Q.1 List Comprehensions

Let's learn about list comprehensions! You are given three integers and representing the dimensions of a cuboid along with an integer . Print a list of all possible coordinates given by on a 3D grid where the sum of i+j+K is not equal to n.Here, 0<=i<=x,0<=j<=y,0<=k<=z.Please use list comprehensions rather than multiple loops, as a learning exercise.

Example

$x = 1$
$y = 1$
$z = 2$
$n = 3$

All permutations of $[i, j, k]$ are:
$[[0, 0, 0], [0, 0, 1], [0, 0, 2], [0, 1, 0], [0, 1, 1], [0, 1, 2], [1, 0, 0], [1, 0, 1], [1, 0, 2], [1, 1, 0], [1, 1, 1], [1, 1, 2]]$.

Print an array of the elements that do not sum to $n = 3$.

$[[0, 0, 0], [0, 0, 1], [0, 0, 2], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 2]]$

Input Format

Four integers x, y, z and n, each on a separate line.

Constraints

Print the list in lexicographic increasing order.

Sample Input 0

```
1
1
1
2
```

Sample Output 0

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

In [5]:
```
print('Input')
x = int(input())
y = int(input())
z = int(input())
```

```
n = int(input())
print('Output \n')
print ([[i,j,k] for i in range(x+1)for j in range(y+1)for k in range (z+1)if((i+j+k)!= n )] )
```

```
Input
1
1
1
2
Output

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

Q.2 Find the Runner-Up Score!

Given the participants' score sheet for your University Sports Day, you are required to find the runner-up score. You are given scores. Store them in a list and find the score of the runner-up.

Sample Input 0

```
5
2 3 6 6 5
```

Sample Output 0

```
5
```

In [10]:
```
print('Total Scores:')
n = int(input())
score = list(map(int, input().split()))
print('\nRunner-up Score:')
print(sorted(list(set(score)))[-2])
```

```
Total Scores:
5
20 30 40 50 21

Runner-up Score:
40
```

Q.3 Lists

Consider a list (list = []). You can perform the following commands:

1. insert i e : Insert integer at position .
2. print: Print the list.
3. remove e : Delete the first occurrence of integer .
4. append e : Insert integer at the end of the list.
5. sort: Sort the list.
6. pop : Pop the last element from the list.
7. reverse : Reverse the list.

Initialize your list and read in the value of followed by lines of commands where each command will be of the types listed above. Iterate through each command in order and perform the corresponding operation on your list.

Input Format

The first line contains an integer,N, denoting the number of commands. Each line of the subsequent lines contains one of the commands described above.

Constraints

The elements added to the list must be integers.

Output Format

For each command of type print, print the list on a new line.

Sample Input 0

```
12
insert 0 5
insert 1 10
insert 0 6
print
remove 6
append 9
```

```
append 1
sort
print
pop
reverse
print
```

Sample Output 0

```
[6, 5, 10]
[1, 5, 9, 10]
[9, 5, 1]
```

In [1]:

```
print('element range in list')
N = int(input())
list=[]
for i in range(N):
    operation=input().split()
    if operation[0]=='insert':
        list.insert(int(operation[1]),int(operation[2]))
    elif operation[0]=='print':
        print('\n',list,'\n')
    elif operation[0]=='append':
        list.append(int(operation[1]))
    elif operation[0]=='sort':
        list.sort()
    elif operation[0]=='remove':
        list.remove(int(operation[1]))
    elif operation[0]=='pop':
        list.pop()
    else:
        list.reverse()
```

```
element range in list
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]
```

Q.4 Nested Lists

Given the names and grades for each student in a Physics class of students, store them in a nested listand print the name(s) of any student(s) having the second lowest grade.

Note: If there are multiple students with the same grade, order their names alphabetically and print eachname on a new line.

Input Format

The first line contains an integer,N , the number of students. The 2N subsequent lines describe each student over 2 lines; the first line contains a student's name, andthe second line contains their grade.

There will always be one or more students having the second lowest grade.

Output Format

Print the name(s) of any student(s) having the second lowest grade in Physics; if there are multiple students, order their names alphabetically and print each one on a new line.

Sample Input 0

```
5
Harry
37.21
```

```
Berry
37.21
Tina
37.2
Akriti
41
Harsh
39
```

Sample Output 0

```
Berry
Harry
```

In [2]:

```
print('Total number of students')
number_of_students=int(input())
scorecard={}

for i in range(number_of_students):
    name=str(input())
    marks=float(input())
    scorecard[name]=[marks]

score=sorted(list(scorecard.values()))

second_lowest_score=score[1]
second_lowest_student=[]

for names,marks in scorecard.items():
    if (marks==second_lowest_score):
        second_lowest_student+=[names]
second_lowest_student.sort()
print('\nOutput\n')
for student in second_lowest_student:
    print(student)
```

```
Total number of students
5
shubham
11.5
shub
12.5
```

```
mahesh
12.5
suresh
20
rakesh
18

Output

mahesh
shub
```

Q.5 Finding the percentage

You have a record of N students. Each record contains the student's name, and their percent marks in Maths, Physics and Chemistry. The marks can be floating values. The user enters some integer N followed by the names and marks for N students. You are required to save the record in a dictionary data type. The user then enters a student's name. Output the average percentage marks obtained by that student, correct to two decimal places.

Input Format

The first line contains the integer N, the number of students. The next N lines contains the name and marks obtained by that student separated by a space. The final line contains the name of a particular student previously listed.

Constraints

- $2 \leq N \leq 10$
- $0 \leq Marks \leq 100$

Output Format

Print one line: The average of the marks obtained by the particular student correct to 2 decimal places.

Sample Input 0

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

Sample Output 0

```
56.00
```

Explanation 0

Marks for Malika are $\{52, 56, 60\}$ whose average is $\frac{52+56+60}{3} \Rightarrow 56$

Sample Input 1

```
2
Harsh 25 26.5 28
Anurag 26 28 30
Harsh
```

Sample Output 1

```
26.50
```

In [3]:

```
print('Number of students:')
n = int(input())
student_marks = {}

print('\nStudent Names and marks:')
for i in range(n):
    name, *line = input().split()
    scores = list(map(float, line))
```

```
    student_marks[name] = scores

print('\nName of student whose avg score wants:')
query_name = input()

avg=sum(student_marks[query_name])/len(student_marks[query_name])

print('\nAvg score of {}:'.format(query_name))
print('%.2f'%avg)
```

```
Number of students:
3

Student Names and marks:
A 30 40 50
B 40 50 60
c 10 40 60

Name of student whose avg score wants:
A

Avg score of A:
40.00
```

Q.6 Check Subset

You are given two sets,A and B. Your job is to find whether set A is a subset of set B.

```
If set A is subset of set B, print True.
If set A is not a subset of set B, print False.
```

Input Format

```
The first line will contain the number of test cases,T .
The first line of each test case contains the number of elements in set A .
The second line of each test case contains the space separated elements of set A .
The third line of each test case contains the number of elements in set B .
The fourth line of each test case contains the space separated elements of set B .
```

Output Format

Output True or False for each test case on separate lines.

In [2]:
```
print('Total number of test cases:')
T=int(input())

for i in range (T):
    print('\nNumber of elements in set A:')
    a =input()
    print('\nelements of set A:')
    A = set(input().split())
    print('\nNumber of elements in set B:')
    b = int(input())
    print('\nelements of set B:')
    B = set(input().split())
    print('\nIs A is Subset of B')
    print(A.issubset(B))
```

```
Total number of test cases:
1

Number of elements in set A:
5

elements of set A:
1 2 3 4 5

Number of elements in set B:
5

elements of set B:
4 5 6 7 8

Is A is Subset of B
False
```

Q.7 Check the strict Superset

You are given a set A and n other sets.Your job is to find whether set A is a strict superset of each of the n sets.

Print True, if A is a strict superset of each of the n sets. Otherwise, print False.

A strict superset has at least one element that does not exist in its subset.

In [8]:
```
print('Parent Set:')
A=set(input().split())
print('\nTotal Number of Child set:')
N=int(input())

for i in range(N):
    B=set(input().split())

print('\nPrint True if Parent set is Superset of all child sets else print False')
print(A.issuperset(B))
```

```
Parent Set:
1 2 3 4 5

Total Number of Child set:
3
1 2 3
3
4 5

Print True if Parent set is Superset of all child sets else print False
True
```

Q.8 Tuples

Given an integer,n ,n and space-separated integers as input, create a tuple,t , of those n integers. Then compute and print the result of hash(t).

Note: hash() is one of the functions in the **builtins** module, so it need not be imported.

Input Format

The first line contains an integer, n, denoting the number of elements in the tuple. The second line contains n space-separated integers describing the elements in tuple t.

Output Format

Print the result of .

Sample Input 0

```
2
1 2
```

Sample Output 0

3713081631934410656

In [7]:
```
n = int(input())
integer_list = map(int,input().split())
t=tuple(integer_list)
print(hash(t))
```

