Week 8 PYQs

15 June 2024 16:16

Counting Inversions:

1. Quiz 2, Jan 24

Question Label : Short Answer Question

Let `L` be an integer list of length `n`. The number of **inversions** is the number of the different pairs `(i, j)` where:

- `0 <= i < j < n`
- `L[i] > L[j]`

The total number of **inversions** for `L = [1, 3, 5, 7, 9, 8, 6, 4, 2]` is ___.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

[redacted]

Text Areas : PlainText

Possible Answers :

[redacted]

2. End Term, Jan 24

Question Label : Short Answer Question

In a list `L`, two elements `L[i]` and `L[j]` form a inversion if `L[i] > L[j]` and `i < j`. The total number of inversions for the list `L = [3, 4, 8, 9, 7, 5, 1]` is __.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

[redacted]

3. End Term, Sep 23

Question Label : Multiple Choice Question

In a list `L`, two elements `L[i]` and `L[j]` form a **significant inversion** if `L[i] > 2 * L[j]` and `i < j`. The total number of significant inversions for **L = [1, 11, 6, 3, 5, 2]** is___.

Options :

a. 4
b. 5
c. 6
d. 7

4. Quiz 2, Sep 23

Question Label : Short Answer Question

In a list L, two elements $L[i]$ and $L[j]$ form an **inversion** if $L[i] > L[j]$ and $i < j$.

What is the number of inversion pairs for the list $L = [1, 5, 4, 2, 6, 3]$?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

5. End Term, Sep 23

Question Label : Multiple Choice Question

Let's take a list of integers **L = [2, 20, 6, 3, 5]**. Which of the below are the all inversion pairs of the list **L**?

Options :

6406532324456. ✖ (20, 2), (20, 6), (20, 3), (20, 5), (6, 3)

6406532324457. ✔ (20, 6), (20, 3), (20, 5), (6, 3), (6, 5)

6406532324458. ✖ (20, 6), (20, 3), (20, 5), (6, 3), (3, 5)

6406532324459.

✖ (20, 2), (20, 3), (20, 5), (20, 6)

6. Quiz 2, May 23

Question Label : Short Answer Question

In a list `L`, two elements `L[i]` and `L[j]` form an inversion if `L[i] > L[j]` and `i < j`. Consider a list `L` of length `n` in which all elements are distinct. List `L` has exactly 21 inversions. The minimum possible value of `n` is ___.

Answer:

Closest Pair of Points:

1. Quiz 2, Jan 24

> Question Label : Multiple Choice Question
>
> Apply the divide and conquer strategy to find the **closest pair of points** in a set. After dividing the set into two halves and recursively finding the closest pairs in each half, what additional step is required?
>
> **Options :**
>
> 6406532578263. ✖ Combine the results directly
>
> 6406532578264. ✖ Perform a linear search for the closest pair
>
> 6406532578265. ✔ Consider pairs that span both halves
>
> 6406532578266. ✖ Sort the points by their distances

Recursion Trees:

1. Quiz 2, Jan 24

> Consider the following recurrence relation for an algorithm:-
>
> $T(n) = 4T(n/2) + O(n)$
>
> Base Case:- $T(1) = 1$
>
> The complexity of this algorithm is__ .
>
> **Options :**
>
> [redacted] $O(n)$
>
> [redacted] $O(\log n)$
>
> [redacted] $O(n^2)$
>
> [redacted] $O(n \log n)$

2. Quiz 2, Sep 23

Question Label : Multiple Choice Question

Consider the following recurrences.

1. $T_1(n) = 3T_1(n/3) + O(n^2)$
2. $T_2(n) = 9T_2(n/3) + O(n)$

Base Case:- $T_1(1) = T_2(1) = 1$

Select the correct complexity for given recurrences.

Options :

[redacted] $T_1 = O(n^2)$ and $T_2 = O(n^2)$

[redacted] $T_1 = O(n\ log\ n)$ and $T_2 = O(n^2)$

[redacted] $T_1 = O(n\ log\ n)$ and $T_2 = O(n^2\ log\ n)$

[redacted] $T_1 = O(n^2)$ and $T_2 = O(n\ log\ n)$

Screen clipping taken: 25-06-2024 14:53

3. Quiz 2, Sep 23

Question Label : Multiple Choice Question

Consider the following function to return the minimum element in the list `L` of size `n`.

```
def find_min(L, low, high):
    if low == high:
        return L[low]
    mid = (low + high) // 2
    min1 = find_min(L, low, mid)
    min2 = find_min(L, mid + 1, high)
    return min(min1, min2)
```

Which of the following represents the correct recurrence relation for the given function `find_min`?

