**PROBLEM SOLVING**

(Solving various problems using C language)

*Summer Internship Report Submitted in partial fulfillment*

*of the requirement for under graduate degree of*

**Bachelor of Technology**

In

**Computer Science Engineering**

By

**Sakshi Kasangottuwar**

**221710307053**

*Under the Guidance of*



Department Of Computer Science & Engineering

GITAM School of Technology

GITAM (Deemed to be University)

Hyderabad-502329

June 2019

1.

i

**DECLARATION**

I submit this industrial training work entitled “**Solving various problems using C** **language**” to GITAM (Deemed To Be University), Hyderabad in partial fulfillment of the requirements for the award of the degree of “**Bachelor of** **Technology**” in “**Computer Science & Engineering**”. I declare that it was carried out independently by me under the guidance of Asst. Professor, GITAM (Deemed To Be University), Hyderabad, India.

The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma.

Place: HYDERABAD SAKSHI KASANGOTTUWAR.

Date: 20-07-20 221710307053

2.

ii

GITAM (DEEMED TO BE UNIVERSITY)

     Hyderabad-502329, India

                                                                Dated:

**CERTIFICATE**

         This is to certify that the Industrial Training Report entitled **“PROBLEM SOLVING USING C LANGUAGE”** is being submitted by SAKSHI KASANGOTTUWAR (221710307053) in partial fulfillment of the requirement for the award of **Bachelor of Technology in Computer Science Engineering** at GITAM (Deemed To Be University), Hyderabad during the academic year 2019-20

                     It is faithful record work carried out by her at the **Computer Science Engineering Department,,** GITAM University Hyderabad Campus under my guidance and supervision.

Mr. S Phani Kumar

Assistant Professor Professor and HOD

Department of CSE Department of CSE

3.

iv

**ACKNOWLEDGEMENT**

Apart from my effort, the success of this internship largely depends on the encouragement and guidance of many others. I take this opportunity to express my gratitude to the people who have helped me in the successful competition of this internship.

I would like to thank respected **Dr. N. Siva Prasad,** Pro Vice Chancellor, GITAM Hyderabad and **N. Seetharamaiah,** Principal, GITAM Hyderabad

I would like to thank respected **Mr. S Phani Kumar,** Head of the Department of Computer Science and Engineering for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present a internship report. It helped me a lot to realize of what we study for.

I would like to thank the respected facultieswho helped me to make this internship a successful accomplishment.

I would also like to thank my friends who helped me to make my work more organized and well-stacked till the end.

                                                                                              SAKSHI KASANGOTTUWAR

221710307053

4.

**Table of Contents**

**1 Introduction to the project 6**

**2 Problem 1 7**

2.1 Problem statement 7

2.2 Coding 8

2.3 Output 8

**3 Problem 2 9**

3.1 Problem statement 9

3.2 Coding 10

3.3 Output 11

**4 Problem 3 12**

4.1 Problem statement 13

4.2 Coding 14

4.3 Output 14

**5 Problem 4 15**

5.1 Problem statement 16

5.2 Coding 17

5.3 Output 18

**6 Problem 5 19**

6.1 Problem statement 19

6.2 Coding 20

6.3 Output 21

**7 Problem 6 21**

5.

7.1 Problem statement 21

7.2 Coding 23

7.3 Output 24

**5 Hardware requirements 25**

**6 Software requirements 25**

6.

**1. Introduction to the project**

Problem Solving is the Process of Designing and carrying out certain steps to reach a Solution. 6 problems which are listed below are of different complexity and require different approach and logics in order to achieve desired Output/Solution..

**1.Consecutive Prime sum**: In this problem we have find whether the sum of two consecutive prime numbers is prime or not.

2.**Balancing stars:** In this code we are going to check the count of stars between the balanced parenthesis has the count greater than or equal to two.

**3.Stone game-one four:** When we evaluate this problem we can see that there is a specific pattern formation if we divide the number with 5 the remainder is 0 or 2 we can say that Bob wins otherwise Alice wins the game.

