

Codekata Report:

Name: Meena J

Email: meenait16bsc@gmail.com

1. You are given A = Length of a rectangle & B = breadth of a rectangle. Find its area "C".

(A and B are natural numbers)

Sample Input:

2
3

Sample Output:

6

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
A = int(input())
B = int(input())
area = A * B
print(round(area,1))
```

Compilation Details:

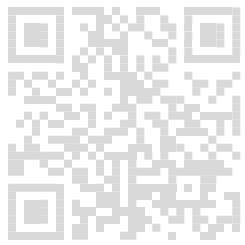
TestCase1:

Input:

< hidden >

Expected Output:

< hidden >



Output:

144

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

30

Compilation Status: Passed

Execution Time:

0.009s

2. You are provided with a number check whether its odd or even.

Print "Odd" or "Even" for the corresponding cases.

Note: In case of a decimal, Round off to nearest integer and then find the output. Incase the input is zero, print "Zero".

Sample Input:

2

Sample Output:

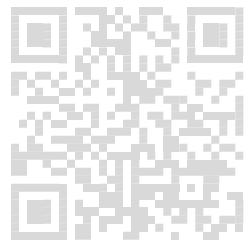
Even

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3



Source Code:

```
N=int (input ())
if N==0:
print("Zero")
elif N%2==0:
print("Even")
else:
print("Odd")
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Even

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Odd

Compilation Status: Passed

Execution Time:

0.01s

3. You are given three numbers A, B & C. Print the largest among these three numbers.



Sample Input:

1
2
3

Sample Output:

3

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
a=int(input())
b=int(input())
c=int(input())
print(max(a,b,c))
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

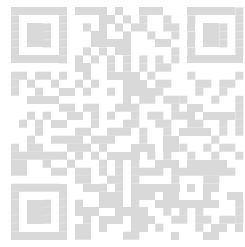
3

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:



Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.009s

**4. Using the method of looping, write a program to print the table of 9 till N in the format as follows:
(N is input by the user)**

9 18 27...

Print NULL if 0 is input

Sample Input:

3

Sample Output:

9 18 27

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
n=int(input())
```

```
if (n>=1):
```

```
i=0
```

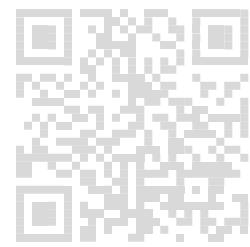
```
for i in range(1,n):
```

```
    m=i*9
```

```
    print(m,end=' ')
```

```
else:
```

```
    print("NULL")
```



Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

```
9 18 27
```

Compilation Status: Passed

Execution Time:

```
0.01s
```

TestCase2:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

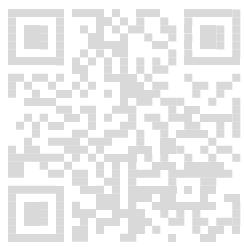
```
9
```

Compilation Status: Passed

Execution Time:

```
0.01s
```

5. You are provided with two numbers. Find and print the smaller number.



Sample Input:

23 1

Sample Output:

1

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
A,B = map(int,input().split())
print(min(A,B))
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

2

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

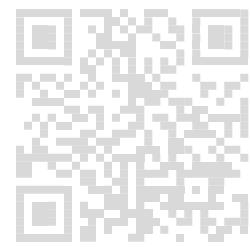
Output:

32

Compilation Status: Passed

Execution Time:

0.009s



6. You will be provided with a number. Print the number of days in the month corresponding to that number.

Note: In case the input is February, print 28 days. If the Input is not in valid range print "Error".

Sample Input:

8

Sample Output:

31

Completion Status: Completed

Concepts Included:

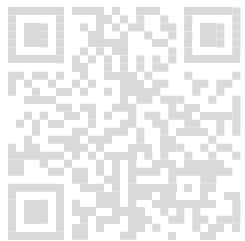
absolute beginner

Language Used: PYTHON 3

Source Code:

```
month = int(input())

if(month==2):
    print(28)
elif(month==4|month==6|month==9|month==11):
    print(30)
elif(month==1|month==3|month==5|month==7|month==8|month==10|month==12):
    print(31)
else:
    print('Error')
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Error

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Error

Compilation Status: Passed

Execution Time:

0.01s

7. Write a code to get an integer N and print the values from N to 1.

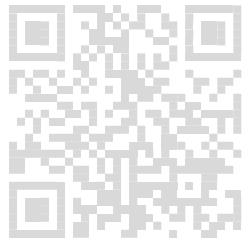
Sample Input:

10

Sample Output:

10
9
8
7

6
5
4
3
2
1



Completion Status: Completed

Concepts Included:

absolute beginner

basics

Looping

Language Used: PYTHON 3

Source Code:

```
N=int(input())
```

```
for i in range(N,0,-1):  
    print(i)
```

Compilation Details:

TestCase1:

Input:

```
< hidden >
```

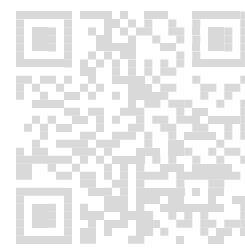
Expected Output:

```
< hidden >
```

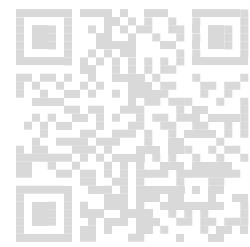
Output:

100
99
98
97
96
95
94
93
92
91
90
89
88

87
86
85
84
83
82
81
80
79
78
77
76
75
74
73
72
71
70
69
68
67
66
65
64
63
62
61
60
59
58
57
56
55
54
53
52
51
50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34



33
32
31
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1



Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

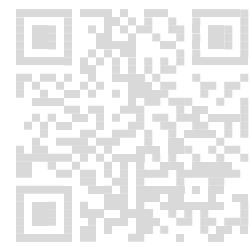
< hidden >

Output:

5
4
3

2
1

Compilation Status: Passed



Execution Time:

0.01s

8. Let "A" be a year, write a program to check whether this year is a leap year or not.

Print "Y" if its a leap year and "N" if its a common year.