Options :

[redacted] $T(n) = T(n/2) + n$

[redacted] $T(n) = 2T(n/2) + 1$

[redacted] $T(n) = 2T(n/2) + n$

[redacted] $T(n) = T(n/2) + 1$

Screen clipping taken: 25-07-2024 09:11

4. End Term, May 23

Consider the following recurrence relation for an algorithm:-

$T(n) = 2T(n/3) + O(n^2)$

Base Case:- $T(1) = 1$

The complexity of this algorithm is__ .

Options :

[redacted] $O(n)$

[redacted] $O(\log^2 n)$

[redacted] $O(n^2)$

[redacted] $O(n^3)$

[redacted]

[redacted] $O(n \log n)$

Screen clipping taken: 14-06-2024 13:18

5. Quiz 2, May 23

Maximum subarray sum:- Given an array of integers, the goal is to find a contiguous subarray (i.e., a subarray with elements positioned adjacent to each other in the original array) that has the largest possible sum.

Consider the following implementation `max_subarray_sum` to find the maximum subarray sum in an array:

```python
def max_crossing_sum(arr, low, mid, high):
    left_sum = float('-inf')
    curr_sum = 0
    for i in range(mid-1, low - 1, -1):
        curr_sum += arr[i]
        if curr_sum > left_sum:
            left_sum = curr_sum

    right_sum = float('-inf')
    curr_sum = 0
    for i in range(mid, high):
        curr_sum += arr[i]
        if curr_sum > right_sum:
            right_sum = curr_sum
    return left_sum + right_sum

# In First call low = 0, high = len(arr)
def max_subarray_sum(arr, low, high):
    if high - low <= 1:
        return arr[low]

    mid = (low + high) // 2

    left_sum = max_subarray_sum(arr, low, mid)
    right_sum = max_subarray_sum(arr, mid, high)
    cross_sum = max_crossing_sum(arr, low, mid, high)

    return max(left_sum, right_sum, cross_sum)
```

What is the worst-case time complexity of this algorithm when applied to an array of size n?

Screen clipping taken: 25-07-2024 09:44

Quick / Fast Select:

1. Quiz 2, Jan 24

Question Label : Multiple Select Question

Consider the following statements and choose the correct ones.

Options :

[redacted] The worst case running time of Quick select algorithm to find the kth largest number is $O(n)$

[redacted] The time taken to find the median in an unsorted list using the Median of Medians(MoM) algorithm is $O(n)$

[redacted] The Quick select algorithm is an example of the divide-and-conquer approach.

[redacted] Using the Fast Select (Quick Select using MoM for pivot selection) strategy, the worst-case running time will be $O(n^2)$ to find the kth largest number.

Screen clipping taken: 25-06-2024 14:39

2. End Term, Jan 24

Question Label : Short Answer Question

Consider the following function `MoM`.

```
def MoM(L): # Median of medians
  if len(L) <= 5:
    L.sort()
    return(L[len(L)//2])
  # Construct list of block medians
  M = []
  for i in range(0,len(L),5):
    X = L[i:i+5]
    X.sort()
    M.append(X[len(X)//2])
  return(MoM(M))
```

What median value will be returned by the given `MoM` function for the following list?

```
[6,7,8,10,11,10,15,13,14,17,2,3,4,3,5]
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

Screen clipping taken: 25-06-2024 14:46

3. Quiz 2, Sep 23

Consider the following statements and choose the correct ones.

Options :

[redacted] The worst case running time of the Quick select algorithm to find the k-th largest number is $O(n^2)$

[redacted] The time taken to find the median in an unsorted list using the Median of Medians (MoM) algorithm is $O(n)$

[redacted] Quick select algorithm is an example of the divide-and-conquer approach to solving problems

[redacted] Using Fast Select (Quick Select using MoM for pivot selection) strategy, the worst-case running time will be $O(n \log n)$.

Screen clipping taken: 25-06-2024 14:55

4. End Term, Sep 23

Consider the following implementation for Median of Medians(MoM).

```
def MoM(L): # Median of medians
  if len(L) <= 5:
    L.sort()
    return(L[len(L)//2])
  # Construct list of block medians
  M = []
  for i in range(0,len(L),5):
    X = L[i:i+5]
    X.sort()
    M.append(X[len(X)//2])
  return(MoM(M))
```

Let **L = [8, 9, 5, 4, 1, 3, 6, 11,10, 19, 16, 6, 19, 18, 7, 20, 5, 76, 32, 2]**. What is the returned value of **MoM(L)** using the list **L**?

Options :

6406532324452. ✖ 18

6406532324453. ✖ 19

6406532324454. ✔ 16

6406532324455. ✖ 20

Screen clipping taken: 20-07-2024 14:05