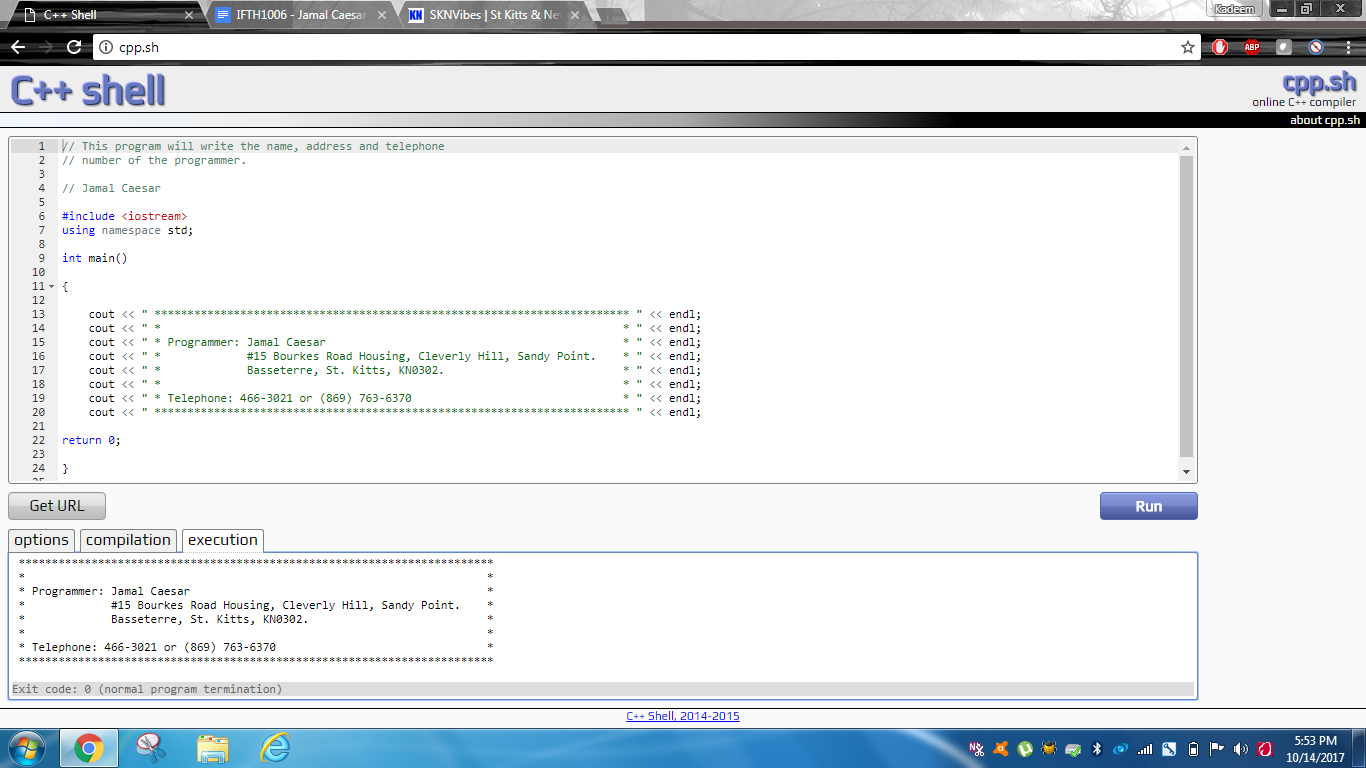
# Lab 2.

## Fill in the blank Questions

1. A **constant**  is a memory location whose value cannot change during the execution of the program.
2. **Floating point** is a data type that only holds numbers with no fractional component.
3. **Integer**  is a data type that holds numbers with fractional components.
4. is an arithmetic operator that gives the remainder of a division problem.
5. Cout << is an example of the **Output/iostream** fundamental instruction.
6. **Boolean** data types only have two values: true and false.
7. One byte consists of 8 bits.
8. // or /\* in C++ indicates the start  of a comment/statement.
9. A **variable** is a memory location whose value can change during the execution of the program.
10. A **Character datatype** can hold a sequence of characters such as a name.

