

In [1]:

```
#Question 1: List Comprehension
```

```
if __name__ == '__main__':
```

```
    x = int(input())
```

```
    y = int(input())
```

```
    z = int(input())
```

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

```
print([[a, b, c] for a in range(x + 1) for b in range(y + 1) for c in range(z + 1) if a + b + c != n])
```

2

2

2

2

```
[[0, 0, 0], [0, 0, 1], [0, 1, 0], [0, 1, 2], [0, 2, 1], [0, 2, 2], [1, 0, 0], [1, 0, 2], [1, 1, 1], [1, 1, 2], [1, 2, 0], [1, 2, 1], [1, 2, 2], [2, 0, 1], [2, 0, 2], [2, 1, 0], [2, 1, 1], [2, 1, 2], [2, 2, 0], [2, 2, 1], [2, 2, 2]]
```

In [2]:

```
#Question 2: Runner up
```

```
if __name__ == '__main__':
```

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

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

```
    b = set(arr)
```

```
    b.remove(max(b))
```

```
    print(max(b))
```

5

2 3 3 5 6

5

In [3]:

```
#Question 3: Nested Lists
```

```
if __name__ == '__main__':
```

```
    name = [[input(), float(input())] for i in range(int(input()))]
```

```
score = sorted(set(j for i, j in name))[1]
```

```
print("\n".join(sorted(i for i, j in name if j == score)))
```

5

Harry

37.21

Berry

37.21

Tina

37.2

Akriti

41

Harsh

39

Berry

Harry

In [4]:

```
#Question 4: Percentage
```

```
if __name__ == '__main__':
```

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

```
    student_marks = {}
```

```
    for _ in range(n):
```

```
        name, *line = input().split()
```

```
        scores = list(map(float, line))
```

```

        student_marks[name] = scores
    query_name = input()
    l=list(student_marks[query_name])
    le=len(l)
    s=sum(l)
    av=s/le
    print('%.2f'%av)

```

```

3
Krishna 67 68 69
Arjun 70 98 63
Malika 52 56 60
Malika
56.00

```

In [5]:

#Question 5: Lists

```

if __name__ == '__main__':

    def commdr(lst, instruct):
        if instruct[0] == 'insert':
            lst.insert(int(instruct[1]), int(instruct[2]))
        elif instruct[0] == 'print':
            print(lst)
        elif instruct[0] == 'remove':
            lst.remove(int(instruct[1]))
        elif instruct[0] == 'append':
            lst.append(int(instruct[1]))
        elif instruct[0] == 'sort':
            lst.sort()
        elif instruct[0] == 'reverse':
            lst.reverse()
        elif instruct[0] == 'pop':
            lst.pop()
        else:
            print("Command not recognized!")

    N = int(input())
    lst = []
    for command in range(0,N):
        temp = [str(i) for i in input().strip().split()]
        commdr(lst, temp)

```

```

12
insert 0 5
insert 1 10
insert 0 6
print
[6, 5, 10]
remove 6
append 9
append 1
sort
print
[1, 5, 9, 10]
pop
reverse
print
[9, 5, 1]

```

In [8]:

#Question 6: Tuple
#getting different output here but got expected in hackerrank

```

if __name__ == '__main__':
    n = int(input())
    integer_list = map(int, input().split())

    t = tuple(integer_list)
    print(hash(t))

```

```
2
1 2
-3550055125485641917
```

In [13]:

```
#Question 7: Sets
def average(array):
    average=sum(set(arr))/len(set(arr))
    return average

if __name__ == '__main__':
    n = int(input())
    arr = list(map(int, input().split()))
    result = average(arr)
    print(result)
```

```
10
161 162 187 182 212
180.8
```

In [14]:

```
#Question 8: No Idea
n, m = [int(x) for x in input().split()]

array = [int(x) for x in input().split()]
A = {int(x) for x in input().split()}
B = {int(x) for x in input().split()}

happiness = 0

for value in array:
    change = 1 if value in A else -1 if value in B else 0
    happiness += change

print(happiness)
```

```
3 2
1 5 3
3 1
5 7
1
```

In [17]:

```
#Question 9: Set Symmetric_difference
a,i,e,j = input(),input().split(),input(),input().split()
for i in sorted(set(i)^set(j), key = int):

    print(i)
```

```
4
2 4 5 9
4
2 4 11 12
5
9
11
12
```

