```
24     double volumeInCubicFeet = lengthInInches * widthInInches *
        heightInInches / 12 / 12 / 12;
25
26     return volumeInCubicFeet;
27 }
28
29 int main() {
30     // Initialize a variable of type Volume
31     Volume room;
32     room.length.feet = 10;
33     room.length.inches = 6;
34     room.width.feet = 12;
35     room.width.inches = 3;
36     room.height.feet = 8;
37     room.height.inches = 9;
38
39     // Calculate the volume of the room
40     double volume = calculateVolume(room);
41
42     // Print the result
43     cout << "The volume of the room is: " << volume << " cubic
        feet" << endl;
44
45     return 0;
46 }
47 }
```

/tmp/2REa38gVLZ.o

The volume of the room is: 1125.47 cubic feet

Create a structure a type date that contains three members: the day, the month, the year, all of type int. Here the user enter a date in the format 10/9/2007, store it in a variable of type date, then retrieve the value form the variable and print them out in the same format.



The screenshot shows a C++ Online Compiler interface. The code in `main.cpp` defines a `Date` struct with `int` members for `day`, `month`, and `year`. The `main` function prompts the user to enter a date in the format `day/month/year`, reads the input into the `Date` struct, and then prints the date in the same format. The output shows the user entering `10/9/2007` and the program printing `Entered date: 10/9/2007`.

```
1 #include <iostream>
2
3 using namespace std;
4
5 // Structure to represent a date
6 struct Date {
7     int day;
8     int month;
9     int year;
10 };
11
12 int main() {
13     Date date;
14
15     // Prompt the user to enter a date
16     cout << "Enter a date in the format day/month/year: ";
17     char slash;
18     cin >> date.day >> slash >> date.month >> slash >> date.year;
19
20     // Print the date
21     cout << "Entered date: " << date.day << "/" << date.month << "
22         << date.year << endl;
23
24     return 0;
25 }
```

Output:

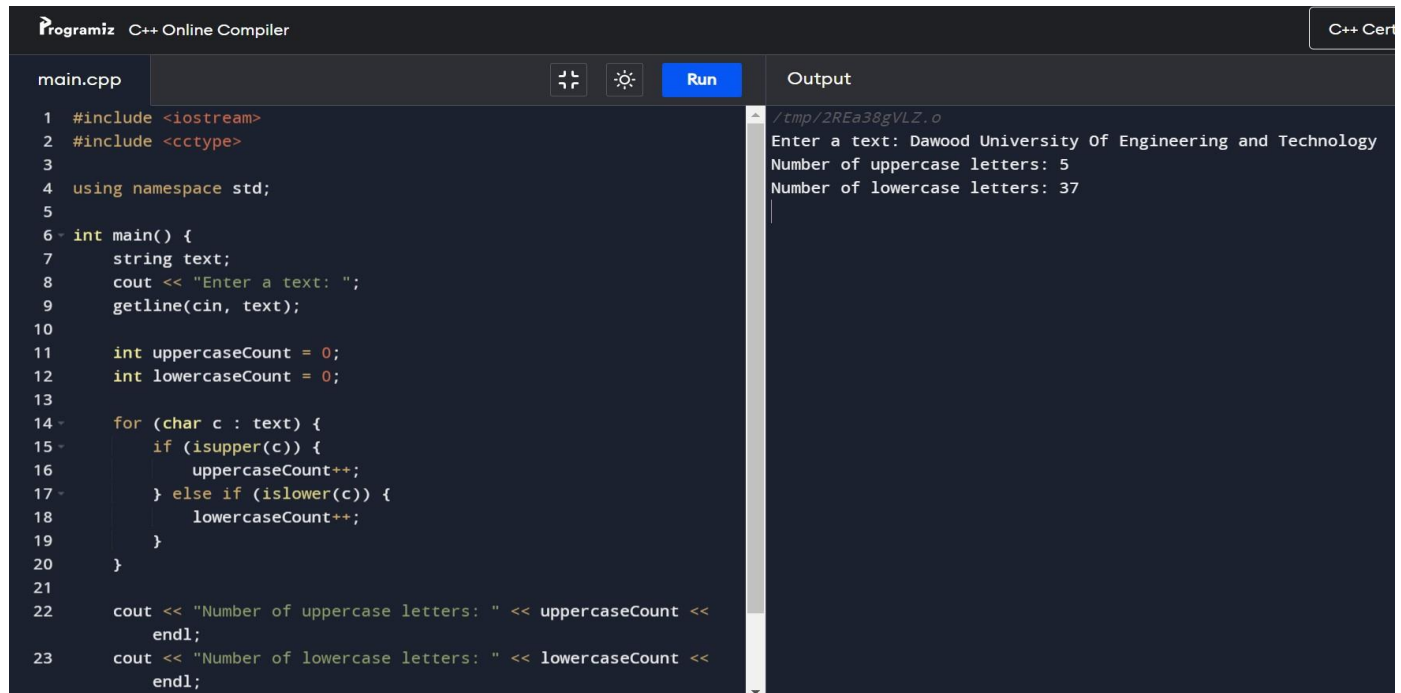
```
/tmp/2REa38gVLZ.o
Enter a date in the format day/month/year: 10/9/2007
Entered date: 10/9/2007
```