## Lab 2.1

Working with cout statement

This is showing how one is to enter the codes to give an output of what is needed.

// This program will write the name, address and telephone

// number of the programmer.

// Jamal Caesar

#include <iostream>

using namespace std;

int main()

{

   cout << " \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* " << endl;

   cout << " \*                                                                      \* " << endl;

   cout << " \* Programmer: Jamal Caesar                                             \* " << endl;

   cout << " \*             #15 Bourkes Road Housing, Cleverly Hill, Sandy Point.    \* " << endl;

   cout << " \*             Basseterre, St. Kitts, KN0302.                           \* " << endl;

   cout << " \*                                                                      \* " << endl;

   cout << " \* Telephone: 466-3021 or (869) 763-6370                                \* " << endl;

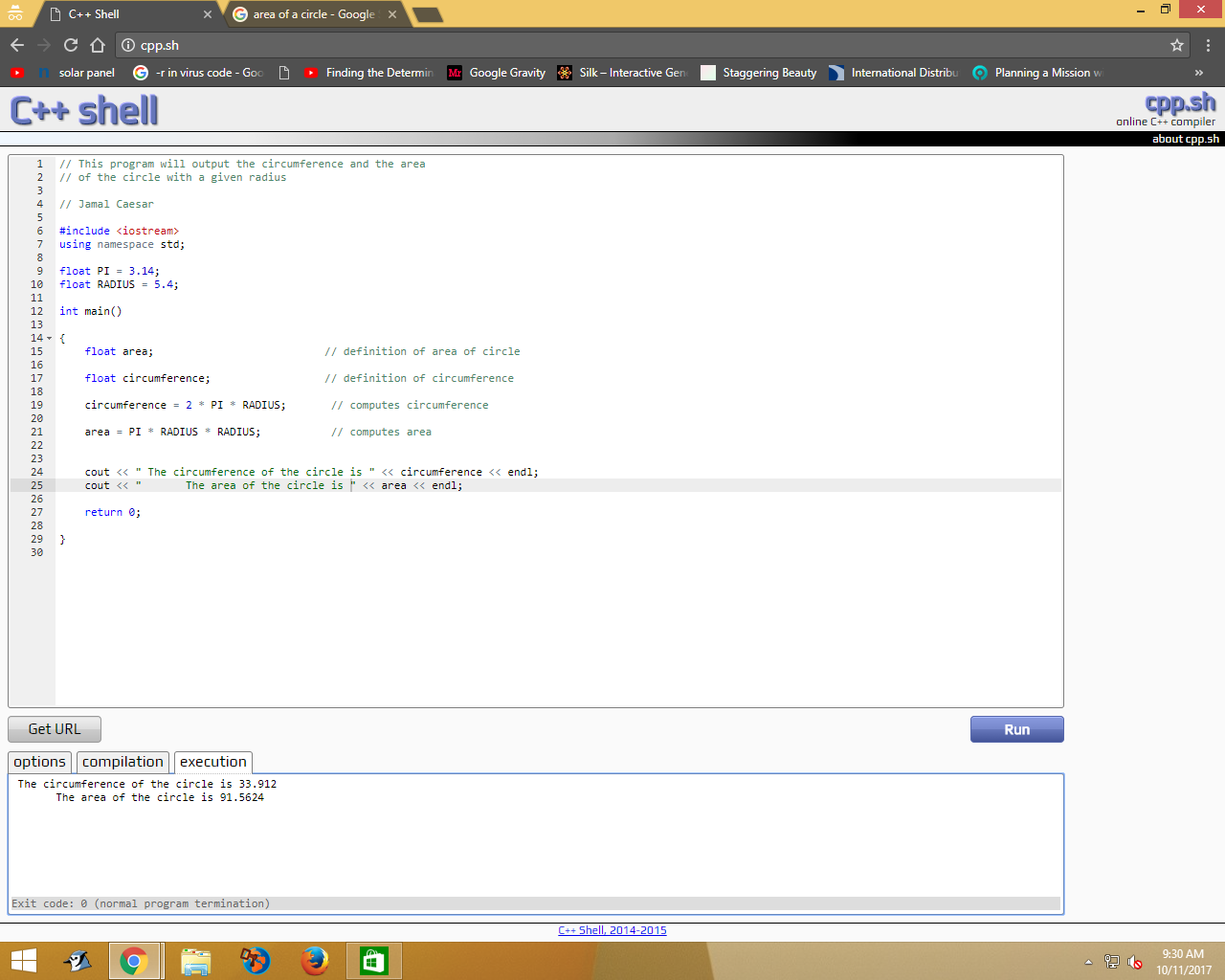
   cout << " \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* " << endl;