Sample Input:

2020

Sample Output:

Y

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
year = int(input())

if (year % 4) == 0:
    if (year % 100) == 0:
        if (year % 400) == 0:
            print("Y")
        else:
            print("N")
    else:
        print("Y")
else:
    print("N")
```

Compilation Details:

TestCase1:



Input:

< hidden >

Expected Output:

< hidden >

Output:

N

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Y

Compilation Status: Passed

Execution Time:

0.01s

9. Write a code to get an integer N and print the sum of values from 1 to N.

Sample Input:

10

Sample Output:

55

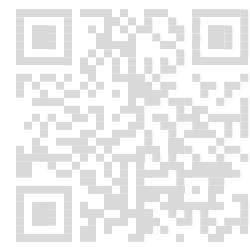
Completion Status: Completed

Concepts Included:

absolute beginner

basics

Looping



Language Used: PYTHON 3

Source Code:

```
a = int(input())  
total = a * (a + 1) // 2  
print(total)
```

Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

5050

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

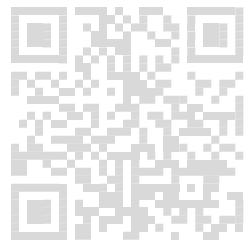
Output:

1225

Compilation Status: Passed

Execution Time:

0.01s



10. You are given with Principle amount(\$), Interest Rate(%) and Time (years) in that order. Find Simple Interest.

Print the output up to two decimal places (Round-off if necessary).

$$(\text{S.I.} = P \times T \times R / 100)$$

Sample Input:

1000 2 5

Sample Output:

100.00

Completion Status: Completed

Concepts Included:

absolute beginner

Language Used: PYTHON 3

Source Code:

```
p,t,r = map(float,input().split(" "))

si=(p*t*r)/100

simple = "{:.2f}".format(si)

print(simple)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

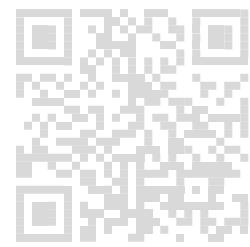
Output:

100.00

Compilation Status: Passed

Execution Time:

0.009s



TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

112.20

Compilation Status: Passed

Execution Time:

0.01s

11. Write a code to get 2 integers as input and find the HCF of the 2 integer without using recursion or Euclidean algorithm.

Sample Input:

2 3

Sample Output:

1

Completion Status: Completed

Concepts Included:

absolute beginner

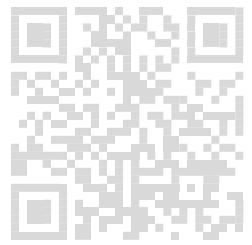
basics

Looping

Language Used: PYTHON 3

Source Code:

```
import math  
x,y = map(int,input().split(" "))  
print(math.gcd(x,y))
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

19

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

15

Compilation Status: Passed

Execution Time:

0.01s

12. Write a code get an integer number as input and print the sum of the digits.

Sample Input:

124

Sample Output:

7



Completion Status: Completed

Concepts Included:

absolute beginner

basics

Looping

Language Used: PYTHON 3

Source Code:

```
n = int(input())
r = 0
while n>0:
    d = n%10
    r = r+d
    n = n//10
print(r)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

45

Compilation Status: Passed

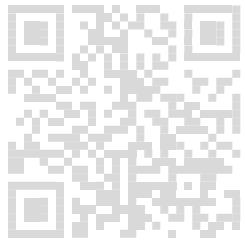
Execution Time:

0.01s

TestCase2:

Input:

< hidden >

**Expected Output:**

< hidden >

Output:

49

Compilation Status: Passed**Execution Time:**

0.01s

13. Given a string 'S' print the sum of weight of the String. A weight of character is defined as the ASCII value of corresponding character.

Sample Input:

abc

Sample Output:

294

Completion Status: Completed**Concepts Included:**

strings

Language Used: PYTHON 3**Source Code:**

```
s = input()  
t = list(map(ord,s))  
print(sum(t))
```

Compilation Details:**TestCase1:****Input:**

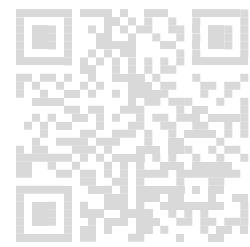
< hidden >

Expected Output:

< hidden >

Output:

294



Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

201

Compilation Status: Passed

Execution Time:

0.01s

14. You are given two numbers. Your task is to multiply the two numbers and print the answer.

Sample Input:

99999 99998

Sample Output:

9999700002

Completion Status: Completed

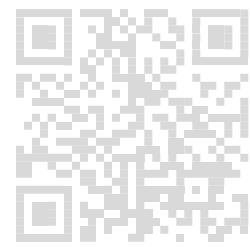
Concepts Included:

strings

Language Used: PYTHON 3

```
n=list(map(str,input().split()))
```

```
print(int(n[0])*int(n[1]))
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

9999300006

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

9999700002

Compilation Status: Passed

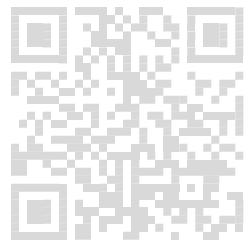
Execution Time:

0.01s

15. You are given an array. Your task is to sort the array in given manner. Print the elements in increasing order of the frequency. If frequency is same print smaller one first.

Sample Input:

1 1 3 2



Sample Output:

2 3 1

Completion Status: Completed

Concepts Included:

array

Language Used: PYTHON 3

Source Code:

```
#taking number of elements as input
n = int(input())
# taking the list of number as input
arr= list(map(int,input().split()))

#importing the library for getting the counts of the elements
from collections import Counter

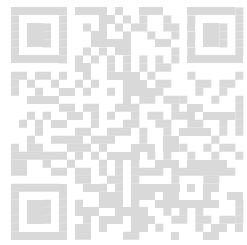
def frequencySort(nums):
    cn = Counter(nums)
    #alternatively this can be done using dict also
    # dict1 = {}
    # for i in nums:
    #     dict1[i]=nums.count(i)
    # for k, v in sorted(dict1.items(), key = lambda kv: (kv[1],kv[0])):
    #     out.append(v*k)

