Answer all questions (4 Qs)

Question one:
1- Complete the following:
   A. If \( T_1(n) = O(f(n)) \) \( \& \) \( T_2(n) = O(f(n)) \) then
      1- \( T_1(n) + T_2(n) = \) ________________
      2- \( T_1(n) * T_2(n) = \) ________________
   B. A function is said to be in \( O(f(n)) \), denoted \( T(n) = O(f(n)) \) \( \& \) \( n > 0 \) if there exist \( n_0 \) \( \& \) \( c \) positive constants ,such that……

C. A function is said to be in \( \Omega (f(n)) \), denoted \( T(n) = \Omega (f(n)) \) \( \& \) \( n > 0 \) if there exist \( n_0 \) and \( c \), positive constants ,such that……

D. A function is said to be in \( \Theta (f(n)) \), denoted \( T(n) = \Theta (f(n)) \) \( \& \) \( n > 0 \) if there exist \( n_0 \) and \( c_1, c_2 \) positive constants ,such that……

2- Put a circle round the correct answer:
   1- binary search algorithm is application of:
      2- Divide and conquer technique
      3- Greedy technique
      4- All are wrong
   2- Efficiency class of multiplication of two dimension matrices is:
      2- \( O(n^3) \)
      3- \( O(n^2) \)
      4- \( O(n \log n) \)
   3- the rate of growth for an algorithm is \( T(n) = \sum_{i=0}^{n} d \), it’s efficiency class
      2- \( O(n) \)
      3- \( O(n^2) \)
      4- \( O(n \log n) \)
4- the rate of growth for an algorithm \( T(n) = \sum_{j=1}^{n} \sum_{i=1}^{n} d \) it’s efficiency class

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- O((n^3))</td>
<td>1- O(n)</td>
</tr>
<tr>
<td>4- All are wrong</td>
<td>3- O((n^4))</td>
</tr>
</tbody>
</table>

5- The worst-case for the algorithm of Searching an element in array of size \(n^2\), is

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- One comparison Operation</td>
<td>1- (n/2) Operations</td>
</tr>
<tr>
<td>4- (\frac{n(n-1)}{2}) Operations</td>
<td>3- (N^2) Operations</td>
</tr>
</tbody>
</table>

6- A sequential search algorithm is application of

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- Divide and conquer technique</td>
<td>1- Decrease and conquer</td>
</tr>
<tr>
<td>4- All are wrong</td>
<td>3- Greedy technique</td>
</tr>
</tbody>
</table>

7- for The following algorithm

```
for i = 1 to n do
  Begin
    j = i div 2
  end
end for
```

The efficiency class is:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- O((n^3))</td>
<td>1- O(n)</td>
</tr>
<tr>
<td>4- O((n \log n))</td>
<td>3- O((n^2))</td>
</tr>
</tbody>
</table>

8- the growth rate of the algorithm defined the relationship

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

The efficiency class is:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- O((n^3))</td>
<td>1- O(n)</td>
</tr>
<tr>
<td>4- O((n \log n))</td>
<td>3- O((n^2))</td>
</tr>
</tbody>
</table>

9- design method characterized by brute force it:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- smart</td>
<td>1- easy to use</td>
</tr>
<tr>
<td>4- Answers 1 and 3</td>
<td>3- many uses</td>
</tr>
</tbody>
</table>

10 - A selection sort algorithm is application of

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2- Divide and conquer technique</td>
<td>1- Decrease and conquer</td>
</tr>
<tr>
<td>4- All are wrong</td>
<td>3- Greedy technique</td>
</tr>
</tbody>
</table>
1- Prove the following :
   a- \( \frac{1}{2}n(n+1) = \Omega(n^3) \)
   b- \( \frac{1}{2}n(n+1) = \Theta(n^3) \)

2- Find a solution the sequence:
   \[ \sum_{i=1}^{N} (3i + 7) \]

3- Arranged the following growth rates upward ?.

<table>
<thead>
<tr>
<th>n \log n</th>
<th>n+n^2+n^3</th>
<th>n^4+\log n</th>
<th>2^n</th>
<th>n^2 \sqrt{n}</th>
</tr>
</thead>
<tbody>
<tr>
<td>log n</td>
<td>n^3</td>
<td>n</td>
<td>n!</td>
<td>(1/2)^n</td>
</tr>
</tbody>
</table>

**QUESTION 3:-**

1. Write any algorithm to find a common divider of two numbers ?
2. Put in summation notation the following loops :
   
   For j = 1 to n do
   Print (j)
   Then find efficiency class \( O(f(n)) \) of it ?
3. Using the Euclidean algorithm Find gcd (138 , 75) with the writing of all steps
4. A growth rate of a iterative algorithm is \( T(n) = \sum c \), i=1 to n , and c is constant , find it’s efficiency class \( O(f(n)) \).

**QUESTION 4 :**

1- Consider the following algorithm :
   Algorithm test1(n)
   // input : n integer ,
   S=0
   For i=1 to n do
   S = S + i*i*i
   Return s
   a- What does this algorithm compute ?
   b- What is its basic operation ?
   c- How many times is the basic operation executed?
   d- efficiency class of this algorithm ?

-着力打造算法表自变量成绩O(f(x))
  للها :- 2n×n
  For i= 1 to n do
  For j= 1 to n do
  \( C_{ij} = A_{ik} + B_{kj} \)