**4:Catch 22:** In this code we calculated the distance covered by the robot before falling into the ditch .In the code implementation we use if ,else, for and while loops.

**5.Super ASCII string checker:** In a super string,each character occurs ascii number of times.In this code we are going to check weather the string satisfies the given condition or not.

**6:Area of crazy ring:** In this code we are going to calculate the area of crazy ring using mathematical formulas.

In order to avoid math math library we applied Newton Square Root method .

I have executed projects in C language. For C, I have used Dev C++..

6.

**2. Problem 1**

**Consecutive Prime Sum**

In this program we are going to check if the sum of two consecutive prime number is prime or not if it is prime then we increment the count value.

**2.1 :** Some prime numbers can be expressed as Sum of other consecutive prime numbers.  
For example  
  
5 = 2 + 3  
17 = 2 + 3 + 5 + 7  
41 = 2 + 3 + 5 + 7 + 11 + 13  
  
Your task is to find out how many prime numbers which satisfy this property are present in the range 3 to N subject to a constraint that summation should always start with number 2.  
  
Write code to find out number of prime numbers that satisfy the above mentioned property in a given range.

**Input Format:**  
  
First line contains a number N

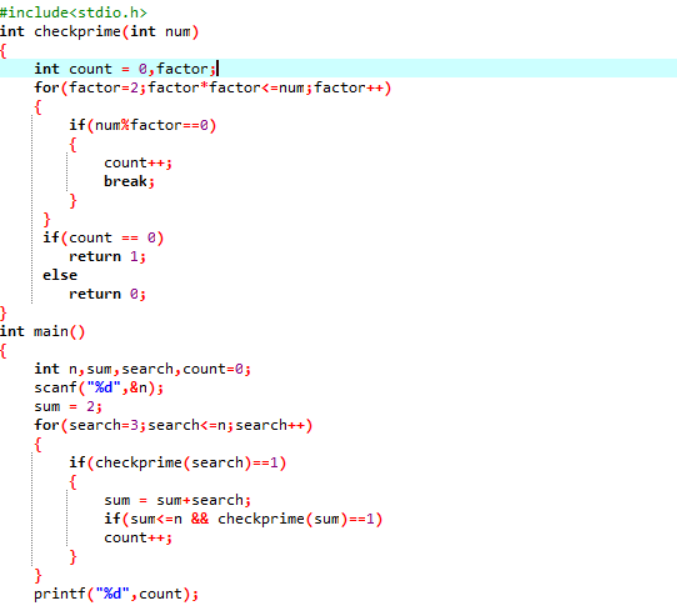
**Output Format:**  
  
Print the total number of all such prime numbers which are less than or equal to N.

**Constraints: 2<N<=12,000,000,000**

**7.**

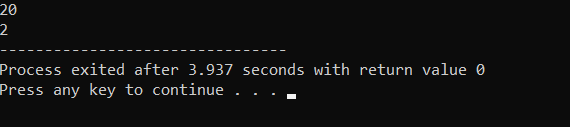
|  | **Input** | **Output** | **Comment** |
| --- | --- | --- | --- |
| 1 | 20 | 2 | (Below 20 there are two such members; 5 and 17) 5=2+3 17=2+3+65+7 |
| 2 | 15 | 1 |  |

**2.2 :**





**2.3:output**



8.

**3. Problem 2**

In this code we are going to check the count of stars between the balanced parenthesis has the count greater than or equal to two.There is a stack implemention in this code.

**Balancing stars**

**3.1:**

CODU loves to play with a string of brackets. He considers string as a good string if it is balanced with stars. A string is considered as balanced with stars if the string contains balanced brackets and between every pair of bracket i.e. between opening and closing brackets, there are at least 2 stars(\*) present. CODU knows how to check whether a string is balanced or not but this time he needs to keep a track of stars too. He decided to write a program to check whether a string is good or not. But CODU is not as good in programming as you are, so he decided to take help from you. Will you help him with this task? You need to print Yes and number of balanced pair if string satisfies following conditions(string is good if it satisfies following 2 conditions):

1. The string is balanced with respect to all brackets.

2. Between every pair of brackets, there are at least two stars.

However, if the string doesn't satisfy the above conditions then print No and a number of balanced pairs in a string as an output.

