DEPAUL UNIVERSITY

Global Learning Experience

UKRAINIAN CATHOLIC UNIVERSITY

Assignment 3

Due Wednesday October 16 at 6p.m. Chicago time for DePaul students

and

Due Tuesday October 15 at midnight Kyiv time for UCU students

You have all been assigned to mixed teams of DePaul and UCU students, based on your availability. You are expected to meet synchronously on Zoom/Teams for at least 1 hour this week. The exact day and time (in your respective time zone) for the synchronous meeting depends on your group. If you are not clear on when your group is supposed to meet, please check your Discord channel as soon as possible. As a last resort, reach out to Prof. Besana or Prof Zakharchenko.

TASKS

Here you find three pieces of Python code (a .py file with the code is also available on your learning management system). Each piece solves a specific problem. The code you are given is correct and can be successfully executed. However, the code may violate PEP8 conventions in different ways, may contain unnecessary lines, or may be cumbersome. For each piece of code, you are required to:

1) Before your group meeting, try to understand what the code does (do not use AI tools to accomplish this task). What problem is the code trying to solve? Be ready to discuss this with your group.
2) Before your group meeting, try to identify all needed refinements and PEP8 violations (do not use AI tools to accomplish this task). Be ready to discuss this with your group.
3) Working with your group, reach a clear understanding of what the code is designed to do. What problem is the code trying to solve?
4) Working with your group, reach an agreement on all needed refinements, PEP8 violations, and agree on how to fix them, generating a final version of the cleaned up code.
5) Working with your group, thinking back to the strategy and tactics you practiced last week, develop prompts for the AI tool of your choice to:
    a) Understand the main purpose of each piece of code. What problem is the code trying to solve?
    b) Refine the code and clean it up according to PEP8.

6) Working with your group, compare your solutions with the AI tool's solution. Were there significant differences? Did you miss anything? Did the AI tool miss anything? Did you need to write multiple prompts, iterating some of the tactics you learned last week?

DELIVERABLES

1. A video report, up to 15 minutes in length, in which the following items are addressed for each of the three pieces of code. Every group member must appear in the video and address some of the items:
    a. A description of what your group thought the code does.
    b. A description of all refinements and PEP8 adjustments your group thought the code needed. Please be explicit on where in the code the changes were made and why they were made.
    c. A description of what the AI tool thought the code does.
    d. A description of what refinements and PEP8 adjustments the AI tool suggested, possibly in multiple iterations.
    e. A reflection on the whole experience, comparing your group work with the work of the AI tool. Were there major differences? Did anything surprise you?

CODE 1

```
def CheckNumbers( numberList ):
   evenCount=0
   for i in range(0,len(numberList)):
     if numberList[i]%2==0:
       evenCount=evenCount+1
     else:
         pass
   if(evenCount>0):
       print(f"There are {evenCount} even numbers in the list")
   else:
     print("No even numbers found")
   return evenCount
```

CODE 2

```
def f(mas):
    n = len(mas)
    p = n - 1
    while p > 0:
        m = 0
        for i in range(p):
            if mas[i] > mas[m]:
                m = i
        z = mas[m]
        mas[m] = mas[p]
        mas[p] = z
        p = p - 1
    for i in range(n):
        print(mas[i], end=" ")
```

CODE 3

```
def find(text):
    text = text.lower()
    LONGEST = ""

    for i in range(len(text)):
        for j in range(i, len(text)):
            str = text[i:j+1]
            if len(str) > len(LONGEST):
                left = 0
                right = len(str) - 1
                IsPal = True

                while left < right:
                    if str[left] != str[right]:
                        IsPal = False
                        break
                    left += 1
                    right -= 1

                if IsPal:
                    LONGEST = str
    return LONGEST
```