Practical 13

PROBLEM STATEMENT 01

Write a program in C++ that inputs a text from the user and displays the number of uppercase letters and lowercase letters in it.

PROBLEM STATEMENT 0

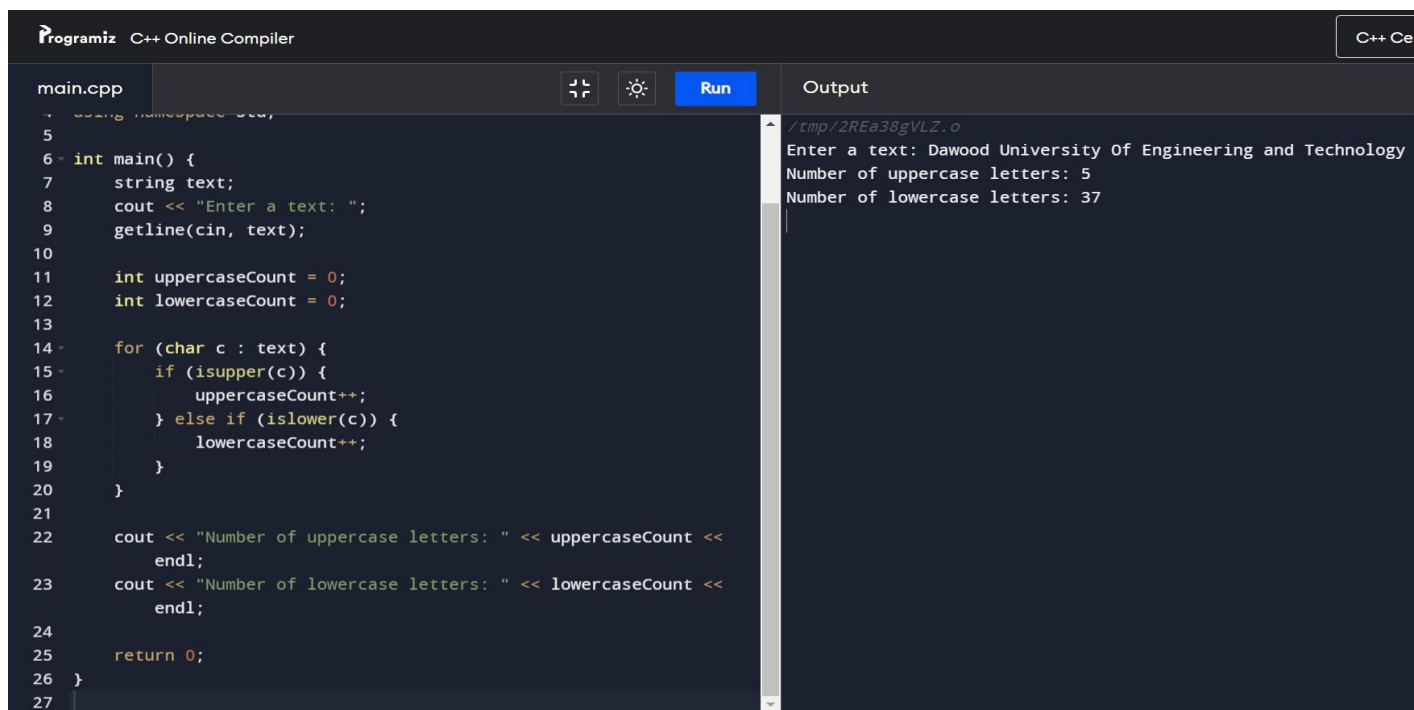


The screenshot shows the Programiz C++ Online Compiler interface. The code in `main.cpp` is as follows:

```
1 #include <iostream>
2 #include <cctype>
3
4 using namespace std;
5
6 int main() {
7     string text;
8     cout << "Enter a text: ";
9     getline(cin, text);
10
11     int uppercaseCount = 0;
12     int lowercaseCount = 0;
13
14     for (char c : text) {
15         if (isupper(c)) {
16             uppercaseCount++;
17         } else if (islower(c)) {
18             lowercaseCount++;
19         }
20     }
21
22     cout << "Number of uppercase letters: " << uppercaseCount << endl;
23     cout << "Number of lowercase letters: " << lowercaseCount << endl;
24 }
```

