CIS 1100 Optional Review Exercises

Date: September 22nd, 2024
Topic(s): Loops & Sequences

Exercise 0: Review

- Loops (for, while)
- Sequences (strings, lists, tuples)
- List Comprehensions
- Exam prep question

Exercise 0 - For vs While Loop

Write a `while` loop that prints numbers from 1 to 10, but stops if it reaches a number divisible by 5. After doing this write it as a for loop!

```
i = 0
while i < 10 :
    if i % 5 != 0:
        print(i)
    i += 1
```

```
for i in range(10):
    if i % 5 != 0:
        print(i)
```

Exercise 1 - Nested Loops

Make a number pyramid pattern that starts at 1 based on the number of rows the user inputs. Ex: Row = 4

1
1 2
1 2 3
1 2 3 4

```
row = int(input("Rows: "))
for i in range(1, row + 1):
    print()
    for j in range(1, i + 1):
        print(j, end=' ')
```

Exercise 3 - List Comprehension

Rewrite the following block of code in List Comprehension.

```
s = "photosynthesis"
ans = []
vowels = ["a", "e", "i", "o", "u"]
for ch in s:
    if ch in vowels:
        ans.append(ch.upper())
print(ans)
```

Ans:

```
s = "photosynthesis"
ans = []
vowels = ["a", "e", "i", "o", "u"]
ans = [ch.upper() for ch in s if ch in vowels]
print(ans)
```

Exercise 4 - Long Loop Exam Prep

Harry wants to curate the music listening session for the CIS1100 grading party, but he isn't sure what songs to play! He thinks of uniquely identifying numbers for each song and randomly selects a song.

However, he rarely wants to repeat a song, so he is going to track in a sequence whether the song has been played yet. In the case the song has a song id that's divisible by 13, he is ok with repeating with the song. If he randomly generates a song id that has already been played and is not divisible by 13, then he is going from the end of

the playlist (the 1100th song) to the beginning to play the first song not played. Afterwards he goes back to picking random songs.

Harry also wants to print each song played in the following format
Song ID: (song_id) Song Number: (song_number)

Assuming that he will play a total of 50 songs at this party, there are 1100 possible unique songs in this playlist. Write the block of code that meets all the requirements above.

ANS KEY:

```
import random

has_played = [False] * 1100
num_songs_played = 0

for i in range(50):
    random_song_id = random.randint(1, 1100)
    if (not has_played[random_song_id]) or (random_song_id % 13
== 0):
        has_played[random_song_id] = True
        num_songs_played += 1
        print(f"Song ID: {random_song_id} Song Number:
{num_songs_played}")
    else :
        index = 1099
        while has_played[index]:
            index -= 1
        has_played[index] = True
        num_songs_played += 1
        print(f"Song ID: {index} Song Number:
{num_songs_played}")
```