    #making a 2-dimensional array to store the values
    out = []
    for k, v in sorted(cn.items(), key = lambda kv: (kv[1],kv[0])):
        out.append(v*k)

    # this gives the one-dimensional array
    o = []
    for i in out:
        o+= i

    #finally getting all the unique values from the array
    final = []
    for i in o:
        if i not in final:
            final.append(i)
    return final

print(*frequencySort(arr))
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

2 3 1

Compilation Status: Passed

Execution Time:

0.012s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

3 13 1

Compilation Status: Passed

Execution Time:

0.011s

16. Given a string S, print it after changing the middle element to * (if the length of the string is even, change the 2 middle elements to *). Sample Testcase :INPUTHelloOUTPUTThe*lo

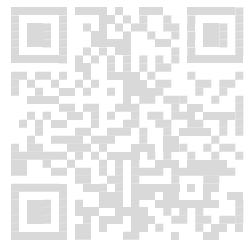
Completion Status: Completed

Concepts Included:

array

strings

Language Used: PYTHON 3



Source Code:

```
S = input().strip()
D = len(S)//2

if len(S)%2==0:
print(S[0:D-1]**+S[D+1:])
else:
print(S[0:D]**+S[D+1:])
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

*

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

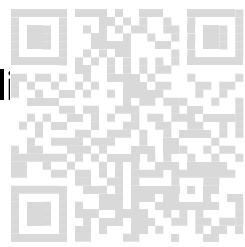
sa**ad

Compilation Status: Passed

Execution Time:

0.01s

17. Prateek finds it difficult to judge the minimum element in the list of elements given to him. Your task is to develop the algorithm in order to find the minimum element.



Sample Input:

5
3 4 9 1 6

Sample Output:

1

Completion Status: Completed

Concepts Included:

array

numbers

mathematics

Language Used: PYTHON 3

Source Code:

```
N=int(input())
l=list(map(int,input().split()))
l.sort()
print(l[0])
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

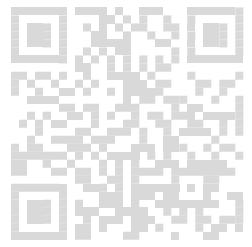
Output:

1

Compilation Status: Passed

Execution Time:

0.01s



TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.009s

18. Ram is the CEO of an MNC. He wants to order the employee salaries in ascending order so that he can do a salary hike based on the salary values of employees. He selects you to do the task of sorting the salaries. Sort the salaries in ascending order and pass on the information to Ram.

Sample Input:

8
7000 8000 6500 1200 4000 2800 3000 5230

Sample Output:

1200 2800 3000 4000 5230 6500 7000 8000

Completion Status: Completed

Concepts Included:

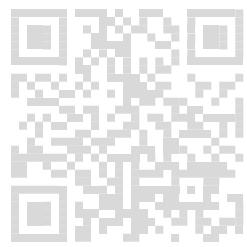
sorting

array

Language Used: PYTHON 3

Source Code:

```
employee=input()
No=list(map(int,input().split()))
No.sort()
print(*No)
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1 2 3 4 5

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1 8 9

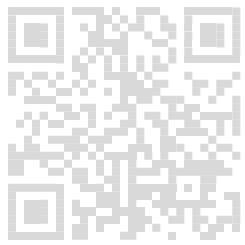
Compilation Status: Passed

Execution Time:

0.01s

19. Given 2 numbers N and K followed by N elements, print the number of repetition of K otherwise print '-1' if the element not found. Sample Testcase : INPUT 6 2 1 2 3 5 7 8 OUTPUT 0

Completion Status: Completed



Concepts Included:

basics

mathematics

array

Language Used: PYTHON 3

Source Code:

```
n,m=map(int,input().split())
b=map(int,input().split()[:n])
b=list(b)
count=-1
for i in b:
if(i==m):
count+=1
print(count)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

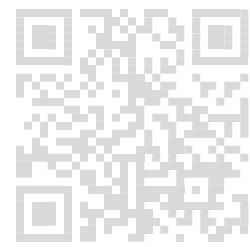
Output:

-1

Compilation Status: Passed

Execution Time:

0.009s



20. Ria is always fascinated by the number 2. She always wants to know who came second in a race, the second person to set foot on the moon and so on. She is given a list of numbers and asked to find the maximum. As always, she reports the second highest number as the maximum because according to her, 2 is higher than 1. Find out which was the number that Ria would have reported, given a list of N numbers.

Sample Input:

10
1 9 8 7 6 5 2 3 4 10

Sample Output:

9

Completion Status: Completed

Concepts Included:

searching

array

Language Used: PYTHON 3

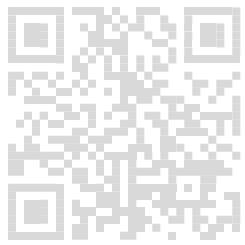
Source Code:

```
size=int(input())
elem=list(map(int,input().split()))
elem.sort()
print(elem[-2])
```

Compilation Details:

TestCase1:

Input:



< hidden >

Expected Output:

< hidden >

Output:

2

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

76

Compilation Status: Passed

Execution Time:

0.01s

21. Given a sentence and string S, find how many times S occurs in the given sentence. If S is not found in the sentence print -1
Input Size : |sentence| <= 1000000 (complexity O(n)). Sample Testcase : INPUT1 enjoy doing codekataOUTPUT1

Completion Status: Completed

Concepts Included:

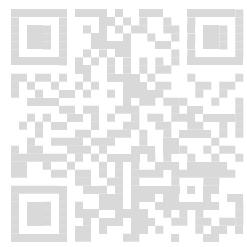
strings

array

Language Used: PYTHON 3

Source Code:

```
S=input().split()  
K=input()  
res=S.count(K)  
if (res>=1):  
    print(res)  
else:  
    print("-1")
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

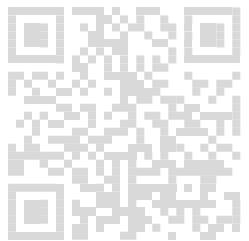
-1

Compilation Status: Passed

Execution Time:

0.01s

22. Pk finds it difficult to judge the minimum element in the list of elements given to him. Your task is to develop the algorithm in order to find the minimum element.



Note:Don't use sorting

Sample Input:

5
3 4 9 1 6

Sample Output:

1

Completion Status: Completed

Concepts Included:

mathematics

array

Language Used: PYTHON 3

Source Code:

```
N=int(input())
l=list(map(int,input().split()))
l.sort()
print(l[0])
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

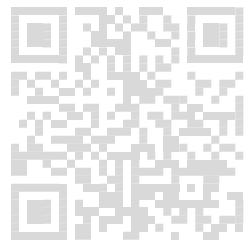
Output:

2

Compilation Status: Passed

Execution Time:

0.009s



TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.01s

23. Mr.Stark wants to order the employee ids, which are recorded in a 2D matrix, in ascending order. He wants to do it so as to allot a new id to a person who joins as a fresher. You are the CTO of the Stark industries and you are asked by Mr.Stark to sort the data.

Sample Input:

```
3 3
87 21 34
89 32 78
12 23 45
```

Sample Output:

```
12 21 23
32 34 45
78 87 89
```

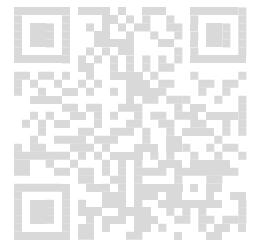
Completion Status: Completed

Concepts Included:

sorting

array

Language Used: PYTHON 3



Source Code:

```
n,m = map(int,input().split())
matrix=[]
for i in range(n):
    row = list(map(int,input().split()))
    matrix.append(row)

temp=[0]*(n*m)
k=0
for i in range(n):
    for j in range(m):
        temp[k]=matrix[i][j]
    k+=1
temp.sort()
k=0
for i in range(n):
    for j in range(m):
        matrix[i][j]=temp[k]
    k+=1

for row in matrix:
    print(*row)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1 2 3
7 8 9

Compilation Status: Passed

Execution Time:

0.01s

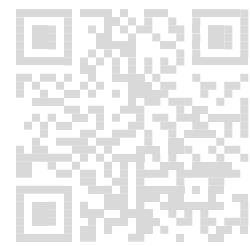
TestCase2:

Input:

< hidden >

Expected Output:

< hidden >



Output:

23 34 54
56 64 89

Compilation Status: Passed

Execution Time:

0.01s

24. You are an intern at GUVI and the company wants to organise its data and delete unnecessary extra storage elements used. You are given k arrays of unequal dimensions. Sort the k arrays individually and concatenate them.

Sample Input:

3
2
98 12
6
1 2 3 8 5 9
1
11

Sample Output:

12 98 1 2 3 5 8 9 11

Completion Status: Completed

Concepts Included:

sorting

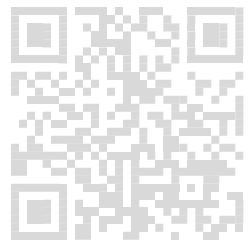
array

Language Used: PYTHON 3

Source Code:

```
a=int(input())
b=[]
for i in range(a):
    x=int(input())
    y=list(map(int,input().split()))
```

```
y.sort()  
for j in range(len(y)):  
    b.append(y[j])  
print(*b)
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1 2 3 10 12 43 66 76

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

2 45 67 9 12 56

Compilation Status: Passed

Execution Time:

0.01s

25. Find the minimum among 10 numbers. Sample
Testcase :INPUT5 4 3 2 1 7 6 10 8 9 OUTPUT1

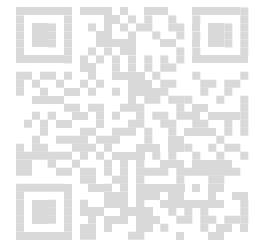
Completion Status: Completed

Concepts Included:

basics

mathematics

Language Used: PYTHON 3



Source Code:

```
I = list(map(int,input().split()))  
I.sort()  
print(I[0])
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1

Compilation Status: Passed

Execution Time:

0.01s

26. You are provided with a number 'n'. Your task is to tell whether that number is saturated. A saturated number is a number which is made by exactly two digits.



Sample Input:

121

Sample Output:

Saturated

Completion Status: Completed

Concepts Included:

mathematics

numbers

Language Used: PYTHON 3

Source Code:

```
inputt = input()
sett = list(set(inputt))

if(len(sett) == 2):
    print("Saturated")
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

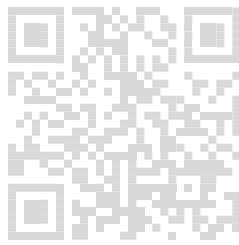
Output:

Saturated

Compilation Status: Passed

Execution Time:

0.01s



TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Saturated

Compilation Status: Passed

Execution Time:

0.009s

27. A person saves his monthly saving according to given schema. He saves same amount of money which is equal to the money saved in immediate previous two months. Assume, initially he saved 1000 rupees and in first month he saved another 1000. Your task is to tell how much he had totally saved at the end of 'n' months

Sample Input:

1

Sample Output:

2000

Completion Status: Completed

Concepts Included:

mathematics

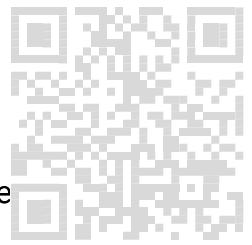
array

Language Used: PYTHON 3

Source Code:

```
n=int(input())
def fib(n):
    if n==1:
        return 1000
    else:
        return fib(n-1)+fib(n-2)
```

```
elif n==2:  
    return 4000 # (1000 + 2000)+1000  
else:  
    return fib(n-1)+fib(n-2)+1000 # He saves same amount of money = the money save  
in immediate previous two months +1000  
print(fib(n))
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

4000

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

7000

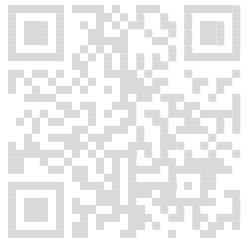
Compilation Status: Passed

Execution Time:

0.014s

28. Given 2 numbers N,M. Print 'yes' if their product is a perfect square else print 'no'. Sample Testcase :INPUT5 5OUTPUTyes

Completion Status: Completed



Concepts Included:

mathematics

basics

Language Used: PYTHON 3

Source Code:

```
N,M=map(int,input().split())
product=N*M
if product**0.5==N:
    print("yes")
else:
    print("no")
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

yes

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

no

Compilation Status: Passed



Execution Time:

0.009s

29. You are given a large number made of only 0's and 1's. Your task is to find the max no of consecutive 1's. If there are no 1's print -1

Sample Input:

101011111

Sample Output:

5

Completion Status: Completed

Concepts Included:

mathematics

bit manipulation

binary

Language Used: PYTHON 3

Source Code:

```
ui = input()
n = len(ui)

count = 0
result = 0
for i in range(0, n):
    if ui[i] == '0':
        count = 0
    else:
        count += 1
    result = max(result, count)

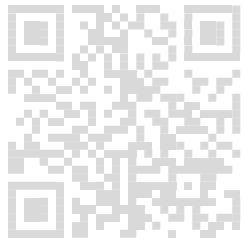
print(result) if count > 0 else print(-1)
```

Compilation Details:

TestCase1:

Input:

1111

**Expected Output:**

< hidden >

Output:

3

Compilation Status: Passed**Execution Time:**

0.009s

TestCase2:**Input:**

< hidden >

Expected Output:

< hidden >

Output:

-1

Compilation Status: Passed**Execution Time:**

0.01s

30. Write a code get an integer number as input and print the odd and even digits of the number separately.

Sample Input:

1234

Sample Output:

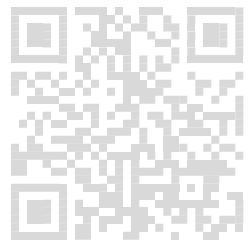
2 4
1 3

Completion Status: Completed**Concepts Included:**

basics

absolute beginner

Looping



Language Used: PYTHON 3

Source Code:

```
a=int(input())
ev=[]
od=[]
b=list(map(int, str(a)))
for x in b:
if (x%2 ==0):
ev.append(x)
else:
od.append(x)
print(*sorted(ev))
print(*sorted(od))
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

2 2 4
3 3 3

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

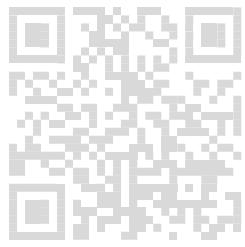
2 2 2 4

3 3 5 5 5

Compilation Status: Passed

Execution Time:

0.01s



31. Write a code to get 2 integers as input and add the integers without any carry.

Sample Input:

44 66

Sample Output:

0

Completion Status: Completed

Concepts Included:

basics

Looping

Language Used: PYTHON 3

Source Code:

```
a, b = list(map(int, input().split()))
```

```
c = a + b
```

```
d = str(c)
```

```
A = str(a)
```

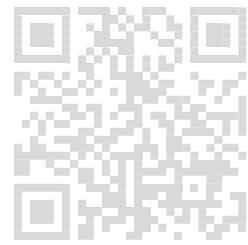
```
B = str(b)
```

```
if c < 110 and ( len(d) > len(A) or len(d) > len(B)) :  
    print(d[-1])
```

```
elif c > 110 :  
    print((c - 100)/2)
```

```
else:  
    print(c)
```

Compilation Details:



TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

9

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

10.0

Compilation Status: Passed

Execution Time:

0.009s

32. Given a string S consisting of 2 words reverse the order of two words .**Input Size :** |S| <= 10000000**Sample Testcase :**INPUTHello world OUTPUTworld hello

Completion Status: Completed

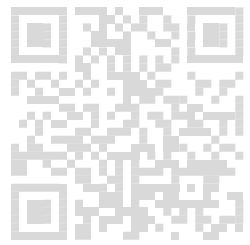
Concepts Included:

strings

basics

companies

Language Used: PYTHON 3



Source Code:

```
s=input().split()  
a = s[::-1]  
print(*a)
```

Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

world hello

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

a h

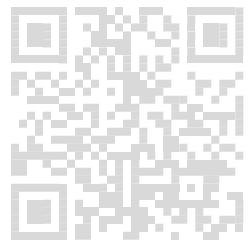
Compilation Status: Passed

Execution Time:

0.009s

33. Given a number N followed by N numbers.Find the smallest number and largest number and print both the indices(1 based indexing).Input Size : N <= 100000Sample Testcase :INPUT51 2 3 4

50OUTPUT1 5



Completion Status: Completed

Concepts Included:

array

basics

Language Used: PYTHON 3

Source Code:

```
n=int(input())
N=input().split()
n=len(N)
A=max(N)
B=min(N)

print((N.index(B)+1),(N.index(A)+1))
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

2 5

Compilation Status: Passed

Execution Time:

0.01s

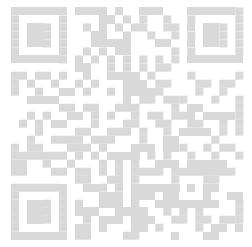
TestCase2:

Input:

< hidden >

Expected Output:

< hidden >



Output:

4 1

Compilation Status: Passed

Execution Time:

0.009s

34. Let P represent Paper, R represent Rock and S represent Scissors. Given 2 out of the 3 determine which one wins. If its a draw print 'D'. Sample Testcase :INPUTR POUTPUTP

Completion Status: Completed

Concepts Included:

strings

basics

Language Used: PYTHON 3

Source Code:

```
N,M=input().split()
if((N=='P' and M =='R') or (N=='R' and M=='P')):
    print("P")
elif((N=='S' and M=='P') or (N=='P' and M=='S')):
    print("S")
elif((N=='R' and M=='S') or (N=='S' and M=='R')):
    print("R")
else:
    print("D")
```