The output of the program is shown in the Output panel:

```
/tmp/2REa38gVLZ.o
Enter a text: Dawood University Of Engineering and Technology
Number of uppercase letters: 5
Number of lowercase letters: 37
```



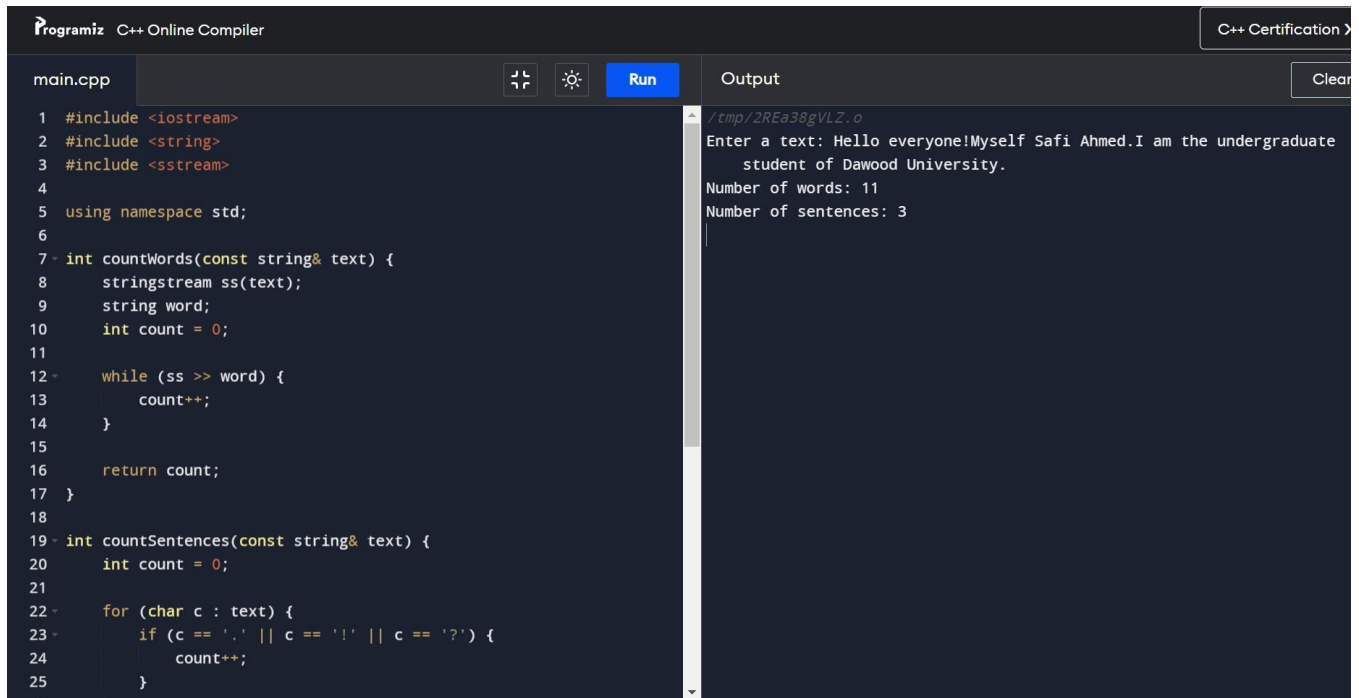
The screenshot shows the Programiz C++ Online Compiler interface. The code in `main.cpp` is as follows:

```
1 using namespace std;
2
3
4
5
6 int main() {
7     string text;
8     cout << "Enter a text: ";
9     getline(cin, text);
10
11     int uppercaseCount = 0;
12     int lowercaseCount = 0;
13
14     for (char c : text) {
15         if (isupper(c)) {
16             uppercaseCount++;
17         } else if (islower(c)) {
18             lowercaseCount++;
19         }
20     }
21
22     cout << "Number of uppercase letters: " << uppercaseCount << endl;
23     cout << "Number of lowercase letters: " << lowercaseCount << endl;
24
25     return 0;
26 }
27
```

