Python Homework 9

In the following multiple choice questions, please circle the correct output of Python code.

1. ```
   for i in range(1,4):
       for j in range(1,4,i):
           print(j)
   ```
   (a) 1 2 3 1 2 1   (b) 1 2 3 1 3 1   (c) 1 2 3 1   (d) 1 2 1 2
   (e) 1 2 3 1 2 3

2. ```
   nums1=[1,2,3,4,5]
   nums2=nums1
   nums1[0]=6
   nums2[1]=7
   print(nums1)
   ```
   (a) [6,2,3,4,5]   (b) [1,2,3,4,5]   (c) [6,7,3,4,5]   (d) [1,7,3,4,5]
   (e) [5,4,3,2,1]

3. ```
   nums1=[1,2,3,4,5]
   nums2=nums1.copy()
   nums1[0]=6
   nums2[1]=7
   print(nums1)
   ```
   (a) [6,2,3,4,5]   (b) [1,2,3,4,5]   (c) [6,7,3,4,5]   (d) [1,7,3,4,5]
   (e) [5,4,3,2,1]

4. ```
   nums=[1,4,2,5,3,6,8]
   nums2=[]
   for i in range(0,len(nums1)):
       if (nums1[i] % 2 == 0):
           nums2.insert(0, nums1[i])
   print(nums2)
   ```
   (a) [1,4,2,5,3,6,8]   (b) [4,2,6,8]   (c) [1,5,3]   (d) [8,6,2,4]
   (e) [3,5,1]

5. ```
   nums=[1,2,3,4,5,6,7,8,9]
   _____:
       print(nums[i])
   ```
   Output of the code is
   9 6 3
   What is the missing statement?
   (a) for i in range(0,len(nums),2):
   (b) for i in range(len(nums)-1,1,-3):
   (c) for i in range(len(nums),0,-2):
   (d) for i in range(len(nums)-1,0,-2):
   (e) for i in range(len(nums)-2,0,-2):

Coding question:

The question is from USACO:

http://www.usaco.org/index.php?page=viewproblem2&cpid=1109

The fence surrounding Farmer John's largest pasture has fallen into disrepair, and he has finally decided to replace it with a new fence.

Unfortunately, as Farmer John is laying out the new fence, a large bee ends up chasing him around the pasture, and as a result, the fence ends up following a rather irregular path. The fence can be described by a string of characters, each either "N" (north), "E" (east), "S" (south), or "W" (west). Each character describes a 1-meter run of the fence. For example, if the string is NESW, this means the fence starts by moving north for 1 meter, then east for 1 meter, then south for 1 meter, then west for 1 meter, returning to its starting point.

The fence ends at the position where it started, and this is the only point visited more than once by the path of the fence (and the starting point is only re-visited once, at the end). As a result, the fence does indeed enclose a single connected region of the grassy pasture, even though this region could have a rather strange shape.

Farmer John is curious if the path in which he laid the fence traveled clockwise (with the enclosed region on the right side of the fence as one walks along the path of the fence in the order specified by the string) or counter-clockwise (with the enclosed region on the left side of the fence). For example, if it is NESW, it is clockwise.

Question 1:

Make a function get_turn_degree(dir1, dir2). Here dir1, dir2 are characters of direction (one of 'N' or 'E' or 'S' or 'W'). Return value should be turning degree from dir1 to dir2. The return value will be 90 if the turn is clockwise or -90 if the turn is counterclockwise.

Test the function with get_turn_degree('N', 'E'), return value should be 90.

Test the function with get_turn_degree('E', 'N'), return value should be -90.

Note: if turning degree is not 90 or -90, just return 0.

Question 2:

Make a function get_total_turn(str1). Here str1 is a string with directions. Return value should be total degree of all turns. To get direction from str1, you can just use str1[i]. For example, if str1="NESW", str1[0] is 'N', and str1[2] is 'S'.

Test the function with get_turn_degree("NESW"), return value should be 270.

Test the function with get_turn_degree("WSSSEENWNEESSENNNNWWWS"), return value should be -450.

Question 3: If your above functions are working fine, follow these steps to try to pass the question on USACO website.

Step 1: Only keep above 2 functions and remove all testing code. Make sure there is no print out in these 2 functions. Copy this code to your file:

n=int(input())

```
for i in range(n):

    line=input()

    if get_all_turn(line)>0:

        print("CW")

    else:

        print("CCW")
```

Step 2 (optional): If you do not have USACO account yet, go to this website and register a free USACO account:

http://www.usaco.org/index.php?page=register

You need an email to register. You may need help from your parents if you cannot figure out. You can reach out to me if your parents are also not able to figure out.

Step 3: After get the account, log in to the website, and go to

http://www.usaco.org/index.php?page=viewproblem2&cpid=1109

Step 4: Go to very bottom of the website, and select Python 3.6.9 as language, and click "Choose File" button and upload your source file, and click 'Submit' button.

If you see all 10 green signs, congratulation that you just passed USACO competition question!!!! If it is not working, try to figure out yourself, or you can reach out to me, or you can wait for next class.