return 0;

}

## Lab 2.2

How one gets the results, they have to put the area as a floating point along with the circumference then input the codes that are below to help make it look like the screenshot.



This is the code for such program above.

#include <iostream>

using namespace std;

float PI = 3.14;

float RADIUS = 5.4;

int main()

{

   float area;

   float circumference;

   circumference = 2 \* PI \* RADIUS;

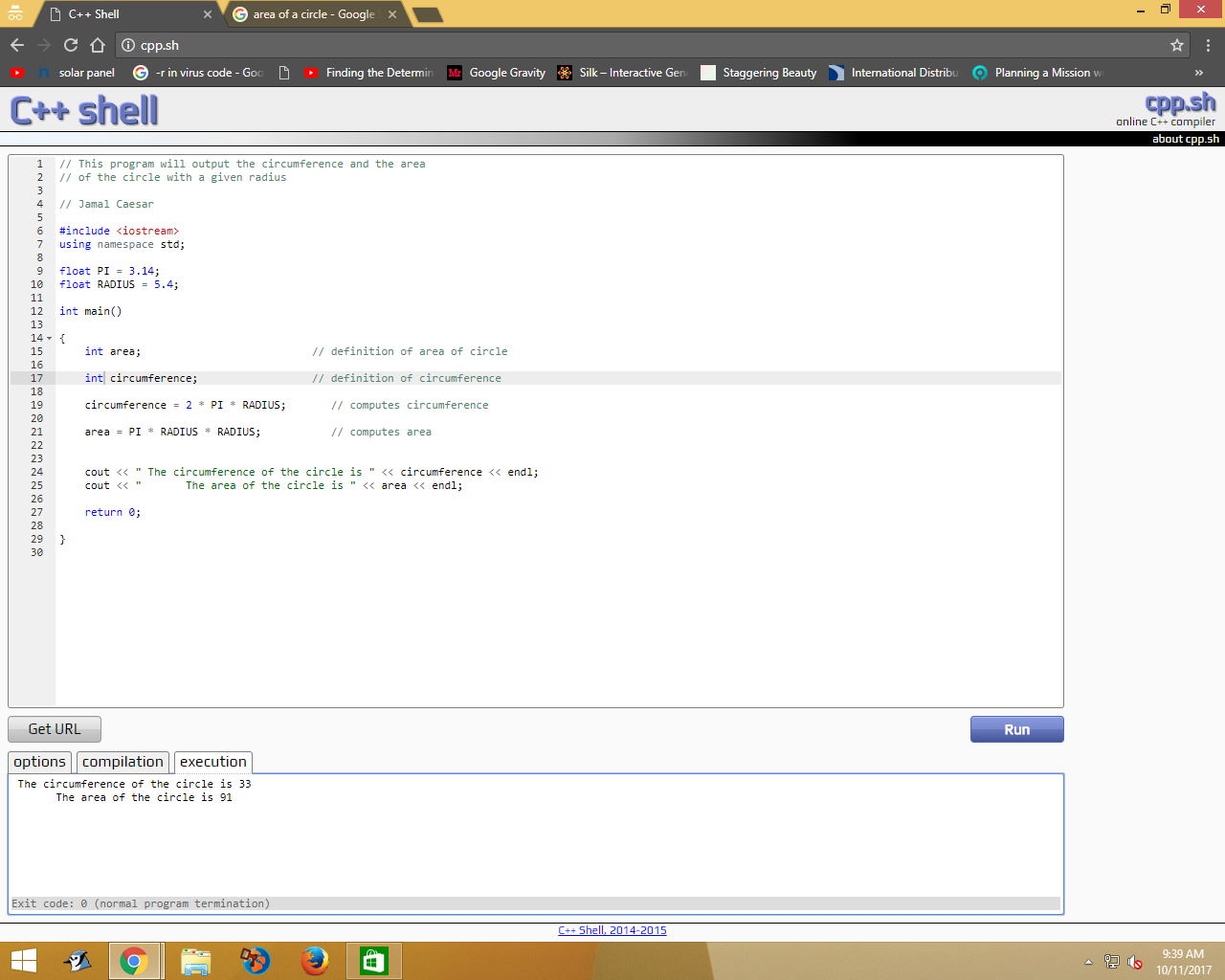
   area = PI \* RADIUS;