The output of the program is shown in the Output panel:

```
/tmp/2REa38gVLZ.o
Enter a text: Dawood University Of Engineering and Technology
Number of uppercase letters: 5
Number of lowercase letters: 37
```

Write a program in C++ that inputs a text from the user and displays the number of words and sentences in it.

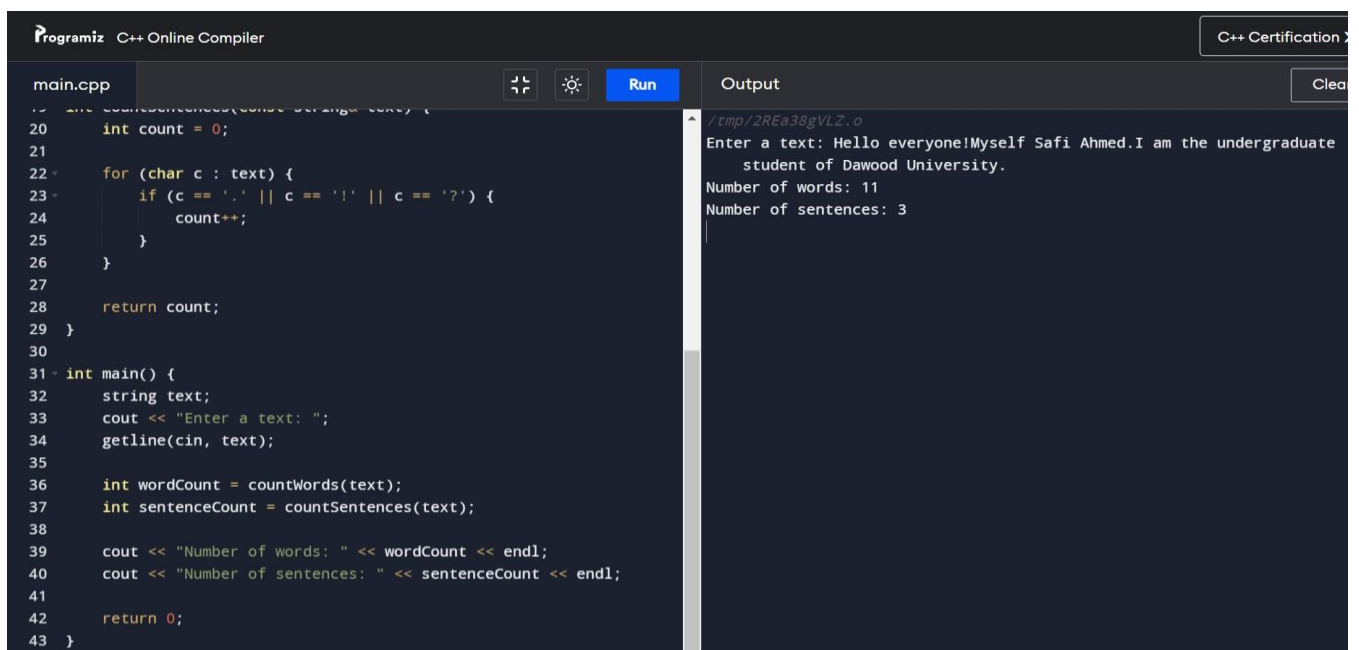


The screenshot shows the Programiz C++ Online Compiler interface. The code in `main.cpp` defines two functions: `countWords` and `countSentences`. `countWords` uses a `stringstream` to split the input text by spaces and counts the resulting words. `countSentences` iterates through the text, counting periods, exclamation marks, and question marks as sentence delimiters. The `Run` button is highlighted in blue. The output on the right shows the program's execution with the input text "Hello everyone!Myself Safi Ahmed.I am the undergraduate student of Dawood University." and the resulting counts: 11 words and 3 sentences.

```
1 #include <iostream>
2 #include <string>
3 #include <sstream>
4
5 using namespace std;
6
7 int countWords(const string& text) {
8     stringstream ss(text);
9     string word;
10    int count = 0;
11
12    while (ss >> word) {
13        count++;
14    }
15
16    return count;
17 }
18
19 int countSentences(const string& text) {
20     int count = 0;
21
22     for (char c : text) {
23         if (c == '.' || c == '!' || c == '?') {
24             count++;
25         }
26     }
27 }
```

Output

```
/tmp/2REa38gVLZ.o
Enter a text: Hello everyone!Myself Safi Ahmed.I am the undergraduate
student of Dawood University.
Number of words: 11
Number of sentences: 3
```



This screenshot shows the complete C++ program, including the `main` function. The `main` function prompts the user to enter a text, reads the input using `getline`, and then calls `countWords` and `countSentences` to calculate the word and sentence counts. The results are displayed using `cout`. The `Run` button is highlighted in blue. The output on the right is identical to the first screenshot, showing the input text and the calculated counts of 11 words and 3 sentences.

```
20 int countSentences(const string& text) {
21     int count = 0;
22
23     for (char c : text) {
24         if (c == '.' || c == '!' || c == '?') {
25             count++;
26         }
27     }
28
29     return count;
30 }
31
32 int main() {
33     string text;
34     cout << "Enter a text: ";
35     getline(cin, text);
36
37     int wordCount = countWords(text);
38     int sentenceCount = countSentences(text);
39
40     cout << "Number of words: " << wordCount << endl;
41     cout << "Number of sentences: " << sentenceCount << endl;
42
43     return 0;
44 }
```

