# Problem: Distance between Two Points

1. Read the four values corresponding to the x and y axes of two points in the plane, p1 (x1, y1) and p2 (x2, y2) and calculate the distance between them, showing four decimal places after the comma, according to the formula:
2. Distance = https://resources.beecrowd.com.br/gallery/images/problems/UOJ_1015.png

## Input

1. The input file contains two lines of data. The first one contains two double values: ***x1 y1*** and the second one also contains two double values with one digit after the decimal point: ***x2 y2***.

## Output

1. Calculate and print the distance value using the provided formula, with 4 digits after the decimal point.

# #include <bits/stdc++.h>

# #include <iostream>

# #include <math.h>

# using namespace std;

# 

# int main() {

# double x1,x2,y1,y2,Distance;

# cin>>x1>>y1;

# cin>>x2>>y2;

# Distance=sqrt((x2-x1)\*(x2-x1)+(y2-y1)\*(y2-y1));

# cout<<fixed;

# cout<<setprecision(4)<<Distance<<endl;

# 

# return 0;

# }

# Problem: Salary with bonus.

Make a program that reads a seller's name, his/her fixed salary and the sale's total made by himself/herself in the month (in money). Considering that this seller receives 15% over all products sold, write the final salary (total) of this seller at the end of the month , with two decimal places.

- Don’t forget to print the line's end after the result, otherwise you will receive “Presentation Error”.

- Don’t forget the blank spaces.

## Input

The input file contains a text (employee's first name), and two double precision values, that are the seller's salary and the total value sold by him/her.

## Output

Print the seller's total salary, according to the given example.

| **Input Samples** | **Output Samples** |
| --- | --- |
| JOAO 500.00 1230.30 | TOTAL = R$ 684.54 |

# #include <bits/stdc++.h>

# #include <iostream>

# using namespace std;

# int main()

# {

# double salary;

# int product;

# char word[256];

# double Total;

# cout<<"Enter your employee name:";

# cin>>word;

# cout<<"\nEnter your employee salary=";

# cin>>salary;

# cout<<"\nEnter her total sale product=";

# cin>>product;

# Total=salary+(product\*15)/100;

# //cout<<"\nYour employee Total salary="<<Total;

# cout<<fixed;

# cout<<"Total salary="<<setprecision(2)<<Total;

# return 0;

# }

# Problem: Bhaskara’s formula.

Read 3 floating-point numbers. After, print the roots of bhaskara’s formula. If it's impossible to calculate the roots because a division by zero or a square root of a negative number, presents the message “Impossivel calcular”.

## Input

Read 3 floating-point numbers (double) A, B and C.

## Output

Print the result with 5 digits after the decimal point or the message if it is impossible to calculate.

| **Input Samples** | **Output Samples** |
| --- | --- |
| 10.0 20.1 5.1 | R1 = -0.29788 R2 = -1.71212 |

|  |  |
| --- | --- |
| 0.0 20.0 5.0 | Impossivel calcular |

#include<bits/stdc++.h>

#include<iostream>

#include<math.h>

using namespace std;

int main()

{

double a,b,c,t,r1,r2;

cout<<"First digit input=";

cin>>a;

cout<<"\nSecond digit input=";

cin>>b;

cout<<"\nThird digit input=";

cin>>c;

if(((b\*b)-(4\*a\*c)<0||a==0))

{

cout<<"\nImpossible the Solution for equation\n";

}

else{

t=sqrt((b\*b)-(4\*a\*c));

r1=((-b+t)/2\*a);

r2=((-b-t)/2\*a);

cout<<fixed;

cout<<setprecision(5)<<"\nR1= "<<r1<<endl;

cout<<setprecision(5)<<"\nR2= "<<r2<<endl;

}

return 0;

}

**4.Interval**

You must make a program that read a float-point number and print a message saying in which of following intervals the number belongs: [0,25] (25,50], (50,75], (75,100]. If the read number is less than zero or greather than 100, the program must print the message “Fora de intervalo” that means "Out of Interval".

The symbol '(' represents greather than. For example:  
[0,25] indicates numbers between 0 and 25.0000, including both.  
(25,50] indicates numbers greather than 25 (25.00001) up to 50.0000000.

## Input

The input file contains a floating-point number.

## Output

The output must be a message like following example.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 25.01 | Intervalo (25,50] |

**#include<iostream>**

**using namespace std;**

**int main()**

**{**

**float f;**

**cout<<"Enter your number=";**

**cin>>f;**

**if(f<0.0000&&f>100.0000){**

**cout<<"Fora de intervalo that menas out of interval\n";**

**}**

**else if(f>=0.0000&&f<=25.0000){**

**cout<<"Intervalo [0,25]\n";**

**}**

**else if(f>25.0000&&f<=50.0000){**

**cout<<"Intervalo (25,50]\n";**

**}**

**else if(f>50.0000&&f<=75.0000){**

**cout<<"Intervalo (50,75]\n";**

**}**

**else if(f>75.0000&&f<=100.0000){**

**cout<<"Intervalo (75,100]\n"; }**

**return 0;**

**}**

5.Bank notes

In this problem you have to read an integer value and calculate the smallest possible number of banknotes in which the value may be decomposed. The possible banknotes are 100, 50, 20, 10, 5, 2 e 1. Print the read value and the list of banknotes.

