CMPT 120

Lecture 36 – Practice Exam 10

In-Class Activity

Course grading scheme on our course website: **Best 7 in-class** exercises out of 10: 1% each, for a total of 7%

- Our in-class activity #10 -> 1%
 - Write your answer to question _____ on the provided sheet of paper
 - Write your lastname, firstname and student number
 - At the end of today's class, hand in your sheet of paper in the appropriate pile:
 - Pile 1 -> if your lastname start with a letter that is between 'A' and 'L'
 - Pile 1 is on your left-hand side of the classroom
 - Pile 2 -> if your lastname start with a letter that is between 'M' to letter 'Z'
 - Pile 2 is on your right-hand side of the classroom

Try to answer the questions 1st without using your computer, then confirm your answer using IDLE!

Theory and Understanding

How to analyze complexity

- 1. Count the number of times a **critical operation** is executed
 - Usually seen in a loop
 - Express this number as a function of **n** (number of elements) 0-> use Standard Standard Refence functions:
- 2. Disregard constants
- 3. Disregard lower exponent terms (e.g., n when both n² and **n** are present).

Reference Functions

Category	Reference Function
Constant	1
Logarithmic	log ₂ (n)
Linear	n
nlogn	nlog ₂ (n)
Quadratic	n²
Cubic	n ³
Exponential	a ⁿ , a>1

Calculating Time Complexity – Example 1

 Critical operations depending on n? 1 count = 0
2 for i in range(n):
3 count = count + 10
4 for j in range(n):
5 count = count + j

How many additions are executed?

• **n** + **n** = 2**n**

(for loop at lines 2 and 3 repeated **n** times THEN for loop at lines 4 and 5 repeated **n** times)

- So, what is the order of this code fragment (its time complexity/efficiency)?
 - ○(2**n**) -> 2 * ○(**n**) -> ○(**n**)
 - Can discard the factor "2"

Calculating Time Complexity – Question 1 1 x = 0

- Critical operations depending on n?
 - Answer:

- How many addition/assignments are executed?
 - Answer:
- So, what is the order of this code fragment (its time complexity/efficiency)?
 - Answer:

Calculating Time Complexity – Example 2 1 count = 0

 Critical operations depending on n?



- How many addition/assignments are executed?
 - (**n** + **n**) * **n** = 2**n** * **n** = 2**n**²

(for loop at lines 3 and 4 repeated **n** times in 1st j loop THEN for loop at lines 5 and 6 repeated **n** times in 2nd j loop, both loops repeated **n** times in outer for loop)

- So, what is the order of this code fragment (its time complexity/efficiency)?
 - $O(2n^2) \rightarrow 2 * O(n^2) \rightarrow O(n^2)$
 - Can discard the factor "2"

Calculating Time Complexity – Question 2 $1 \mod 1$

2

3

4

- Critical operations depending on n?
 - Answer:

count = 0
for i in range(n):
 for j in range(n):
 count = count + 10

- How many addition/assignments are executed?
 - Answer:
- So, what is the order of this code fragment (its time complexity/efficiency)?
 - Answer:



Question 3 - Matching

Match each statement on the left with the most appropriate word(s) on the right.

- 1. The statement that calls an already executing function.
- 2. A definition which defines something in terms of itself. To be useful it must include base cases which are not recursive.
- 3. A branch of the conditional statement in a recursive function that does not give rise to further recursive calls.
- 4. A function that calls itself recursively without ever reaching the base case.
- 5. The process of calling the function that is already executing.

- a. base case
- b. recursion
- c. recursive call
- d. recursive definition
- e. infinite recursion

Question 4 - Binary

- 1. How many distinct numbers can I represent with ...
 - a) 1 bit? What are these numbers?
 - b) 4 bits? What are these numbers (in binary)?

2. How many distinct numbers can I represent with 7 bits?

These numbers range from _____ to _____

3. How many distinct numbers can I represent with 1 byte?

These numbers range from _____ to ____

4. How many distinct numbers can I represent with 32 bits?

These numbers range from _____ to

Question 5 - Conversion

1. Convert 10011011 into an integer (decimal number):

2. a) What is the binary equivalent of 57?

b) What is the binary equivalent of 157?

Question 6

- In class, we learned a Selection sort algorithm that swapped the smallest number in a list with the first element @ index 0.
 Selection sort can also be implemented by selecting the largest number in the list, and swapping it with the last element @ index "len(list)-1".
- Using this **updated** algorithm, suppose you have the following list of numbers to sort:

[11, 7, 12, 14, 19, 1, 6, 18, 8, 20] Which list below represents the partially sorted list after **three** complete iterations of **Selection sort**?

- a. [7, 11, 12, 1, 6, 14, 8, 18, 19, 20]
- b. [7, 11, 12, 14, 19, 1, 6, 18, 8, 20]
- c. [11, 7, 12, 14, 1, 6, 8, 18, 19, 20]
- d. [11, 7, 12, 14, 8, 1, 6, 18, 19, 20]
- e. None of the above

12

Question 7

- Assume that a problem can be solved with two different algorithms, Algorithm A and Algorithm B, and you need to decide which algorithm to implement based on their time complexity.
- Algorithm A has a time complexity of O(n).
- Algorithm **B** has a time complexity of $O(n \log_2 n)$.
- Which algorithm (Algorithm A or Algorithm B) would you choose, if you had a very large dataset, i.e., if n was very large?
- Answer:

Question 8 – Test Cases

If we wanted to completely test this Python code fragment, how many test cases would we need, i.e., how many different width values must we enter? Note that we are looking at the minimum number of test cases.

What do we mean by "completely"? We mean that each of our test cases will execute a section of the program, such that all of our test cases will execute all statements in our program at least once.

```
width = int(input("Please, enter a width: "))
if width > 0 :
  if width > 10:
    if width \% 2 == 0 :
       # then do something with width
    else :
      print(f"width {width} is not even.")
  else:
    print(f"0 < width {width} <= 10.")
else:
  print("width <= 0.")</pre>
A. 4
B. 2
C. 1
D. 5
E. There are no test cases that could completely test the Python code fragment above.
```

Question 9

- Translate the following message: 01000111 01101111 01101111 01100100 00100000 01101100 01110101 01100011 01101011 00100000 01101001 01101110 00100000 01111001 01101111 01110101 01110010 00100000 01100110 01101001 01101110 01100001 01101100 00100000 01100101 01111000 01100001 01101101 01110011 00100001 0001010
- Answer:

Question 10 – List Comprehension

1. What would this code fragment produce?

[i*i for i in range(10)]

2. What would this code fragment produce? [['*' for j in range(4)] for i in range(4)]

Question 10 (cont'd)

3. Rewrite the code fragment below into a for loop such that both code fragments (the one below and your loop) produce the same result.

[i*i for i in range(10)]

4. Rewrite the code fragment below into for loops such that both code fragments (the one below and your loops) produce the same result.

[['*' for j in range(4)] for i in range(4)]

Coding —

Try to solve the problem (i.e., write your Python program) 1st on a piece of paper without using your computer!

18

Question 1 - Searching

Step 1 - Problem Statement

 Given a list of integers and a target, write a search function that will return a list containing all the indices where the target can be found in the list. If it cannot be found, return an empty list.

Requirements

Your solution must use the append function.

Step 4 – Testing

• You must write at least 3 test cases.

Question 2 - Vowel Counter

Step 1 - Problem Statement

 Write a function called countVowels (aString, vowelCount) that returns the number of vowels in the string aString using recursion.



Question 3 - Conversion

Step 1 - Problem Statement

Write a function that converts a given binary number (entered as a string) into a decimal number and returns it.

Question 4 - Conversion

Step 1 - Problem Statement

Write a function that converts a given decimal number (entered as an integer) into a binary number and returns it.