Output

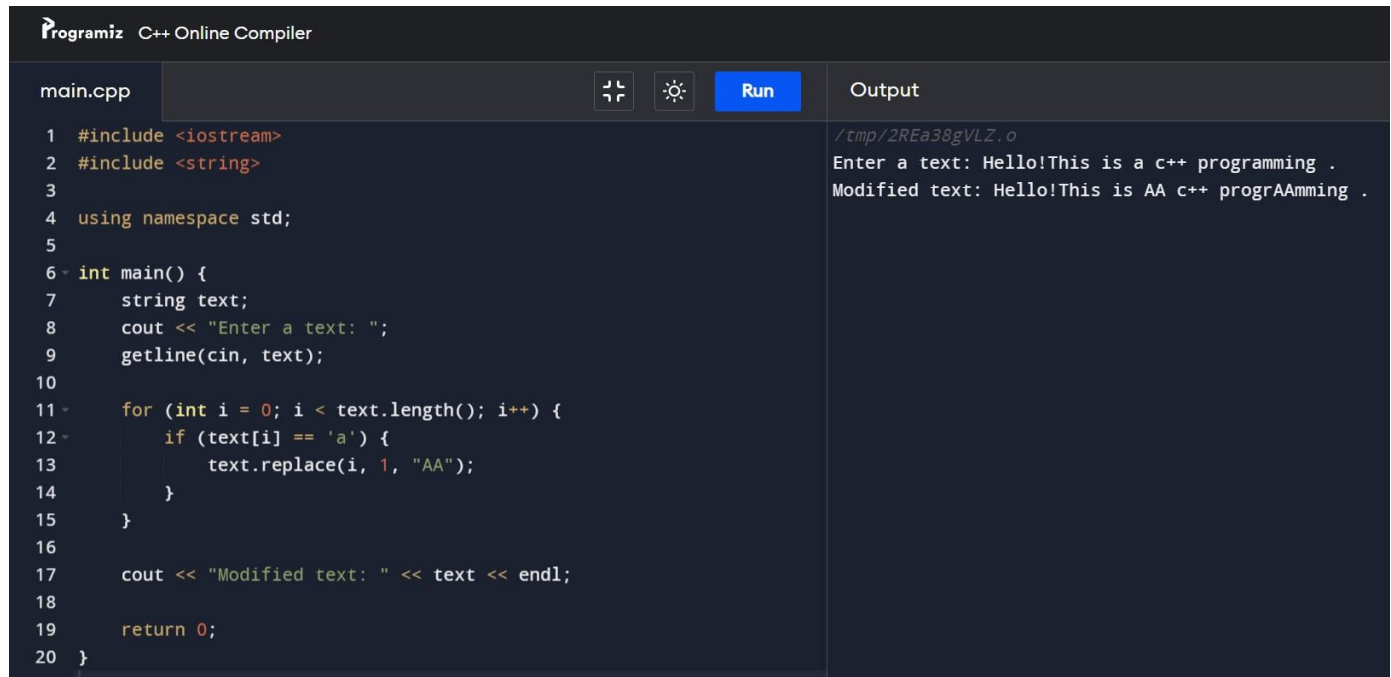
```
/tmp/2REa38gVLZ.o
Enter a text: Hello everyone!Myself Safi Ahmed.I am the undergraduate
student of Dawood University.
Number of words: 11
Number of sentences: 3
```


PROBLEM STATEMENT 0

Practical 14

PROBLEM STATEMENT 01

Write a program in C++ that inputs a text from the user and replaces all the occurrences of lowercase letter 'a' to "AA".



