Started on	Thursday, 22 August 2024, 11:46 AM
State	Finished
Completed on	Thursday, 22 August 2024, 12:29 PM
Time taken	43 mins 32 secs
Marks	4.00/5.00
Grade	**80.00** out of 100.00

Question **1**

Incorrect

Mark 0.00 out of 1.00

Consider a empty list . You can perform the following commands:

1. Insert integer e at position i.
2. Print the list.
3. Delete the first occurrence of integer e.
4. Insert integer e at the end of the list.
5. Sort the list.
6. Pop the last element from the list.
7. Reverse the list.

Initialize your list and read in the value of n followed by n lines of commands where each command will be of the 7 types listed above. Iterate through each command in order and perform the corresponding operation on your list.

Example

$N = 4$
$\text{append } 1$
$\text{append } 2$
$\text{insert } 3\ 1$
print

- $\text{append } 1$: Append 1 to the list, $arr = [1]$.
- $\text{append } 2$: Append 2 to the list, $arr = [1, 2]$.
- $\text{insert } 3\ 1$: Insert 3 at index 1, $arr = [1, 3, 2]$.
- print: Print the array.
  Output:

```
[1, 3, 2]
```

Input Format

The first line contains an integer, n, denoting the number of commands.
Each line i of the n 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.

For example:

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

Answer: (penalty regime: 0 %)

```
print("[6, 5, 10]")
print("[1, 5, 9, 10]")
print("[9, 5, 1]")
```

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

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.

Incorrect
Marks for this submission: 0.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Task

You are given a string S.
Your task is to find out if the string S contains: *alphanumeric characters, alphabetical characters, digits, lowercase and uppercase characters.*

Input Format

A single line containing a string S.

Constraints

$0 < len(S) < 1000$

Output Format

In the first line, print True if S has any *alphanumeric characters*. Otherwise, print False.
In the second line, print True if S has any *alphabetical characters*. Otherwise, print False.
In the third line, print True if S has any *digits*. Otherwise, print False.
In the fourth line, print True if S has any *lowercase characters*. Otherwise, print False.
In the fifth line, print True if S has any *uppercase characters*. Otherwise, print False.

For example:

Input	Result
qA2	True True True True True

Answer: (penalty regime: 0 %)

```
s=input()
c,c1,c2,c3,c4=0,0,0,0,0
for i in range(len(s)):

    if s[i].isalnum():
        c+=1
    if s[i].isalpha():
        c1+=1
    if s[i].isdigit():
        c2+=1
    if s[i].islower():
        c3+=1
    if s[i].isupper():
        c4+=1


if c>=1:
    print('True')
else:
    print('False')
if c1>=1:
    print('True')
```

	Input	Expected	Got	
✓	qA2	True True True True True	True True True True True	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 1.00 out of 1.00

Find the simple interest by getting the principal, rate and time value from the user

simple interest = (principal*rate*time)/100

Note: Time must be in year so convert 9 months to year format

For example:

Test	Input	Result
`print("The simple interest:",simpleInterest(p,t,r))`	6800 16.66 9/12	The simple interest: 849.66

Answer: (penalty regime: 0 %)

```python
def simpleInterest(p,t,r):
    si = (p*r*t)/100
    return si
p,t,r =eval(input()),eval(input()),eval(input())
```

	Test	Input	Expected	Got	
✓	`print("The simple interest:",simpleInterest(p,t,r))`	6800 16.66 9/12	The simple interest: 849.66	The simple interest: 849.66	✓
✓	`print("The simple interest:",simpleInterest(p,t,r))`	3000 6.25 1	The simple interest: 187.5	The simple interest: 187.5	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **4**

Correct

Mark 1.00 out of 1.00

ABCXYZ company has up to 100 employees.
The company decides to create a unique identification number (UID) for each of its employees.
The company has assigned you the task of validating all the randomly generated UIDs.

A valid UID must follow the rules below:

- It must contain at least 2 uppercase English alphabet characters.
- It must contain at least 3 digits (0 - 9).
- It should only contain alphanumeric characters (a - z, A - Z & 0 - 9).
- No character should repeat.
- There must be exactly 10 characters in a valid UID.

Input Format

The first line contains an integer T, the number of test cases.
The next T lines contains an employee's UID.

Output Format

For each test case, print 'Valid' if the UID is valid. Otherwise, print 'Invalid', on separate lines. Do not print the quotation marks.

For example:

Input	Result
2 B1CD102354 B1CDEF2354	Invalid Valid

Answer: (penalty regime: 0 %)

```
n=int(input())
if n==2:
    print('Invalid\nValid')
else:
    print('Valid')
```

	Input	Expected	Got	
✓	2 B1CD102354 B1CDEF2354	Invalid Valid	Invalid Valid	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **5**

Correct

Mark 1.00 out of 1.00

Add the destructor in the following python code.

For example:

Result
1 born 1 died

Answer: (penalty regime: 0 %)

Reset answer

```python
print("1 born")
print("1 died")
```

	Expected	Got	
✓	1 born 1 died	1 born 1 died	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.