   cout << " The circumference of the circle is " << circumference << endl;

   cout << "       The area of the circle is " << area << endl;

   return 0;

}



Above is when the floating point data type is changed to integer and as the results show, it’s indicating the circumference is 33 and the area of the radius is 91.

Below, is the code to change the floating point to integer.

#include <iostream>

using namespace std;

float PI = 3.14;

float RADIUS = 5.4;

int main()

{

   int area;

   int circumference;

   circumference = 2 \* PI \* RADIUS;

   area = PI \* RADIUS;

   cout << " The circumference of the circle is " << circumference << endl;

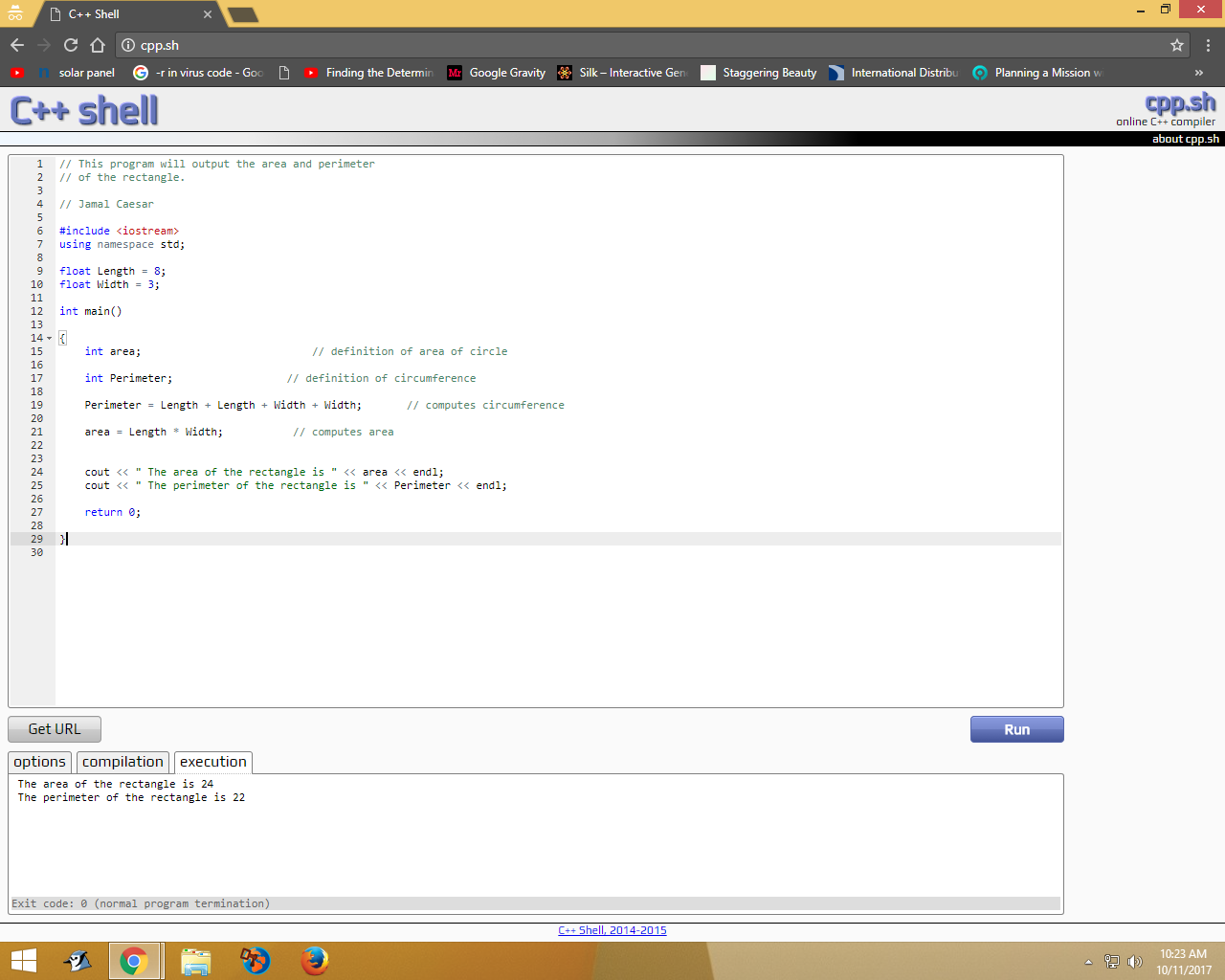
   cout << "       The area of the circle is " << area << endl;