```
4
1 2 3 4
590899387183067792
```

Q.9 Introduction to Sets

A set is an unordered collection of elements without duplicate entries.When printed, iterated or converted into a sequence, its elements will appear in an arbitrary order.

Basically, sets are used for membership testing and eliminating duplicate entries.

Task

Now, let's use our knowledge of sets and help Mickey.

Ms. Gabriel Williams is a botany professor at District College. One day, she asked her student Mickey to compute the average of all the plants with distinct heights in her greenhouse.

Function Description

Complete the average function in the editor below.

average has the following parameters:

int arr: an array of integers Returns

float: the resulting float value rounded to 3 places after the decimal

Input Format

The first line contains the integer, , the size of . The second line contains the space-separated integers, .

Sample Input

```
STDIN                                       Function
-----                                       --------
 10                                         arr[] size N = 10
161 182 161 154 176 170 167 171 170 174     arr = [161, 181, ..., 174]
```

Sample Output

```
169.375
```

In [4]:
```
def average(array):
    # your code goes here
    array=set(array)
    average=sum(array)/len(array)
    return average

print('Total Number of Plants')
n = int(input())
print('\nHeights of plants')
arr = list(map(int, input().split()))
result = average(arr)
print('\nAverage Height')
print(result)
```

```
Total Number of Plants
10

Heights of plants
120 100 130 145 155 165 170 140 120 100
```

```
Average Height
140.625
```

Q.10 Symmetric Difference

Given 2 sets of integers, M and N , print their symmetric difference in ascending order. The term symmetric difference indicates those values that exist in either M or N but do not exist in both.

Input Format

The first line of input contains an integer,M. The second line contains M space-separated integers. The third line contains an integer, N. The fourth line contains N space-separated integers.

Output Format

Output the symmetric difference integers in ascending order, one per line.

Sample Input

```
STDIN       Function
-----       --------
4           set a size M = 4
2 4 5 9     a = {2, 4, 5, 9}
4           set b size N = 4
2 4 11 12   b = {2, 4, 11, 12}
```

Sample Output

```
5
9
11
12
```

In [6]:
```
print('M Set')
M=int(input())
M=set(map(int,input().split()))
```

```
print('\nN Set')
N=int(input())
N=set(map(int,input().split()))

Mdef=M.difference(N)
Ndef=N.difference(M)

diff_value=Mdef.union(Ndef)

print('\nOutput')
for i in sorted(list(diff_value)):
    print(i)
```

```
M Set
4
2 4 5 9

N Set
4
2 4 11 12

Output
5
9
11
12
```

Q.11 Set .discard().remove() & .pop()

You have a non-empty set s , and you have to execute N commands given in N lines.

The commands will be pop, remove and discard.

Input Format

The first line contains integer n, the number of elements in the set s . The second line contains n space separated elements of set s. All of the elements are non-negative integers, less than or equal to 9. The third line contains integer N, the number of commands. The next N lines contains either pop, remove and/or discard commands followed by their associated value.

Output Format

Print the sum of the elements of set s on a single line.

Sample Input

```
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
```

Sample Output

```
4
```

In [2]:

```
print('Number of Element in Set')
n = int(input())
set = set(map(int, input().split()))

print('\nTotal Numbersof  Operations')
N=int(input())

for i in range (N):
    operation=input().split()
    if operation[0]=='pop':
        set.pop()
    elif operation[0]=='remove':
        set.remove(int(operation[1]))
    elif operation[0]=='discard':
        set.discard(int(operation[1]))
    else:
        pass
```

```
print('\nSum of Set')
print(sum(set))
```

```
Number of Element in Set
9
1 2 3 4 5 6 7 8 9

Total Numbersof  Operations
10
pop
remove 9
discard 9
discard 8
remove 7
pop
discard 6
remove 5
pop
discard 5

Sum of Set
4
```

Q.12 No Idea!

There is an array of n integers. There are also **2 disjoint sets**, A and B, each containing m integers. You like all the integers in set A and dislike all the integers in set B. Your initial happiness is 0. For each i integer in the array, if $i \in A$, you add 1 to your happiness. If $i \in B$, you add -1 to your happiness. Otherwise, your happiness does not change. Output your final happiness at the end.

Note: Since A and B are sets, they have no repeated elements. However, the array might contain duplicate elements.

Constraints

$1 \leq n \leq 10^5$
$1 \leq m \leq 10^5$
$1 \leq Any\ integer\ in\ the\ input \leq 10^9$

Input Format

The first line contains integers n and m separated by a space.
The second line contains n integers, the elements of the array.
The third and fourth lines contain m integers, A and B, respectively.

Output Format

Output a single integer, your total happiness.

Sample Input

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

Sample Output