Compilation Details:

TestCase1:

Input:

< hidden >

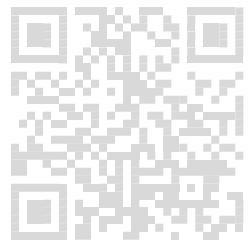
Expected Output:

< hidden >

Output:

D

Compilation Status: Passed



Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

D

Compilation Status: Passed

Execution Time:

0.009s

35. Given 3 numbers A,B,C process and print 'yes' if they can form the sides of a triangle otherwise print 'no'. Input Size : A,B,C <= 100000
Sample Testcase : INPUT 3 4 5 OUTPUT yes

Completion Status: Completed

Concepts Included:

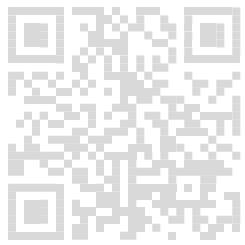
mathematics

basics

Language Used: PYTHON 3

Source Code:

```
a, b, c=input().split(" ")
a=int(a)
b=int(b)
c=int(c)
if c>a and c>b:
    print("yes")
else:
    print("no")
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

no

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

yes

Compilation Status: Passed

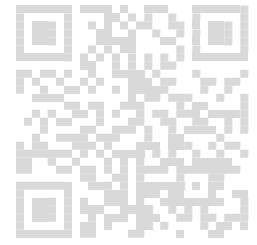
Execution Time:

0.009s

36. Iron Man wants to extract an infinity stone from a safe. The safe is protected by a password and Iron Man knows the clue to the password which is “sum one and two when sorted they are true”. Decode the clue from the test case and help Iron Man open the safe.

Sample Input:

9 8 3 2 1



Sample Output:

3

Completion Status: Completed

Concepts Included:

sorting

array

Language Used: PYTHON 3

Source Code:

```
a = input()
```

```
b = input().split()
```

```
x = sorted([int(X) for X in b ])
# this used to convert array of conerted into list and sorted
print(x[0] + x[1])
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

3

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

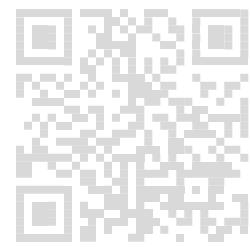
Output:

21

Compilation Status: Passed

Execution Time:

0.009s



37. You are given an array of non-negative integers representing height of walls at index i as A_i and the width of each block is 1. Compute how much air can be encapsulated between the walls of chamber.

Sample Input:

3
7 4 9

Sample Output:

3

Completion Status: Completed

Concepts Included:

array

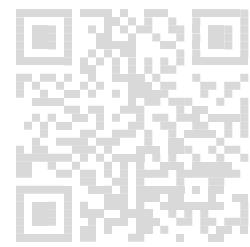
mathematics

Language Used: PYTHON 3

Source Code:

```
n =int(input())
l = [int(x) for x in input().split()]
ans = 0
for i in range(1, len(l)-1):
    if l[i] < l[i-1] and l[i] < l[i+1]:
        ans += min(l[i+1], l[i-1]) - l[i]
```

```
print(ans)
```



Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

3

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

0

Compilation Status: Passed

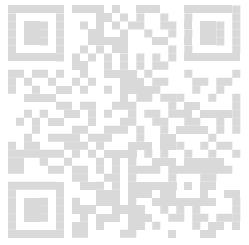
Execution Time:

0.009s

38. You are given with two arrays. Your task is to merge the array such that first array is in ascending order and second one in descending order.

Sample Input:

```
3 3  
23 15 16  
357 65 10
```



Sample Output:

15 16 23 357 65 10

Completion Status: Completed

Concepts Included:

array

Language Used: PYTHON 3

Source Code:

```
n, m = list(map(int,input().split()))
N = list(map(int,input().split()))
M = list(map(int,input().split()))

x=sorted(N)
y=sorted(M,reverse=True)

L = x + y
print(*L)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

15 16 23 357 65 10

Compilation Status: Passed

Execution Time:

0.013s

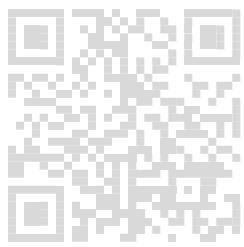
TestCase2:

Input:

< hidden >

Expected Output:

< hidden >



Output:

1 12 13 14 16 19 42 58 65 98 56 55 54 46 32 17 16

Compilation Status: Passed

Execution Time:

0.009s

39. You are given an array of numbers. Print the least occurring element. If there is more than 1 element print all of them in decreasing order of their value.

Sample Input:

9
1 6 4 56 56 56 6 4 2

Sample Output:

2 1

Completion Status: Completed

Concepts Included:

mathematics

array

Language Used: PYTHON 3

Source Code:

```
n = int(input())
arr = list(map(int, input().split()))

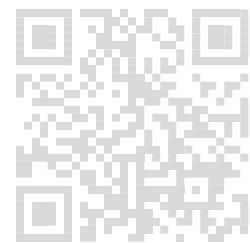
# count the occurrences of each number
freq = {}
for num in arr:
    freq[num] = freq.get(num, 0) + 1

# find the least occurring number
least_freq = min(freq.values())

# collect all numbers with least frequency
result = []
for num, count in freq.items():
    if count == least_freq:
        result.append(num)

print(*result)
```

```
if count == least_freq:  
    result.append(num)  
  
# sort the result in decreasing order  
result.sort(reverse=True)  
  
# print the result  
print(*result)
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

76 25

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

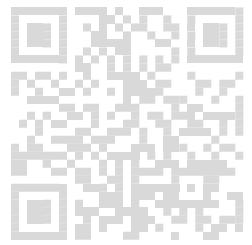
1 0

Compilation Status: Passed

Execution Time:

0.009s

40. Write a code to get the input in the given format and print the output in the given format.



Sample Input:

```
5 3  
1 2 3 4 5
```

Sample Output:

```
5 3  
1 2 3 4 5
```

Completion Status: Completed

Concepts Included:

Input/Output

Language Used: PYTHON 3

Source Code:

```
a = input()  
  
b = input()  
  
print(a)  
  
print(b)
```

Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

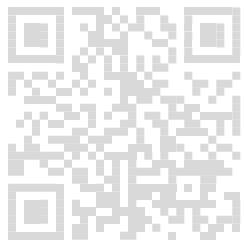
```
5 3  
1 2 3 4 5
```

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:



Input:

< hidden >

Expected Output:

< hidden >

Output:

4 2
1 4 3 2

Compilation Status: Passed

Execution Time:

0.01s

41. Write a code to get the input in the given format and print the output in the given format

Sample Input:

2 4
2 4
2 4

Sample Output:

2 4
2 4
2 4

Completion Status: Completed

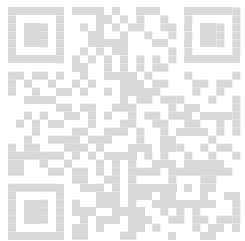
Concepts Included:

Input/Output

Language Used: PYTHON 3

Source Code:

```
a = list(input().split())
b = list(input().split())
c = list(input().split())
```



```
print(*a)  
print(*b)  
print(*c,)
```

Compilation Details:

TestCase1:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

```
2 4  
2 4  
2 4
```

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

```
< hidden >
```

Expected Output:

```
< hidden >
```

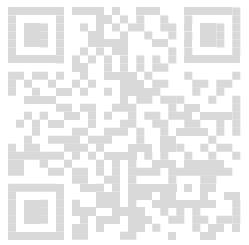
Output:

```
1 3  
2 3  
4 5
```

Compilation Status: Passed

Execution Time:

0.009s



output in the given format

Sample Input:

```
2 5  
2 5 6  
2 4 5
```

Sample Output:

```
2 5  
2 5 6  
2 4 5
```

Completion Status: Completed

Concepts Included:

Input/Output

Language Used: PYTHON 3

Source Code:

```
a = list(input().split())
```

```
b = list(input().split())
```

```
c = list(input().split())
```

```
print(*a)
```

```
print(*b)
```

```
print(*c,)
```

Compilation Details:

TestCase1:

Input:

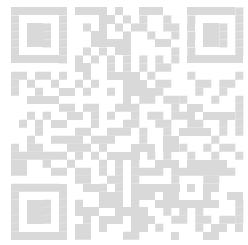
```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

2 5
2 5 6
2 4 5



Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1 2
1 2 4
1 2 3

Compilation Status: Passed

Execution Time:

0.009s

43. Loki wants to steal the tesseract but in order to do so, he has to rearrange the elements in an array in a specific manner which is mentioned in a clue. The clue says ‘cursed are the odd and sorted are the even’. Loki manages to decode the clue which translates to “sort the even positioned elements of an array, starting from the element at index 0, in ascending order”. Manipulate the array so as to help Loki steal the tesseract.

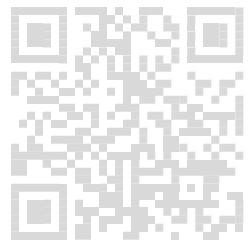
Sample Input:

5
3 9 1 44 6

Sample Output:

1 9 3 44 6

Completion Status: Completed



Concepts Included:

sorting

array

Language Used: PYTHON 3

Source Code:

```
n=int(input())
arr=list(map(int,input().split()))
odd=[]
even=[]
res=[]
for i in range(n):
    if i%2==0 or i==0:
        even.append(arr[i])
        #print(even)
    else:
        odd.append(arr[i])
        #print(odd)
even.sort()
#print(even)

for i in range(len(even)):
    res.append(even[i])
    if i <len(odd):
        res.append(odd[i])
print(*res)
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

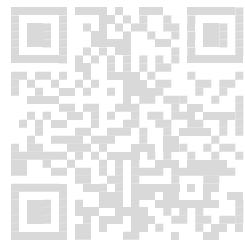
Output:

1 6 3 4 5 2 7

Compilation Status: Passed

Execution Time:

0.01s



TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

23 9 39 5 45 47

Compilation Status: Passed

Execution Time:

0.01s

44. You are given given task is to print whether array is 'majestic' or not.A 'majestic' array is an array whose sum of first three number is equal to last three number.

Sample Input:

7
1 2 3 4 6 0 0

Sample Output:

1

Completion Status: Completed

Concepts Included:

mathematics

array

Amazon

Facebook

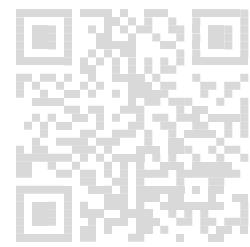
United-Health-Group

guvi-learning-path

Language Used: PYTHON 3

Source Code:

```
n = int(input())
m = list(map(int,input().split()))
a = sum(m[0:3])
b = sum(m[-3:])
if a == b:
    print(1)
else:
    print(0)
```



Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

0

Compilation Status: Passed

Execution Time:

0.009s

45. Your old mobile phone gets broken and so you want to purchase a new smartphone and decide to go through all the online websites to find out which dealer has the best offer for a particular model.

You document the prices of N dealers. Dealer ids start from 0 and go up to N. Find out which dealer has the best price for you.



Constraints:

$1 \leq N \leq 100$

$1 \leq A[i] \leq 100000$

Sample Input:

3
10000 11200 12030

Sample Output:

Dealer0

Completion Status: Completed

Concepts Included:

searching

array

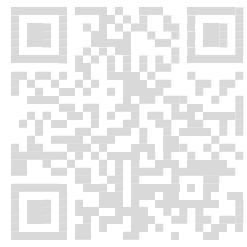
Language Used: PYTHON 3

Source Code:

```
n=int(input())
price=list(map(int,input().split()))
best_p=float("inf")
id=0
for i in range(n):
    if price[i]<best_p:
        best_p=price[i]
        id=i
print("Dealer"+str(id))
```

Compilation Details:

TestCase1:



Input:

< hidden >

Expected Output:

< hidden >

Output:

Dealer9

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Dealer0

Compilation Status: Passed

Execution Time:

0.01s

46. Assume your brother studies in class 2. He has to complete his homework on co-primes. As an elder sibling help him in finding whether the given two numbers is co-prime or not.