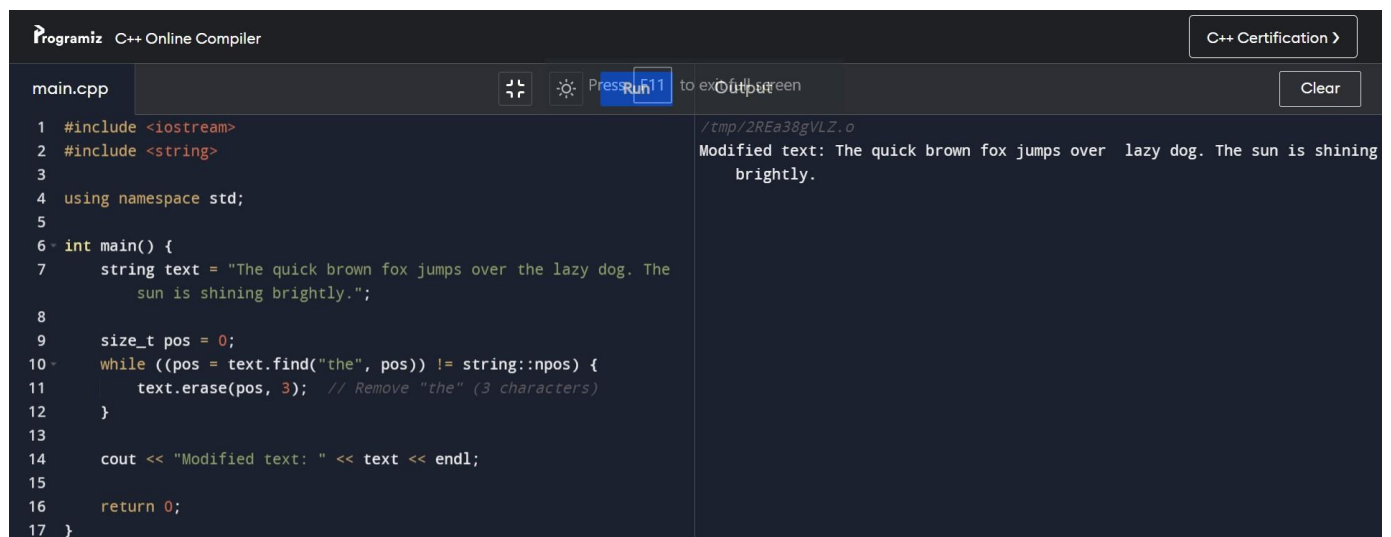
```
main.cpp
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     string text;
8     cout << "Enter a text: ";
9     getline(cin, text);
10
11     for (int i = 0; i < text.length(); i++) {
12         if (text[i] == 'a') {
13             text.replace(i, 1, "AA");
14         }
15     }
16
17     cout << "Modified text: " << text << endl;
18
19     return 0;
20 }
```