**Constraints**

4 <= String length <= 1000

**Input Format**

The first and only line of input contains a string of characters(a-z,A-Z), numbers(0-9), brackets( '{', '[', '(', ')', ']', '}' ) and stars(\*).

**Output**

Print space-separated "Yes" (without quotes) and a number of balanced pairs if the string is good. Else print "No" (without quotes) and the number of balanced pairs.

**Test Case**

**Explanation**

Example 1

Input

{\*\*}

Output

Yes 1

**Explanation**

Here string contains one balanced pair {} and between this pair of the bracket, there are 2 stars present so the output is Yes with the count of balanced pair as 1.

Example 2

Input

{\*\*(\*\*{\*\*[\*\*]})}

Output

Yes 4

**9.**

**Explanation**

A string has balanced brackets and also satisfies 2nd condition. So the output is Yes with the count of balanced pair which is 4.

Example 3

Input

\*\*}xasd[\*\*]sda231

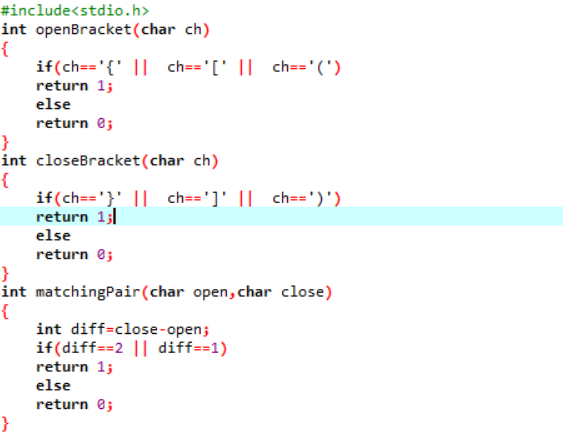
Output

No 1

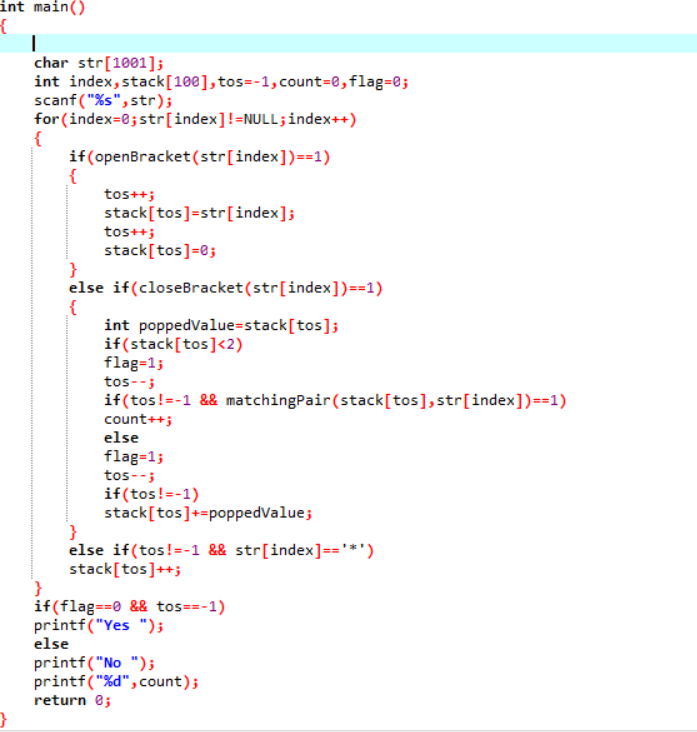
**Explanation**

In this case, the string is not balanced. So the output is No with the count of balanced pair as 1.