Sample Input:

3 5

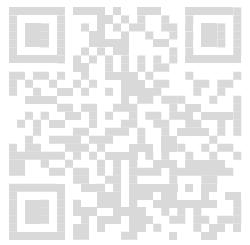
Sample Output:

1

Completion Status: Completed

Concepts Included:

mathematics



Language Used: PYTHON 3

Source Code:

```
from fractions import gcd
n,m=(int(no) for no in input().split())
if gcd(n,m)==1:
    print("1")
else:
    print("0")
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1

Compilation Status: Passed

Execution Time:

0.02s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

1

Compilation Status: Passed

Execution Time:

0.019s

47. In XYZ country there is rule that car's engine no. depends upon car' number plate. Engine no is sum of all the integers present on car's Number plate. The issuing authority has hired you in order to provide engine no. to the cars. Your task is to develop an algorithm which takes input as in form of string(Number plate) and gives back

Engine number.

Sample Input:

HR05-AA-2669

Sample Output:

28

Completion Status: Completed

Concepts Included:

mathematics

strings

Language Used: PYTHON 3

Source Code:

```
import re
x=input()
q=[]
y=re.split("\D+",x)
for i in range(len(y)):

z=list(map(int,"".join(y[i])))
q.append(sum(z))
w=sum(q)
print(w)
```

Compilation Details:

TestCase1:

Input:

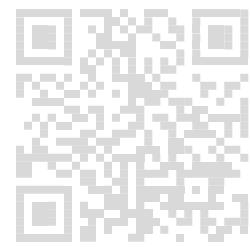
< hidden >

Expected Output:

< hidden >

Output:

28



Compilation Status: Passed

Execution Time:

0.015s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

22

Compilation Status: Passed

Execution Time:

0.015s

48. You are given with a string which comprises of some numbers. Your task is to find the largest integer by converting the string to the corresponding integer.

Sample Input:

I was born on 12 october 1998.

Sample Output:

1998

Completion Status: Completed

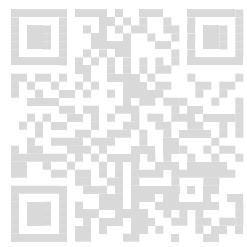
Concepts Included:

mathematics

strings

integer

Language Used: PYTHON 3



Source Code:

```
m = input()

res = ""

a = len(m)

for i in range(0,a):

if m[i]!='.':

res = res+m[i]

n = res.split(" ")

int_lst=[]

for i in n:

if i.isdigit()==True:

int_lst.append(i)

length = len(int_lst)

max_value = 0

for i in range(0,length):

if(int(max_value) < int(int_lst[i])):

max_value = int_lst[i]

print(max_value)
```

Compilation Details:

TestCase1:

Input:

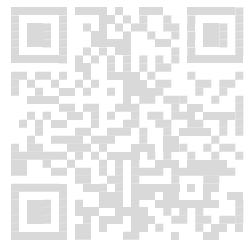
```
< hidden >
```

Expected Output:

```
< hidden >
```

Output:

Compilation Status: Passed



Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

28

Compilation Status: Passed

Execution Time:

0.01s

49. Indian PAN card issuing authority have found some fake PAN cards. They have hired you so that you can validate PAN card for them. Your task is to develop a suitable algorithm which could check if pan is valid or not

1)Pan must have uppercase letters only.

2)It must be of 10 character only

3)From index 1 to 5 all must be letters(A-Z),last index must be letter

4)Rest all must be integer Starting from 1

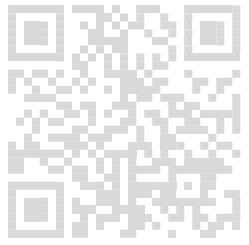
Sample Input:

HXTPS2142R

Sample Output:

pan

Completion Status: Completed



Concepts Included:

strings

Language Used: PYTHON 3

Source Code:

```
n=input()
s=False

if len(n)==10 and n.upper():
    if n[9] not in range(0,9):
        s=True
    if n[0:5].isalnum():
        s=False
    if '0' in n:
        s=False
    if n[5:9].isnumeric():
        s=True

if s:
    print("pan")
else:
    print("not pan")
```

Compilation Details:

TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

pan

Compilation Status: Passed

Execution Time:

0.01s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

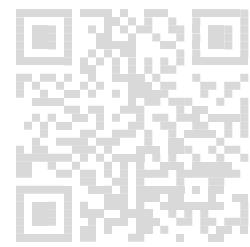
Output:

not pan

Compilation Status: Passed

Execution Time:

0.009s



50. Guvi developed a new system to make sure no two usernames are same. So, they hired you as a developer to develop this system. They have set some rules to do the same.If you see the same username that already exists, just add a number at the end of that username ,else print "Verified".

Sample Input:

4
abc aab abc aba

Sample Output:

Verified Verified abc1 aba

Completion Status: Completed

Concepts Included:

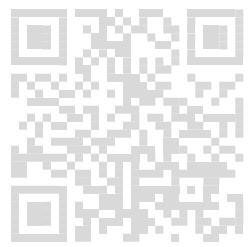
strings

Language Used: PYTHON 3

Source Code:

```
a=int(input())
li=list(input().split(" "))
check=[]
output=[]
for i in li:
if i not in check:
check.append(i)
output.append("Verified")
else:
output.append(i+"1")
print(" ".join(output))
```

Compilation Details:



TestCase1:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Verified Verified abc1 Verified

Compilation Status: Passed

Execution Time:

0.009s

TestCase2:

Input:

< hidden >

Expected Output:

< hidden >

Output:

Verified Verified abc1 Verified

Compilation Status: Passed

Execution Time:

0.009s