Output

```
/tmp/2REa38gVLZ.o
Enter a text: Hello!This is a c++ programming .
Modified text: Hello!This is AA c++ progrAAming .
```

PROBLEM STATEMENT 02

Write a program in C++ that initializes a string object and deletes all the occurrences the word "the" in it.



```
main.cpp
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     string text = "The quick brown fox jumps over the lazy dog. The sun is shining brightly.";
8
9     size_t pos = 0;
10    while ((pos = text.find("the", pos)) != string::npos) {
11        text.erase(pos, 3); // Remove "the" (3 characters)
12    }
13
14    cout << "Modified text: " << text << endl;
15
16    return 0;
17 }
```

Output

```
/tmp/2REa38gVLZ.o
Modified text: The quick brown fox jumps over lazy dog. The sun is shining brightly.
```

Write a program in C++ that inputs three names and displays the name which comes first among them in alphabetical order.

```
Programiz C++ Online Compiler

main.cpp Run Output

1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     string name1, name2, name3;
8
9     cout << "Enter the first name: ";
10    cin >> name1;
11
12    cout << "Enter the second name: ";
13    cin >> name2;
14
15    cout << "Enter the third name: ";
16    cin >> name3;
17
18    string firstAlphabeticalName = name1;
19
20    if (name2 < firstAlphabeticalName) {
21        firstAlphabeticalName = name2;
22    }
23
24    if (name3 < firstAlphabeticalName) {
25        firstAlphabeticalName = name3;
26    }
27
28    cout << "The name that comes first in alphabetical order: " <<
    firstAlphabeticalName << endl;
29
30    return 0;
31 }
```

/tmp/2REa38gVLZ.o
Enter the first name: SAFI
Enter the second name: RAFAY
Enter the third name: YOUSUF
The name that comes first in alphabetical order: RAFAY

```
Programiz C++ Online Compiler

main.cpp Run Output


9     cout << "Enter the first name: ";
10    cin >> name1;
11
12    cout << "Enter the second name: ";
13    cin >> name2;
14
15    cout << "Enter the third name: ";
16    cin >> name3;
17
18    string firstAlphabeticalName = name1;
19
20    if (name2 < firstAlphabeticalName) {
21        firstAlphabeticalName = name2;
22    }
23
24    if (name3 < firstAlphabeticalName) {
25        firstAlphabeticalName = name3;
26    }
27
28    cout << "The name that comes first in alphabetical order: " <<
    firstAlphabeticalName << endl;
29
30    return 0;
31 }
```

/tmp/2REa38gVLZ.o
Enter the first name: SAFI
Enter the second name: RAFAY
Enter the third name: YOUSUF
The name that comes first in alphabetical order: RAFAY




PROBLEM STATEMENT 04

Write a program in C++ that input your name and displays it in reverse order.

PROBLEM STATEMENT 0

 C++ Online Compiler

main.cpp



```
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     string name;
8
9     cout << "Enter your name: ";
10    getline(cin, name);
11
12    cout << "Your name in reverse order: ";
13    for (int i = name.length() - 1; i >= 0; i--) {
14        cout << name[i];
15    }
16
17    cout << endl;
18
19    return 0;
20 }
21
```

Output

/tmp/2REa38gVLZ.o
Enter your name: SAFI AHMED
Your name in reverse order: DEMHA IFAS