```
1
```

Explanation

You gain 1 unit of happiness for elements 3 and 1 in set A. You lose 1 unit for 5 in set B. The element 7 in set B does not exist in the array so it is not included in the calculation.

In [10]:
```
print('Containing total number of  m and n integer')
x=input().split()
m=int(x[0])
n=int(x[1])

print('\narray containing m integer element')
```

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

print('\nSet A and B containing n integer element')
A=set(map(int,input().split()))
B=set(map(int,input().split()))

count_of_happiness=0

for i in arr:
    if i in A:
        count_of_happiness=count_of_happiness+1
    if i in B:
        count_of_happiness=count_of_happiness-1

print('\nTotal Happiness')
print(count_of_happiness)
```

```
Containing total number of  m and n integer
3 2

array containing m integer element
1 3 5

Set A and B containing n integer element
1 3
5 7

Total Happiness
1
```

Q.13 Set.add()

Apply your knowledge of the .add() operation to help your friend Rupal. Rupal has a huge collection of country stamps. She decided to count the total number of distinct country stamps in her collection. She asked for your help. You pick the stamps one by one from a stack of country stamps.

Find the total number of distinct country stamps.

Sample Input

```
7
UK
China
```

```
USA
France
New Zealand
UK
France
```

Sample Output

```
5
```

In [2]:
```
print('Total COuntry Count')
total_Country_stamp=int(input())
Country=set()

print('\nCountry')
for i in range(total_Country_stamp):
    Country.add(input())

print('\ntotal number of distinct country stamps are')
print(len(Country))
```

```
Total COuntry Count
7

Country
INDIA
UK
USA
GERMANY
FRANCE
INDIA
NZ

total number of distinct country stamps are
6
```

Q.14 Set.union() (|)

The students of District College have subscriptions to English and French newspapers. Some students have subscribed only to English, some have subscribed to only French and some have subscribed to both newspapers.

You are given two sets of student roll numbers. One set has subscribed to the English newspaper, and the other set is subscribed to the French newspaper. The same student could be in both sets. Your task is to find the total number of students who have subscribed to at least one newspaper.

In [1]:
```
print('Total count of roll number of Students Subscribed for English Newspaper')
n1=int(input())
print('\nroll number of Students Subscribed for English Newspaper')
A=set(input().split())
print('\nTotal count of roll number of Students Subscribed for French Newspaper')
n2=int(input())
print('\nroll number of Students Subscribed for French Newspaper')
B=set(input().split())

Total_students=A.union(B)
#Total_students=A | B

print('\nTotal number of students who have subscribed to at least one newspaper.')
print(len(Total_students))
```

```
Total count of roll number of Students Subscribed for English Newspaper
9

roll number of Students Subscribed for English Newspaper
1 2 3 4 5 6 7 8 9

Total count of roll number of Students Subscribed for French Newspaper
9

roll number of Students Subscribed for French Newspaper
10 1 2 3 11 21 55 6 8

Total number of students who have subscribed to at least one newspaper.
13
```

Q.15 Set.intersection() (&)

The students of District College have subscriptions to English and French newspapers. Some students have subscribed only to English, some have subscribed only to French, and some have subscribed to both newspapers.

You are given two sets of student roll numbers. One set has subscribed to the English newspaper, one set has subscribed to the French newspaper. Your task is to find the total number of students who have subscribed to both newspapers.

In [2]:
```
print('Total count of Students Subscribed for English Newspaper')
n1=int(input())
print('\nroll number of Students Subscribed for English Newspaper')
A=set(input().split())
print('\nTotal count of Students Subscribed for French Newspaper')
n2=int(input())
print('\nroll number of Students Subscribed for French Newspaper')
B=set(input().split())

Total_students=A.intersection(B)
#Total_students=A & B

print('\nTotal number of students who have subscribed to both newspaper.')
print(len(Total_students))
```

```
Total count of Students Subscribed for English Newspaper
9

roll number of Students Subscribed for English Newspaper
1 2 3 4 5 6 7 8 9

Total count of Students Subscribed for French Newspaper
9

roll number of Students Subscribed for French Newspaper
10 2 3 1 11 21 55 6 8

Total number of students who have subscribed to both newspaper.
5
```

Q.16 Set.Difference() (-)

Students of District College have a subscription to English and French newspapers. Some students have subscribed to only the English newspaper, some have subscribed to only the French newspaper, and some have subscribed to both newspapers.