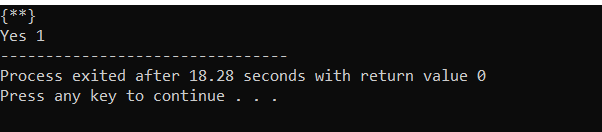
**3.2:**

****

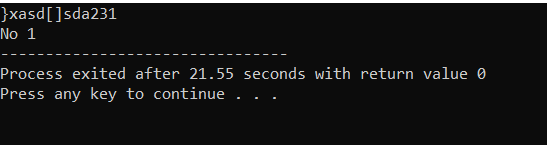
**10.**



**3.3:Output**

****

11.



**12.**

**4:Problem 3**

**Stone Game - One Four**

When we evaluate this problem we can see that there is a specific pattern formation if we divide the number with 5 the remainder is 0 or 2 we can say that Bob wins otherwise Alice wins the game.

**4.1:**

Alice and Bob are playing a game called "Stone Game". Stone game is a two-player game. Let N be the total number of stones. In each turn, a player can remove either one stone or four stones. The player who picks the last stone, wins. They follow the "Ladies First" norm. Hence Alice is always the one to make the first move. Your task is to find out whether Alice can win, if both play the game optimally.

**Input Format:**  
  
First line starts with T, which is the number of test cases. Each test case will contain N number of stones.

**Output Format:**  
  
Print "Yes" in the case Alice wins, else print "No".

**Constraints:**

**1<=T<=1000**

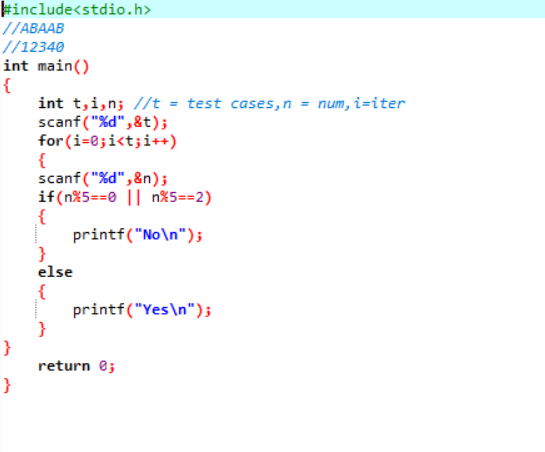
**1<=N<=10000**

[**Sample Input and Output**](https://www.blogger.com/null)

|  |  |  |
| --- | --- | --- |
| **SNo.** | **Input** | **Output** |
| 1 | 3 1 6 7 | Yes Yes No |

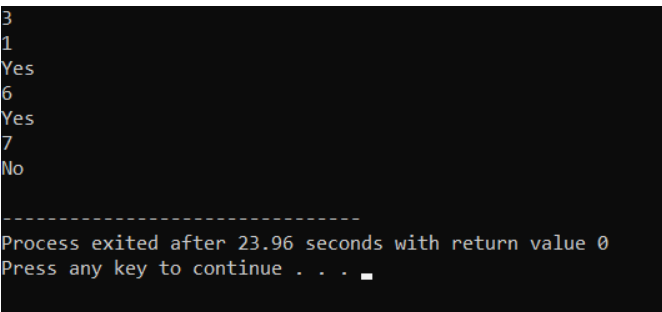
**4.2:**

13.

****

**4.3:**

**Output:**

****

14.

**5.Problem 4**:

In this code we calculated the distance covered by the robot before falling into the ditch .In the code implementation we use if ,else, for and while loops.

**Catch 22**

**5.1:**A robot is programmed to move forward F meters and backwards again, say B meters, in a straight line. The Robot covers 1 meter in T units of time. On Robot's path there is a ditch at a distance FD from initial position in forward direction as well as a ditch at a distance BD from initial position in backward direction. This forward and backward movement is performed repeatedly by the Robot.

Your task is to calculate amount of time taken, before the Robot falls in either ditch, if at all it falls in a ditch.