In [18]:

```
#Question 11: set add
new = set()
for _ in range(int(input())):
    new.add(input())
print(len(new))
```

```
7
UK
```

China
USA
France
New Zealand
UK
France
5

In [21]:

```
#Question 13: Set Union
m = int(input())
s1 = set(input().split());

n = int(input())
s2 = set(input().split());

s3 = s1.union(s2)

print(len(s3))
```

9
1 2 3 4 5 6 7 8 9
9
10 1 2 3 11 21 55 6 8
13

In [23]:

```
#Question 12: Set Remove Pop
int(input())
a = set(map(int, input().split()))
n = int(input())
for i in range(n):
    p = input().split()
    if p[0]=="remove":
        a.remove(int(p[1]))
    elif p[0]=="discard":
        a.discard(int(p[1]))
    else :
        a.pop()
print(sum(list(a)))
```

9
1 2 3 4 5 6 7 8 9
10
pop
remove 9
discard 9
discard 8
remove 7
pop
discard 6
remove 5
pop
discard 5
4

In [24]:

```
#QUESTION 14: set intersection

n1, s1, n2, s2 = (set(input().split()) for i in range(4))
print(len(s1.intersection(s2)))
```

9
1 2 3 4 5 6 7 8 9
9
10 1 2 3 11 21 55 6 8
5

In [25]:

#Question 15: set difference

```
m = int(input())
s1 = set(input().split())

n = int(input())
s2 = set(input().split())

s3 = s1.difference(s2)

print(len(s3))
```

```
9
1 2 3 4 5 6 7 8 9
9
10 1 2 3 11 21 55 6 8
4
```

In [26]:

#Question 16: Set Symmetric difference

```
m = int(input())
s1 = set(input().split())

n = int(input())
s2 = set(input().split())

s3 = s1.symmetric_difference(s2)

print(len(s3))
```

```
9
1 2 3 4 5 6 7 8 9
9
10 1 2 3 11 21 55 6 8
8
```

In [27]:

#Question 17: set mutation

```
n = int(input())
A = set(map(int, input().split()))
N = int(input())
for i in range(N):
    cmd = input().split()
    option = cmd[0]
    s = set(map(int, input().split()))
    if (option == 'update'):
        A |= s
    elif (option == 'intersection_update'):
        A &= s
    elif (option == 'difference_update'):
        A -= s
    elif (option == 'symmetric_difference_update'):
        A ^= s
print(sum(A))
```

```
16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 24 52
4
intersection_update 10
2 3 5 6 8 9 1 4 7 11
update 2
55 66
symmetric_difference_update 5
22 7 35 62 58
difference_update 7
11 22 35 55 58 62 66
38
```

In [31]:

```
#Question 18 Captain room
k=int(input())
rs=list(map(int,input().split()))
a=set()
b=set()
for r in rs:
    if r not in a:
        a.add(r)
        b.add(r)
    else:
        b.discard(r)
b=list(b)
print("The Captain's Room:",b[0])
```

```
5
1 2 3 6 5 4 4 2 5 3 6 1 6 5 3 2 4 1 2 5 1 4 3 6 8 4 3 1 5 6 2
The Captain's Room: 8
```

In [32]:

```
#Question 19: check subset
t = int(input())

for _ in range(t):
    x = input()
    A = set(input().split())
    y = int(input())
    B = set(input().split())
    print(A.issubset(B))
```

```
3
5
1 2 3 5 6
9
9 8 5 6 3 2 1 4 7
True
1
2
5
3 6 5 4 1
False
7
1 2 3 5 6 8 9
3
9 8 2
False
```

In [33]:

```
#Question 20: strict superset

def check():
    s=set(map(int,input().split()))
    if s.issubset(a):
        if len(a)==len(s):
            l.append(0)
        else:
            l.append(1)
    else:
        l.append(0)

a=set(map(int,input().split()))
n=int(input())
l=[]
for i in range(n):
    check()
if all(l)==1:
    print(True)
```

```
else:  
    print(False)
```

```
1 2 3 4 5 6 7 8 9 10 11 12 23 45 84 78  
2  
1 2 3 4 5  
100 11 12  
False
```

```
In [ ]:
```