## Input:

The input file contains an integer value **N**(0 < **N**< 1000000).

## Output:

Print the read number and the minimum quantity of each necessary banknotes in Portuguese language, as the given example. Do not forget to print the end of line after each line, otherwise you will receive “Presentation Error”.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 576 | 576 5 nota(s) de R$ 100,00 1 nota(s) de R$ 50,00 1 nota(s) de R$ 20,00 0 nota(s) de R$ 10,00 1 nota(s) de R$ 5,00 0 nota(s) de R$ 2,00 1 nota(s) de R$ 1,00 |

#include<iostream>

using namespace std;

int main()

{

int totalTaka,reminder;

cout<<"Enter your total amount= ";

cin>>totalTaka;

cout<<"Your Taka: "<<totalTaka<<"\n\n";

cout<<totalTaka/100<<" Single note BDT 100.00\n";

reminder=(totalTaka%100);

cout<<reminder/50<<" Single note BDT 50.00\n";

reminder=(reminder%50);

cout<<reminder/20<<" Single note BDT 20.00\n";

reminder=(reminder%20);

cout<<reminder/10<<" Single note BDT 10.00\n";

reminder=(reminder%10);

cout<<reminder/5<<" Single note BDT 5.00\n";

reminder=(reminder%5);

cout<<reminder/2<<" Single note BDT 2.00\n";

reminder=(reminder%2);

cout<<reminder/1<<" Single note BDT 1.00\n";

return 0;

}

6.Interval 2

Read an integer **N**. This N will be the number of integer numbers **X**that will be read.

Print how many these numbers **X**are in the interval [10,20] and how many values are out of this interval

## Input

The first line of input is an integer **N**(**N**< 10000), that indicates the total number of test cases.  
Each case is an integer number **X**(-107 < **X** < 107).

## Output

For each test case, print how many numbers are in and how many values are out of the interval.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 4 14 123 10 -25 | 2 in 2 out |

#include <iostream>

using namespace std;

int main()

{

int x, a;

int in = 0;

int out = 0;

cin >> x;

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

{

cin >> a;

if(a >= 10 && a <= 20)

in++;

else

out++;

}

cout << in << " in\n";

cout << out << " out";

return 0;

}

# 7. Sum of Consecutive Odd Numbers

Read two integer values **X**and **Y**. Print the sum of all odd values between them.

## Input

The input file contain two integer values.

## Output

The program must print an integer number. This number is the sum off all odd values between both input values that must fit in an integer number.

| **Sample Input** | **Sample Output** |
| --- | --- |
| 6 -5 | 5 |

|  |  |
| --- | --- |
| 15 12 | 13 |

|  |  |
| --- | --- |
| 12 12 | 0 |

# #include<bits/stdc++.h>

# #include<iostream>

# using namespace std;

# int main()

# {

# int x,y, sum=0,i;

# int min,max;

# cout<<"Enter your 1st number: ";

# cin>>x;

# cout<<"Enter your 2nd number: ";

# cin>>y;

# if(x<y){

# min=x;

# max=y;

# }

# else{

# max=x;

# min=y;

# }

# for(i=(min+1);i<max;++i){

# if(i%2!=0){

# sum=sum+i;

# }

# }

# cout<<sum;

# return 0;

# }

## 8**. Fibonacci Series**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main() {
4. **int** n1=0,n2=1,n3,i,number;
5. cout<<"Enter the number of elements: ";
6. cin>>number;
7. cout<<n1<<" "<<n2<<" "; //printing 0 and 1
8. **for**(i=2;i<number;++i) //loop starts from 2 because 0 and 1 are already printed
9. {
10. n3=n1+n2;
11. cout<<n3<<" ";
12. n1=n2;
13. n2=n3;
14. }
15. **return** 0;
16. }

# 9. Prime number

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n, i, m=0, flag=0;
6. cout << "Enter the Number to check Prime: ";
7. cin >> n;
8. m=n/2;
9. **for**(i = 2; i <= m; i++)
10. {
11. **if**(n % i == 0)
12. {
13. cout<<"Number is not Prime."<<endl;
14. flag=1;
15. **break**;
16. }
17. }
18. **if** (flag==0)
19. cout << "Number is Prime."<<endl;
20. **return** 0;
21. }

## **10. Palindrome number**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n,r,sum=0,temp;
6. cout<<"Enter the Number=";
7. cin>>n;
8. temp=n;
9. **while**(n>0)
10. {
11. r=n%10;
12. sum=(sum\*10)+r;
13. n=n/10;
14. }
15. **if**(temp==sum)
16. cout<<"Number is Palindrome.";
17. **else**
18. cout<<"Number is not Palindrome.";
19. **return** 0;
20. }

# 11. Armstrong Number

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n,r,sum=0,temp;
6. cout<<"Enter the Number=  ";
7. cin>>n;
8. temp=n;
9. **while**(n>0)
10. {
11. r=n%10;
12. sum=sum+(r\*r\*r);
13. n=n/10;
14. }
15. **if**(temp==sum)
16. cout<<"Armstrong Number."<<endl;
17. **else**
18. cout<<"Not Armstrong Number."<<endl;
19. **return** 0;
20. }