**Input Format:**  
  
First line contains total number of test cases, denoted by N  
Next N lines, contain a tuple containing 5 values delimited by space  
**F B T FD BD, where**

1. **F** denotes forward displacement in meters
2. **B** denotes backward displacement in meters
3. **T** denotes time taken to cover 1 meter
4. **FD** denotes distance from Robot's starting position and the ditch in forward direction
5. **BD** denotes distance from Robot's starting position and the ditch in backward direction

**15.**

**Output Format:**  
  
For each test case print time taken by the Robot to fall in the ditch and also state which ditch he falls into. Print F for forward and B for backward. Both the outputs must be delimited by whitespace

**OR**  
  
Print No Ditch if the Robot does not fall in either ditch

**Constraints:**

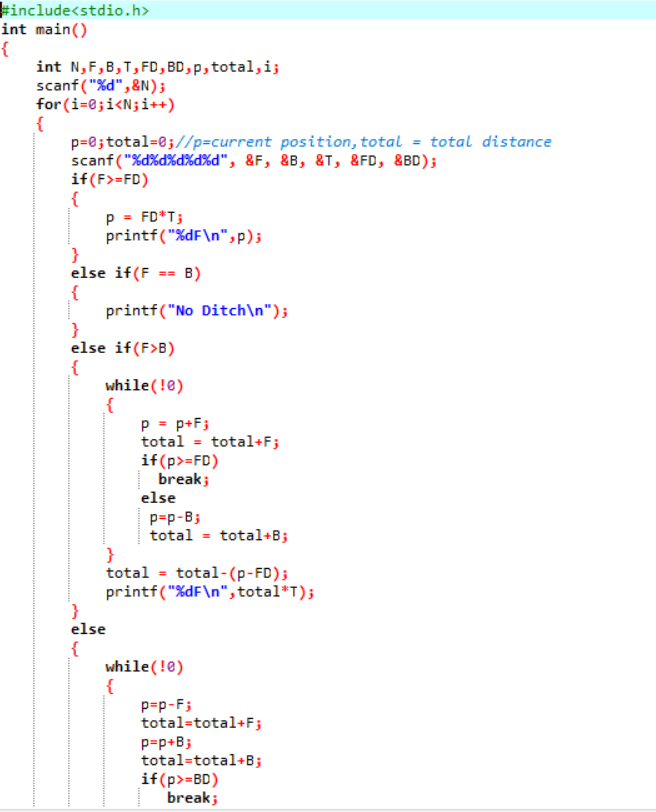
1. **First move will always be in forward direction**
2. **1 <= N <= 100**
3. **forward displacement > 0**
4. **backward displacement > 0**
5. **time > 0**
6. **distance of ditch in forward direction (FD) > 0**
7. **distance of ditch in backward direction (BD) > 0**
8. **All input values must be positive integers only**

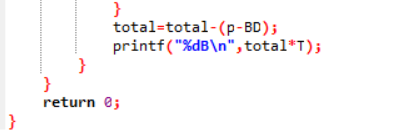
**Sample Input and Output**

|  |  |  |
| --- | --- | --- |
| **SNo.** | **Input** | **Output** |
| 1 | 3 9 4 3 13 10 9 7 1 11 13 4 4 3 8 12 | 63 F 25 F No Ditch |
| 2 | 5 8 4 7 11 22 4 5 4 25 6 4 9 3 6 29 7 10 6 24 12 10 10 1 9 7 | 133 F 216 B 231 B 408 B 9 F |

**5.2:**

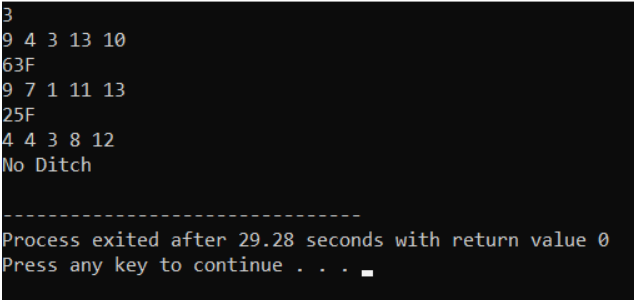
16.

