Coachable

You can share this report by copying the URL above

Save PDF

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023
Completed in 2 hours and 43 minutes

- SCORES
- SKILL RATINGS
- SCORECARD
- CHALLENGE SOLUTIONS
- MULTIPLE CHOICE ANSWERS

Scores

Scores	Average	Qualifying	Final
	72%	70%	97%
Coding challenges × 50%			100%
Multiple choice × 50%			94%
Open-ended			N/A

Cheating	Overall No issues
Pasted code	Not detected
Plagiarism	Not detected
Leaving tab	5 times
AI detection	Not detected

Skill ratings

Learn more about these

Python3	Data Structures	Algorithms
Advanced	Beginner	Beginner

Scorecard

Algorithm skills	Code quality	Python3 skills	Recommenda
☆☆☆☆☆	☆☆☆☆☆	☆☆☆☆☆	

Private notes to share with your team...

Challenge solutions

Expa

DSA Final Anagram and Close Strings

Run solution | Unit tests

Cheating not detected

10/10

Python3
Video
United States

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023
Completed in 2 hours and 43 minutes

SCORES
SKILL RATINGS
SCORECARD
CHALLENGE SOLUTIONS
MULTIPLE CHOICE ANSWERS

DSA Final Queue Run solution | Unit tests	Cheating not detected	10/10 Python3 Video United States
DSA Final Max Tree Height Diff Run solution | Unit tests	Cheating not detected	10/10 Python3 Video United States

Multiple choice answers

1. Suppose we are sorting an list of eight integers using quicksort, and we have just finished the first partitioning with list looking like this:
[2, 5, 1, 7, 9, 12, 11, 10]

Which statement is correct?

✔ The pivot could be either the 7 or the 9

2. Which of the following algorithm design techniques is used in merge sort?

✔ Divide and Conquer

3. What is the output of the following code block?

```
my_string = 'coachable'
for i in range(len(my_string)):
    print(i)
```

✔ 0 1 2 3 ... 8

4. What is the output of the following code block?

```
my_string = 'coachable'
for i in range(len(my_string)):
    my_string[i].upper()
print(my_string)
```

✔ coachable

5. What is the runtime of the following code block, where n is the length of the string_to_construct?

```
string_so_far = ''
string_to_construct = 'coachable'
for i in range(len(string_to_construct)):
  string_so_far += string_to_construct[i]
```

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

- SCORES
- SKILL RATINGS
- SCORECARD
- CHALLENGE SOLUTIONS
- MULTIPLE CHOICE ANSWERS

7. What is the runtime complexity of the following function?

```
def function(n: int) -> int:
  if n == 0 or n == 1:
    return 1
  return function(n-1) + function(n-1)
```

✔ O(2^n)

8. What is the worst-case space complexity of the following function, where n is the size of the input tree?

```
class TreeNode:
  def __init__(self, val, left=None, right=None):
    self.val = val
    self.left = left
    self.right = right

def function(root: TreeNode) -> None:
  if not root:
    return
  function(root.left)
  print(root.val)
  function(root.right)
```

✔ O(n)

9. Which of the following inputs will not work with any implementation of binary search?

✔ [2, 3, 4, 1, 2]

10. Which of the following is the best hash function?

✔ For a number input, return the number mod M

11. Which of the following options is a potential insertion sequence resulting in the linear probing hash table below?

```
A linear probing hash table of length 10 uses the hash function h(x) = x mod 10.
After inserting six integer keys into an initially empty hash table, the array of keys is:

Index 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
Value   |   | 42| 23| 34| 52| 46| 33|   |
```

✔ 46, 34, 42, 23, 52, 33

12. What is the runtime to insert an element to the front of a linked list?

✔ O(1)

13. What is the worst-case runtime to check for the existence of a particular value in a linked list?

✔ O(n)

14. Assume we have a Stack class with the typical operations. What does the stack look like after the following sequen operations?

For all answers, the way to interpret it is the left-hand side is the bottom of the stack and the right-hand side is the to the stack.

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

- SCORES
- SKILL RATINGS
- SCORECARD
- CHALLENGE SOLUTIONS
- MULTIPLE CHOICE ANSWERS

```
stack.pop()
stack.push(10)
stack.push(12)
```

✔ [3, 10, 12]

15. Assume we have a Queue class with the typical operations. What does the queue look like after the following seque of operations?
For all answers, the way to interpret it is the left-hand side is front of the queue and the right-hand side is the back of queue.

```
queue = Queue()
queue.enqueue(3)
queue.enqueue(2)
queue.enqueue(1)
queue.enqueue(0)
queue.dequeue()
queue.dequeue()
queue.enqueue(4)
queue.enqueue(5)
queue.dequeue()
queue.enqueue(10)
queue.enqueue(6)
```

✔ [0, 4, 5, 10, 6]

16. If I wanted to find the K smallest elements in a stream of length N, would I use a min or a max heap of size K? What the runtime be?

