University of Petra	جامعة البترا	جامعة البترا - ثلاثون عاماً University of Petra 30 Year Anniversary
Faculty of Information Technology		كلية تكنولوجيا المعلومات
Department of Computer Science		قسم علم الحاسوب

Advanced Algorithms
601326
Midterm Exam – 2024 2

Instructions for the Exam:

- Write your name and ID number on the exam and answer sheets.
- Write the number of the section that you enrolled in.
- Write the name of your instructor.
- Questions in the exam not allowed.
- Using any type of technology (mobiles, smart watches) not allowed
- Using extra papers or sheets not allowed
- The exam consists of Six questions.

For instructor use only:

Question number	Course ILO	Program ILO	Question weight	Student mark
Q1			4	0.5
Q2	K2		4	4
Q3	I1		7	6
Q4			5	5
Q5			6	2
Q6			4	3
Total /30				20.5

Q1) Prove by induction that: **(4 marks)**

$\boxed{0,5}$

$$\sum_{i=0}^{n} 2^i = 2^{n+1} - 1$$

- let $n = \overset{0}{\boxed{1}}$

$$\sum_{i=0}^{\boxed{1}} 2^i = 2^2 - 1 \Rightarrow \underset{\checkmark}{1} = \overset{2}{3}\ x$$

?

Q2) Consider the following algorithm. **(4 marks)**

4

```
Algorithm XYZ (A [0..n − 1])
//Input: An array A[0..n − 1] of n real numbers
val ← 5                     — 1
res1 ← 0                    — 1
res2 ← 1                    — 1
for i ← 0 to n − 1 do       — n
    if A[i] <= val          — n-1
    res1 ← A[i]+res1        — n-1
    if A[i] > val           — n-1
    res2 ← A[i]*res2        — n-1
return res1 – res2
```

a. What does this algorithm compute?
b. What is the time complexity for the algorithm?

a) the algorithm return (جمع الاعداد الأقل من 5) - (ضرب الاعداد الاكبر من 5) ✓ ✓

b) 5n+1   O(n)

Not decrease & conquer

Q3) Design a **divide-and-conquer** algorithm to find **Minimum** Number in an unsorted array of N elements **(without using any of the sorting algorithms covered in the class).** **Setup** a recurrence relation for your algorithm. **(7 marks)**

6

A array

```
                      ?
minimum ( A, n, min) {
    if (n = = 0)       ?
        return min
    else if ( A[n] < min)
        min = A[n]
    return minimum ( A, n-1, min)
```

$$T(n) = \begin{cases} 0 & , n = 0 \\ T(n-1) & , n > 0 \end{cases}$$

Q4) Consider the following recursive algorithm: **(5 marks)**

5

```
Algorithm Q(n)
//Input: A positive integer n
if n =1
    return 1
else
    return Q(n − 1) + 2 ∗ n − 1
```

a) **Setup** a recurrence relation for the number of multiplications made by this algorithm.

b) **Solve** the recurrence relation using backward substitution and find Big O.

a) $T(n) = \begin{cases} 0 & , n = 1 \\ T(n-1) + 1 & , n > 1 \end{cases}$

b) $T(n) = T(n-1) + 1$

$T(n-1) = T(n-2) + 1$

$T(n-2) = T(n-3) + 1$

$T(n-3) = T(n-4) + 1$

$T(n) = T(n-4) + 4$

$T(n) = T(n-i) + i$

$n - i = 1 \Rightarrow i = n - 1$

$T(n) = T(n-n+1) + n - 1$

$T(n) = n - 1$

$O(n)$

2

Q5) Given the following array, A= -3 ,8, 4, 22, 13, 6, 20, 9 **(6 marks)**

a) Apply Quick Sort algorithm to sort the array elements in ascending order **(show detailed steps)**

b) Compare Quick sort to Insertion Sort in terms of best and worst cases.

a) ~~using median of 3 for the pivot~~

```
S ← partition(A[L....r])

pivot
 = -3        ~~pivot~~
i = 1
j = 9
repeat
    repeat i=i+1 until A[i] > pivot
    repeat j=j-1 until A[j] < pivot
    swap A[i], A[j]
until i≥j
swap A[i], A[i]
swap pivot, A[j]
return j
```

0

-3 8 4 22 13 6 20 9

8 4 22 13 6 20 9

But you should the Quick Sort algo. for sorting the array NOT to write the algo. itself!!

2

b) - using the above quick sort will be in the worst case because the pivot always sorting at the left (n-1) every time — time complexity will be $O(n^2)$

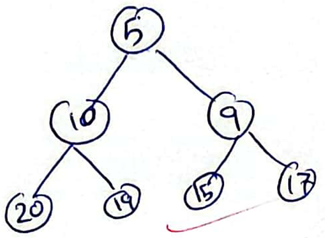
if the pivot was in the middle we will get a best case $O(n\log n)$ - only if it divide the algorithm into two equal parts.

- insertion sort best case when its already sorted $O(n)$

worst case when its reversed sorted $O(n^2)$

Q6) Show that the worst case running time of MAX-HEAPIFY on a heap of size *n* is ***log n*** (Hint: For a heap with n nodes, give node values that cause MAX-HEAPIFY to be called recursively at every node on a simple path from the root down to a leaf.) **(4 marks)**

3



?!

the worst case will accure when its reversed sorted

building max heap will call heapify everytime

?