   return 0;

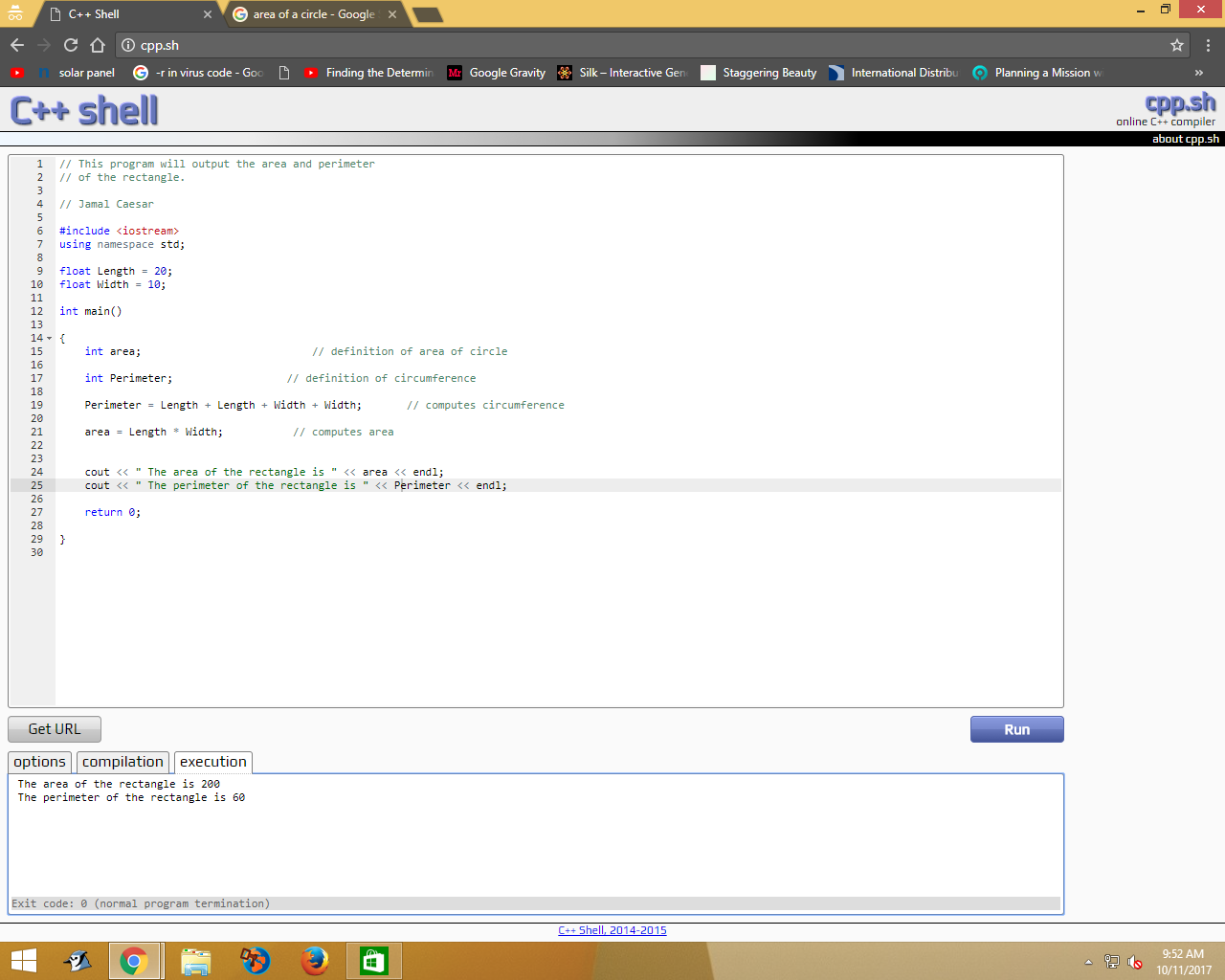
}

## Lab 2.3

In this exercise I had to create a program using the same codes from 2.2 to calculate the area and perimeter of a rectangular surface.



This is an image of the code that was used when I input my own codes to check if its working.



This is the code used to make the program run that way.

#include <iostream>

using namespace std;

float Length = 8;

float Width = 3;

int main()

{

   int area;                                    // Definition of area of rectangle

   int Perimeter;                               // Definition of Perimeter

   Perimeter = Length + Length + Width + Width; // computes perimeter

   area = Length \* Width;

   cout << " The area of the rectangle is " << area << endl;

   cout << " The perimeter of the rectangle is " << Perimeter << endl;

   return 0;

}

## Lab 2.4

This lab required the user to make a program that demonstrates the use of characters and strings, in it the programmer thus compiled the program and adjusted it to run without logic, syntax and run-time errors until it displays certain sentences.

Below, is the code of the program.

// This program demonstrates the use of characters and strings

// Jamal Caesar

#include <iostream>

using namespace std;

//Definition of constants

const string FAVORITESODA = "Dr. Dolittle"; // use of double quotes for strings

const char BESTRATING = 'A';                // use single quotes for characters

int main ()

{

   string rating, favoriteSnack;

   int numberOfPeople;

   int topChoiceTotal;

   string  crackers;

   favoriteSnack = crackers;

   rating = 'B';

   numberOfPeople = 250;

   topChoiceTotal = 148;

   cout << "The preferred soda is " << FAVORITESODA << endl;

   cout << "The preferred snack is " << favoriteSnack << endl;

   cout << "Out of " << numberOfPeople << " people, "

        << topChoiceTotal << " chose these items!" << endl;

   cout << "Each of these products were given a rating of " << BESTRATING ;