You are given two sets of student roll numbers. One set has subscribed to the English newspaper, and one set has subscribed to the French newspaper. Your task is to find the total number of students who have subscribed to only English newspapers.

In [1]:
```
print('Total count of Students Subscribed for English Newspaper')
n1=int(input())
print('\nroll number of Students Subscribed for English Newspaper')
English=set(input().split())
print('\nTotal count of Students Subscribed for French Newspaper')
```

```
n2=int(input())
print('\nroll number of Students Subscribed for French Newspaper')
French=set(input().split())

Total_students=English.difference(French)
#Total_students=English-French

print('\nTotal number of students who have subscribed to only English newspapers.')
print(len(Total_students))
```

```
Total count of Students Subscribed for English Newspaper
9

roll number of Students Subscribed for English Newspaper
1 2 3 4 5 6 7 8 9

Total count of Students Subscribed for French Newspaper
9

roll number of Students Subscribed for French Newspaper
10 2 3 1 11 21 55 6 8

Total number of students who have subscribed to only English newspapers.
4
```

Q.17 Set.Symmetric_difference() (^)

Students of District College have subscriptions to English and French newspapers. Some students have subscribed to English only, some have subscribed to French only, and some have subscribed to both newspapers.

You are given two sets of student roll numbers. One set has subscribed to the English newspaper, and one set has subscribed to the French newspaper. Your task is to find the total number of students who have subscribed to either the English or the French newspaper but not both.

In [3]:
```
print('Total count of Students Subscribed for English Newspaper')
n1=int(input())
print('\nroll number of Students Subscribed for English Newspaper')
English=set(input().split())
print('\nTotal count of Students Subscribed for French Newspaper')
n2=int(input())
print('\nroll number of Students Subscribed for French Newspaper')
French=set(input().split())

#Total_students=English.symmetric_difference(French)
```

```
Total_students=English^(French)

print('\nTotal number of students who have subscribed to either the English or the French newspaper but not both.')
print(len(Total_students))
```

```
Total count of Students Subscribed for English Newspaper
9

roll number of Students Subscribed for English Newspaper
1 2 3 4 5 6 7 8 9

Total count of Students Subscribed for French Newspaper
9

roll number of Students Subscribed for French Newspaper
10 2 3 1 11 21 55 6 8

Total number of students who have subscribed to either the English or the French newspaper but not both.
8
```

Q.18 Set Mutations

You are given a set A and N number of other sets. These N number of sets have to perform some specific mutation operations on set A .

Your task is to execute those operations and print the sum of elements from set A .

Sample Input

```
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
```

Sample Output

38

In [1]:
```
print('Total no of element in Set A')
N1=int(input())
print('\nelement of Set A')
A=set(map(int,input().split()))

print('\nTotal Number of Other Sets')
N2=int(input())

for i in range (N2):
    operation=input().split()
    B=set(map(int,input().split()))
    eval('A.{}({})'.format(operation[0],B))

print('\nSum of Updated Elements in Set A')
print(sum(A))
```

```
Total no of element in Set A
16

element of Set A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 24 52

Total Number of Other Sets
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

Sum of Updated Elements in Set A
38
```

Q.19 The Captain's Room

Mr. Anant Asankhya is the manager at the INFINITE hotel. The hotel has an infinite amount of rooms. One fine day, a finite number of tourists come to stay at the hotel.

The tourists consist of:

→ A Captain.
→ An unknown group of families consisting of K members per group where K≠1 .

The Captain was given a separate room, and the rest were given one room per group.

Mr. Anant has an unordered list of randomly arranged room entries. The list consists of the room numbers for all of the tourists. The room numbers will appear K times per group except for the Captain's room.

Mr. Anant needs you to help him find the Captain's room number.

The total number of tourists or the total number of groups of families is not known to you.

You only know the value of K and the room number list.

Input Format

The first line consists of an integer,K, the size of each group.

The second line contains the unordered elements of the room number list.

Output Format

Output the Captain's room number.

Sample Input

```
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
```

Sample Output

```
8
```

In [3]:
```
print('Size of Group')
group_size = int(input())
print('\nRoom Number List of each Member')
room_number_list=list(map(int,input().split()))

room_number=set()
Duplicate=set()

for i in room_number_list:
    if i not in room_number:
        room_number.add(i)
    else:
        Duplicate.add(i)

Captain_room= (room_number.difference(Duplicate)).pop()
print('\nCaptain Room Number is:',Captain_room)
```

```
Size of Group
5

Room Number List of each Member
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

Captain Room Number is: 8
```

In [ ]: