Written Examination

MED8

Algorithms, Data Structures and Software Engineering for Media Technology

June 2019

Name:_____

Cpr.no.: _____

Study no.: _____

Aalborg University | MSc in Medialogy | Med8

Algorithms, Data Structures and Software Engineering for Media Technology

Ordinary Examination

June 2019

Instructions

- You have 3 hours to complete this examination.
- Neither electronic devices nor written material are allowed in the examination room.
- This examination consists of 10 questions. Each question is worth 10 marks. You must obtain at least 50 marks to pass.
- Do not write any answers on this question paper—answers written on the question paper will be ignored by the examiner. Write all your answers on the writing paper provided.
- Do not write your answers in pencil and do not use a pen with red or green ink. Use a pen with blue or black ink.
- Hand in no more than one answer to each question.
- Do not turn over until you are told to do so by the invigilator.

Question 1

For each of the following equations, state whether it is true or false.

a) $25n^4 + 10n \lg(n^5) + 90 = \Theta(n^4)$

b) $4n^3 \log_2(n^5) = \Theta(n^4)$

c) $\frac{n^3}{2} \log_2(n^4) = O(n^4)$

d) $5^n = O(n^5)$

e) $9n^3 + 4n^2 + 2n + 3 = \Omega(n^3)$

f) $12n^5 + n = o(n^5)$

g) $4n \log_2(n^6) = o(n^2 \log_2 n)$

h) $10n^3 + n^2 + 3n + n \log_2 n + 25 = \omega(n^2 \log_2 n)$

i) $\left(\sqrt{n}\right)^7 \log_2 n = \omega(n^3)$

j) $5n = \Omega\left(\sqrt{n}\right)$

[1 mark for each correct part]

Question 2

The Master Theorem is stated as follows:

Theorem 4.1 (Master theorem)
Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = aT(n/b) + f(n) ,$$

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. If $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$. ■

Given the Master Theorem, as stated above, write down the order of growth in terms of Θ notation for each of the following recurrences.

a) $T(n) = 27T(n/3) + 4n^2$

b) $T(n) = 4T(n/16) + 3\sqrt{n}$

c) $T(n) = 2T(n/8) + \sqrt{n}$

d) $T(n) = 36T(n/6) + 3n^2$

e) $T(n) = 64T(n/4) + \left(\sqrt{n}\right)^5$

[2 marks for each correct part]

Question 3

The following pseudocode describes the partition algorithm, as used in quicksort.

```
PARTITION(A, p, r)
1  x = A[r]
2  i = p − 1
3  for j = p to r − 1
4      if A[j] ≤ x
5          i = i + 1
6          exchange A[i] with A[j]
7  exchange A[i + 1] with A[r]
8  return i + 1
```

Suppose that the array, *A*, initially contains the following values (note that we are using 1-based indexing – the index of each element in the array is shown above it):

1	2	3	4	5	6	7	8	9	10
4	2	8	7	1	3	5	6	4	9

Suppose that the following call to the PARTITION algorithm is executed:

PARTITION(*A*, 2, 9)

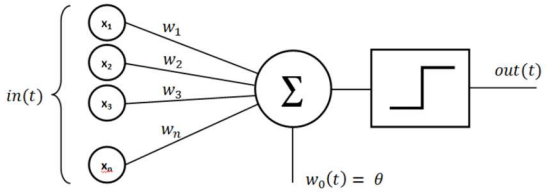
a) Write down the value of each slot in the array *A* after this call to PARTITION has been executed.
b) What value is returned by this call to PARTITION?
c) If we denote the size of *A* by *n*, write down the asymptotic tight upper bound on the running time of PARTITION using appropriate asymptotic order of growth notation (i.e., O, o, Ω, ω or Θ).
d) Write down a loose lower bound on the running time of partition using asymptotic order of growth notation.
e) If the input array *A* is already sorted, what value does PARTITION(*A*, *p*, *r*) return?

[2 marks for each correct part a - e]

Question 4

a) Name an example of a FIFO data structure. [1 mark]
b) Name an example of a LIFO data structure. [1 mark]
c) Suppose we have a stack, S, implemented using an array. The algorithm PUSH(S, x) pushes the element x onto the array S. Suppose the length of the underlying array is stored in *S.length* and that the index of the current top of the stack is stored in *S.top*. Give pseudocode for an implementation of the PUSH algorithm that checks for stack overflow. [3 marks]
d) What is the worst-case asymptotic running time of the PUSH algorithm? [1 mark]
e) Suppose we have a queue, Q, implemented using an array. The algorithm DEQUEUE(Q) removes and returns the element at the head of Q. Give pseudocode for an implementation of the DEQUEUE algorithm that checks for queue underflow. [3 marks].
f) What is the best-case asymptotic running time of the DEQUEUE algorithm? [1 mark]

Question 5



$$y = \sigma\left(b + \sum_{j=1}^{n} w_j x_j\right)$$

Study the figure and equation above that relate to a multilayer perceptron and answer the following questions.

a) State what is represented by each of the following:
   i. $[x_1, x_2,\ldots,x_n]$ [1 mark]
   ii. w_1 [1 mark]
   iii. b [1 mark]

b) Which symbol represents the activation function? [1 mark]

c) What type of activation function is used here? [1 mark]

d) What phenomenon arises that affects training efficiency if your network has too many layers and nodes? [1 mark]

e) What is the main risk of only having fully-connected layers in a multi-layer network? [1 mark]

f) What may be a solution to the problem in part (e)? [1 mark]

g) What kind of machine learning task is each of the following activation functions well suited for:
   i. sigmoid activation function [1 mark]
   ii. linear activation function [1 mark]

Question 6

```
SUMMARY

_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 50)                500050
_________________________________________________________________
dropout_1 (Dropout)          (None, 50)                0
_________________________________________________________________
dense_2 (Dense)              (None, 50)                2550
_________________________________________________________________
dropout_2 (Dropout)          (None, 50)                0
_________________________________________________________________
dense_3 (Dense)              (None, 50)                2550
_________________________________________________________________
dense_4 (Dense)              (None, 1)                 51
=================================================================
Total params: 505,201
Trainable params: 505,201
Non-trainable params: 0
_________________________________________________________________
```

Consider the table above which summarises a deep neural network (DNN) for which the code is shown below. Complete the code of the DNN by replacing the strings WWWWWWW, XXXXXXX, YYYYYYY and ZZZZZZZ (lines 25-28) with appropriate lines of code that add the necessary layers to the DNN. The *dropouts*, in order, are 0.3 and 0.2, the activation function of the hidden layers is relu and the output layer's activation function is sigmoid. [2 marks for each of the four lines of code]

```
1  import numpy as np
2  from keras.utils import to_categorical
3  from keras import models
4  from keras import layers
5  from keras.datasets import imdb
6  (training_data, training_targets), (testing_data, testing_targets) =     imdb.load_data(num_words=10000)
7  data = np.concatenate((training_data, testing_data), axis=0)
8  targets = np.concatenate((training_targets, testing_targets), axis=0)
9
10 def vectorize(sequences, dimension=10000):
11     results = np.zeros((len(sequences), dimension))
12     for i, sequence in enumerate(sequences):
13         results[i, sequence] = 1
14     return results
15
16 data = vectorize(data)
17 targets = np.array(targets).astype("float32")
18 test_x = data[:10000]
19 test_y = targets[:10000]
20 train_x = data[10000:]
21 train_y = targets[10000:]
22 model = models.Sequential()
23 model.add(layers.Dense(50, activation="relu", input_shape=(10000,))) # Input - Layer
24 model.add(layers.Dropout(0.3, noise_shape=None, seed=None))
25 WWWWWWW # Hidden layer
26 XXXXXXX # Dropout
27 YYYYYYY # Hidden layer
28 ZZZZZZZ # Output- Layer
29 model.summary()
30 model.compile(optimizer="adam", loss="binary_crossentropy", metrics=["accuracy"])
31 results = model.fit(train_x, train_y, epochs=2, batch_size=500, validation_data=(test_x, test_y))
32 print("Test-Accuracy:", np.mean(results.history["val_acc"]))
```

Also answer the following questions.

a) What is the purpose of lines 18-21 in the DNN code above? [1 mark]

b) What would the function call `print(len(train_x))` output if `print(len(data))` outputs 50000? [1 mark]

Question 7

a) List the five types of layer that can occur in a CNN. [1/2 mark for each layer type]
b) Illustrate how a residual block works in a neural network. [2½ marks]
c) What is the purpose of a pooling layer in a CNN? [1 mark]
d) The matrices below show the input and a filter that is convolved on the input. Write down the values that will appear in the resulting convolved feature at positions *x* and *y*. [2 marks]

Input

1	0	1	0	1
0	1	1	1	0
1	0	0	1	1
0	1	1	0	0
1	0	0	0	1

Filter

1	0	1
0	1	0
1	0	1

Convolved feature

x		
	y	

e) Given the following input matrix, write down the output matrix that results when a max-pooling filter of size 3 x 3 with stride 2 is applied to it. [2 marks]

1	2	4	1	4	0	1
0	0	1	6	1	5	5
1	4	4	5	1	4	1
4	1	5	1	6	5	0
1	0	6	5	1	1	8
2	3	1	8	5	8	1
0	9	1	2	3	1	4

Question 8

Suppose we have a python file as follows.

```
class MyClass:
    """Some doc"""
    i = 543

    def __init__(self, c="Albert"):
        self.j = 123
        self.c = c

    def f(self):
        return 'hello ' + self.c


class AnotherClass(MyClass):

    def __init__(self, c="Charles"):
        MyClass.__init__(self, c)
        self.d = 3.141

    def f(self):
        return 'hello ' + self.c + ' ' + str(self.d)


x = MyClass()
y = MyClass("Fred")
z = AnotherClass()
```

Assuming this file has just been run, write down what is printed to the console by each of the following print commands. If the command generates an error, give a brief explanation of what causes the error.

a) print(MyClass.i)
b) print(MyClass.__doc__)
c) print(x.j)
d) print(x.i)
e) print(x.f())
f) print(x.c)
g) print(y.c)
h) print(z.f())
i) print(MyClass.j)
j) print(MyClass.f())

[1 mark for each correct part]

Question 9

a) Object composition is generally regarded as preferable to inheritance as a means of reusing functionality. Explain what is meant by *object composition* and why it is usually preferable to inheritance. [4 marks]
b) Briefly explain what is meant by *programming to an interface* and why this is generally a good principle to follow. [2 marks]
c) Briefly explain the *factory method pattern* with the help of a diagram that shows the relationships between the participating classes and interfaces. [4 marks]

Question 10

a) Explain the purpose of the *product backlog* in the scrum framework. [2 marks]
b) Explain what a *sprint* is in the scrum framework. [2 marks]
c) Briefly describe the role of each of the following in a scrum team:
    i. the *product owner* [2 marks]
    ii. the *development team* [2 marks]
    iii. the *scrum master* [2 marks]

END OF EXAMINATION