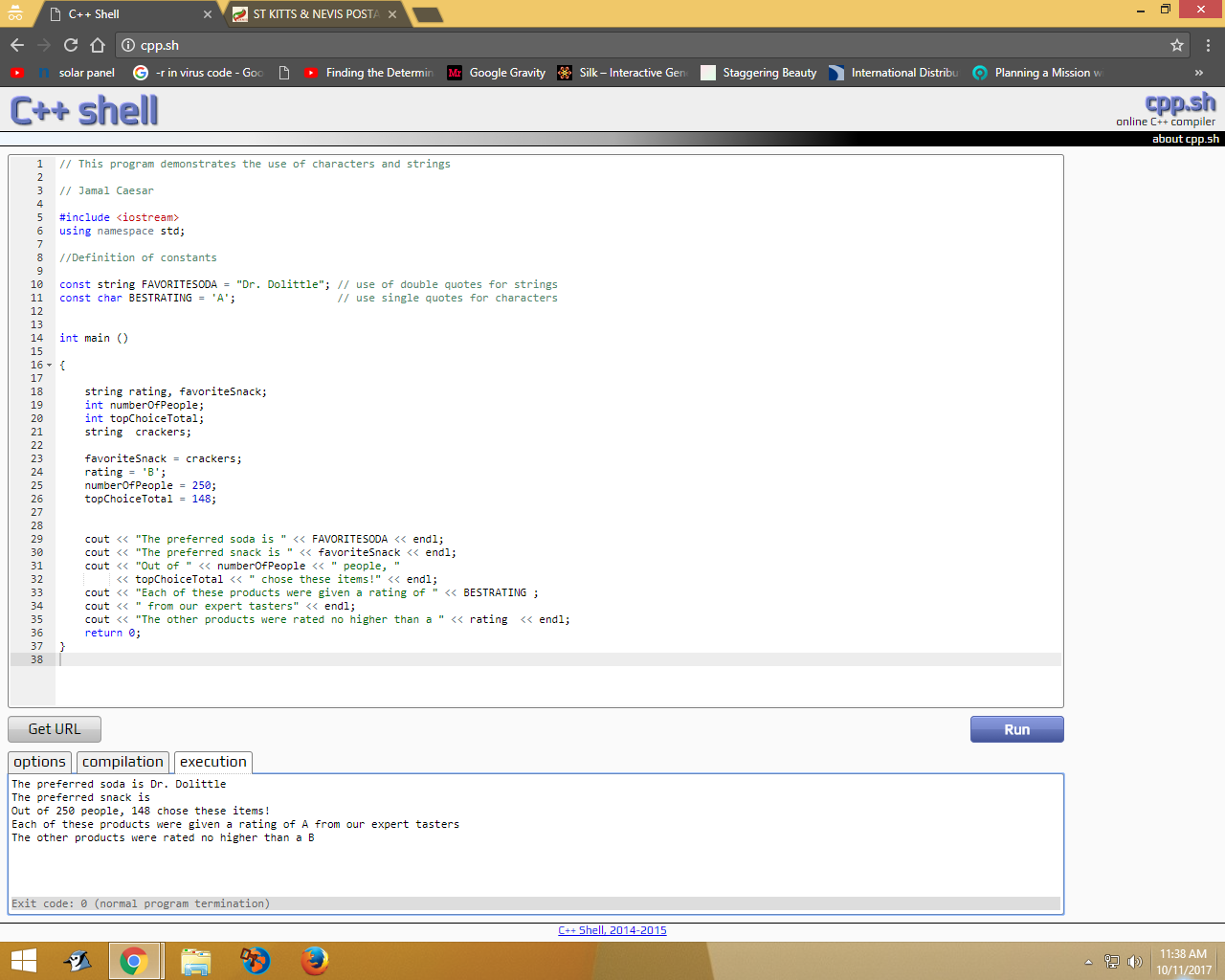
   cout << " from our expert tasters" << endl;

   cout << "The other products were rated no higher than a " << rating  << endl;

   return 0;

}

Now, below is the image of the program.



Exercise 3 asked if it was possible to change the choice of FAVORITESODA by adding code within the main module of the program, the answer would be no because it is a const string (constant string) which means it is a constant string variable which stays the same no matter what. The only way to change it is by going out of the main module program and changing the variable.

Exercise 4 asked if it was possible to change the choice of favoriteSnack by adding a code within the program. The answer would be yes since it is a string variable which means the programmer is able to change the choice of the favoriteSnack by just adding in a code that the programmer wants.