Introduction to Computer Science I - New SME Assignment Solution

Niranjana Srinivasa Ragavan

March 2025

In this document, I have written the explanations for the questions and solutions provided in the given assignment document. I have used PythonTex package to execute the python code written within the Latex. My explanations are based on the assumption of prerequisite knowledge specified in the assignment.

KP1. Understanding Nested List Comprehensions

G1.1 (E) Predicting the Output of a Nested List Comprehension

Question

Consider the following code snippet

```
nested_list = [[(i + j) ** 2 for j in range(1, 4)] for i in range(1, 4)]
print(nested_list)
```

What will be printed?

Solution

```
Output: [[4, 9, 16], [9, 16, 25], [16, 25, 36]]
```

Explanation

Recall that nested list comprehension consists of list comprehensions within a list comprehension. Expression inside `[ ]` is called list comprehension.

- In this code, outer list comprehension `for i in range(1, 4)` which generates values $i = 1, 2, 3$
- For each value of `i`, inner list comprehension `for j in range(1, 4)` generates values $j = 1, 2, 3$
- For each pair of `i` and `j`, the expression `(i + j) ** 2` is computed.
- For each iteration of `i`, the computed result is stored as a sublist and the final result is the list of all these sublists
- **Step by step evaluation**

```
i j (i+j)**2
1 1 4
1 2 9
1 3 16
Sublist for i = 1: [4, 9, 16]
i j (i+j)**2
2 1 9
2 2 16
2 3 25
Sublist for i = 2: [9, 16, 25]
i j (i+j)**2
3 1 16
3 2 25
3 3 36
Sublist for i = 3: [16, 25, 36]
```

G1.2 (M) Converting a `for` Loop to a Nested List Comprehension

Question

Rewrite the following `for` loop as a nested list comprehension:

```
matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
new_matrix = []
for row in matrix:
    new_row = []
    for num in row:
        new_row.append(num + 2)
    new_matrix.append(new_row)
```

Solution

```
new_matrix = [[num + 2 for num in row] for row in matrix]
```

Explanation

KP2. Using Nested List Comprehensions with Strings

G2.1 (E) Predicting Output for Uppercasing Nested Words

Question

What will be printed when the following code is run?

```
sentence = "I am happy"
words = [[char.upper() for char in word] for word in sentence.split()]
print(words)
```

Solution

```
Output: [['I'], ['A', 'M'], ['H', 'A', 'P', 'P', 'Y']]
```

Explanation

KP3. Using Nested List Comprehensions with Conditions

G3.1 (E) Creating a Nested List of Filtered Words

Question

What is the output of the following code snippet?

```
words = [["cat", "elephant"], ["dog", "tiger"], ["fox", "giraffe"]]
long_words = [[word for word in row if len(word) > 3] for row in words]
print(long_words)
```

Solution

```
Output: [['elephant'], ['tiger'], ['giraffe']]
```

Explanation

KP4. Writing Code Using Nested List Comprehensions

G4.1 Writing Code for Creating Given Nested List

Question

Write a Python program using a nested list comprehension to create a 3 x 4 grid filled with zeros and print it.

Solution

```
grid = [[0 for _ in range(4)] for _ in range(3)]
print(grid)
```

Explanation

G4.2 Writing Code for Creating Given Nested List from Strings

Question

Write a nested list comprehension that extracts only vowels from each word in a sentence, storing them in nested lists. For `sentence = "Python Is Amazing"`, the output must be `[['o'], ['I'], ['A', 'a', 'i']]`

Solution

```
sentence = "Python Is Amazing"
vowels = "aeiouAEIOU"
vowel_list = [[char for char in word if char in vowels] for word in sentence.split()]
print(vowel_list)
```

Explanation

G4.3 Writing Code for Creating Given Nested List with Conditionals

Question

Write a nested list comprehension that creates a 5×5 grid, but fills it with 1 if the sum of row and column indices is even, and 0 otherwise, and print it.

Solution

```
grid = [[1 if (i + j) % 2 == 0 else 0 for j in range(5)] for i in range(5)]
print(grid)
```

Explanation