****

****

**5.3:output**

17.

****

18.

**6.Problem 5:**

**Super ASCII string checker**

In a super string,each character occurs ascii number of times.

**6.1:**

In the Byteland country a string "S" is said to super ascii string if and only if count of each character in the string is equal to its ascii value.  
  
In the Byteland country ascii code of 'a' is 1, 'b' is 2 ...'z' is 26.  
  
Your task is to find out whether the given string is a super ascii string or not.

**Input Format:**  
  
First line contains number of test cases T, followed by T lines, each containing a string "S".

**Output Format:**  
  
For each test case print "Yes" if the String "S" is super ascii, else print "No"

**Constraints:**

**1<=T<=100**

**1<=|S|<=400, S will contains only lower case alphabets ('a'-'z').**

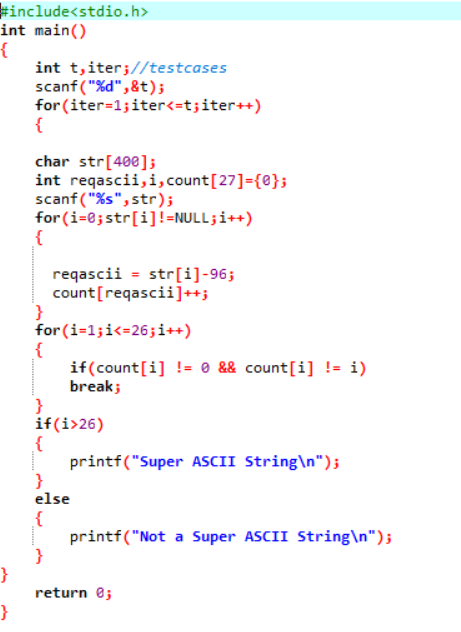
[**Sample Input and Output**](https://www.blogger.com/null)

|  |  |  |
| --- | --- | --- |
| **SNo.** | **Input** | **Output** |
| 1 | 2 bba scca | Yes No |

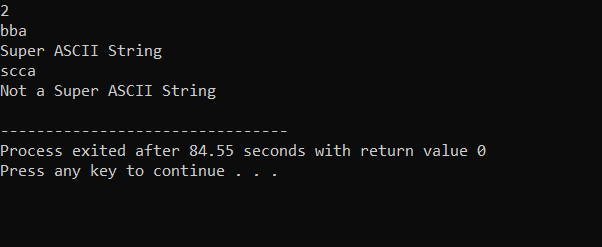
[**Explanation:**](https://www.blogger.com/null)  
 [In case 1, viz. String "bba" -](https://www.blogger.com/null)  
[The count of character 'b' is 2. Ascii value of 'b' is also 2.](https://www.blogger.com/null)  
[The count of character 'a' is 1. Ascii value of 'a' is also 1.](https://www.blogger.com/null)  
[Hence string "bba" is super ascii.](https://www.blogger.com/null)

**6.2:**

19.

****

**6.3:**

****

20.

**7.Problem 6:**

**Area Of The Crazy Ring**

In this code we are going to calculate the area of crazy ring using mathematical formulas.

In order to avoid math math library we applied Newton Square Root method .

**7.1:**

Scientists have found a strange substance having strange properties. There is one triangular substance called Strange Triangle which has a property that if it is placed inside a circle then it expands until all the corners of the triangle are touching the circle. There is another circular substance called Strange Circle, which when placed inside any polygon, expands such that it becomes the largest possible circle which can fit inside the polygon such that it touches every side of the polygon.

Now researchers did a strange experiment. They placed a Strange Triangle inside a normal circle and then placed a Strange Circle inside this Strange Triangle. Thus the ring formed by the two circles, the normal outer circle and the inner strange circle is named Crazy Ring. You are provided with the coordinates of the Strange Triangle on coordinate plane and you have to calculate the area of the **Crazy Ring**formed by the structure, Print "Not Possible" if the ring is not possible to form.

**Input Format:**

1. First line contains two space delimited numbers N1 M1 (N1 and M1 can also be negative)
2. Second line contains two space delimited numbers N2 M2 (N2 and M2 can also be negative)
3. Third line contains two space delimited numbers N3 M3 (N3 and M3 can also be negative)

21.

Where, (N1, M1) , (N2, M2) and (N3, M3) are x and y coordinates of three points representing the Strange Triangle

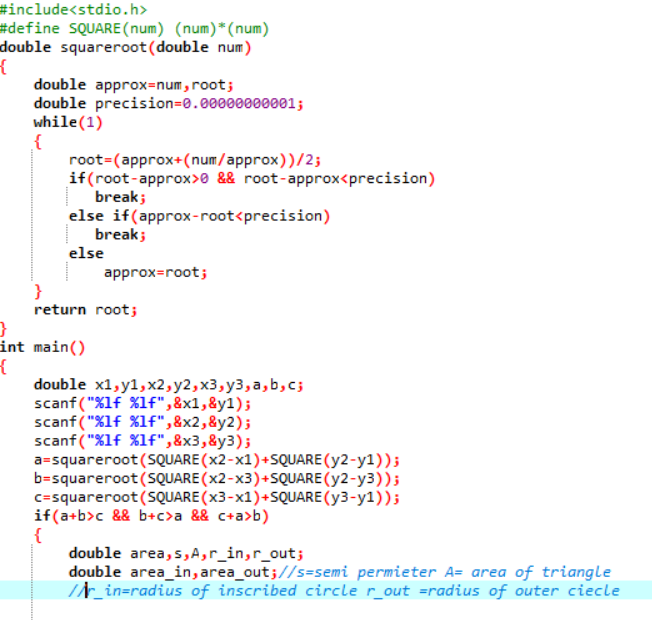
**Output Format:**  
  
Output the area of the crazy ring up to 2 decimal places however calculations are to be performed up to a precision of 11 decimal places  
  
OR  
  
Print "Not Possible" if it is not possible to form the ring

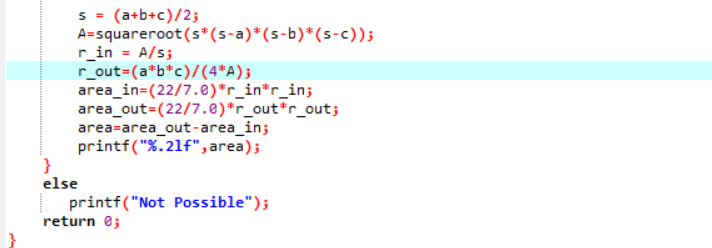
**Sample Input and Output**

|  |  |  |
| --- | --- | --- |
| **SNo.** | **Input** | **Output** |
| 1 | 5 5 5 20 20 5 | 292.79 |
| 2 | 5 5 5 5 5 5 | Not Possible |
| 3 | 5 8 4 3 2 4.34534554521 | 17.91 |

**7.2:**

**22.**

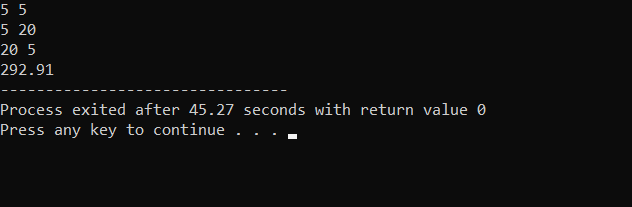
****

****

23.

**7.3:**

**Output:**

****

24.

**7.Hardware Requirements:**

This project can be executed in any system or an android phone without prior to any platform. We can use any online compiler and interpreter.

**6. Software Requirements:**

There are two ways to execute this project

1) Online compilers.

2) Softwares for execution ( Dev c++, etc.)

Online compilers require internet connection. We have many free compilers with which we can code.

Softwares for execution need to be installed on the user’s system specification. These help us to completely execute the project. These softwares are based on the platforms.

25.