✔ Max heap, O(N log K)

17. What is the runtime of pushing an element into a heap of size N? What about popping an element out of the heap?

✔ Push O(log N), Pop O(log N)

18. Suppose I wanted to get the sizes of all subtrees in a tree. For instance, for the tree

```
         A
       /   \
      B     C
    /     /   \
   D     E     F
```

I'd want to output something like {A: 6, B: 2, D: 1, C: 3, E: 1, F: 1}.

What traversal would I use?

✔ Postorder

19. Suppose I wanted to print the node values in a tree level-by-level. For instance, for the tree

```
         A
       /   \
      B     C
    /     /   \
   D     E     F
```

I'd want to output something like [A, B, C, D, E, F]

What traversal would I use?

✔ Breadth-First Search

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

SCORES

SKILL RATINGS

SCORECARD

CHALLENGE SOLUTIONS

MULTIPLE CHOICE ANSWERS

```
graph = {
  1: [2, 3],
  2: [4, 5],
  3: [5, 6],
  4: [7],
  5: [7],
  6: [7],
  7: []
}
```

✔ 1 3 2 5 6 4 7

22. What is the main difference between BFS and DFS?

✔ BFS visits all nodes at the same depth before visiting nodes at deeper depths, while DFS visits nodes in a depth-f manner.

23. What is the runtime and space complexity of the following function?

```
cache = {0: 1, 1: 1}
def f(n: int) -> int:
  if n in cache:
    return cache[n]
  cache[n] = function(n-1) + function(n-2)
  return cache[n]
```

✔ Time O(n), Space O(n)

24. What is the runtime and space complexity of the following code?

```
def f(n: int) -> int:
  a, b = 1, 1
  for i in range(2, n):
    c = a + b
    a, b = b, c
  return a + b
```

✔ Time O(n), Space O(1)

25. Given the following problem:

> "Find the minimum number of moves to tile a 2 x N board with 2 x 1 dominos such that no two dominos overlap."

What is the recurrence relationship for the dynamic programming solution to this problem?

✔ dp[i] = dp[i - 1] + dp[i - 2]

26. What is the runtime of the following code blocks in terms of N?

```
def fn_a(N: int):
  sum = 0
  n = N
  while n > 0:
    for i in range(0, n):
      sum += 1
    n = n // 2
  return sum
```

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

SCORES

SKILL RATINGS

SCORECARD

CHALLENGE SOLUTIONS

MULTIPLE CHOICE ANSWERS

✔ O(N), O(N log N)

27. Which of the following describes the recurrence relationship for the runtime of the mergesort algorithm?

✔ T(n) = n + T(n/2) + T(n/2)

28. Given the following list of tuples

```
arr [(3, "A"), (1, "B"), (2, "C"), (3, "D"), (1, "E"), (2, "F")]
quicksort(arr, key = lambda x: x[0])
```

The `key = lambda x: x[0]` means that we are sorting with quicksort only using the first element of the tuple in the comparison i.e. ties are not broken using the 2nd element.

What will `arr` be after applying quicksort this way?

× [(1, "B"), (1, "E"), (2, "C"), (2, "F"), (3, "A"), (3, "D")]

29. Which of the following best describes the function f?

```
def g(node: Node, d: int):
    if node is None:
        return
    if d % 2 == 0:
        g(node.left, d + 2)
        print(node.val)
        g(node.right, d + 2)
    else:
        g(node.right, d + 1)
        print(node.val)
        g(node.left, d + 1)

def f(root: Node):
   g(root, 0)
```

✔ It performs an in-order traversal of a binary tree and prints the values of the nodes.

30. There is a robot on an m x n grid. The robot is initially located at the top-left corner (i.e., grid[0][0]). The robot tries
move to the bottom-right corner (i.e., grid[m - 1][n - 1]). The robot can only move either down or right at any point in ti
There are also obstacles meaning if `grid[i][j] == 0` then the robot cannot take a path through that cell. If `grid[I][j] ==
then the robot can freely pass through that cell.

What recurrence relation do we use to compute the number of unique paths the robot can take to reach the bottom ri
corner?

× dp[i][j] = dp[i - 1][j] + dp[i][j - 1] if grid[i][j] != 0 else dp[i][j] = 0

31. The below code computes the binomial coefficient `C(n,k) = (n!)/ ((n-k)!(k!))`

```
def binomial_coefficient(n, k):
    result_n = 1
    for i in range(1, n + 1):
        result_n *= i

    result_k = 1
    for i in range(1, k + 1):
        result_k *= i

    result_n_minus_k = 1
```

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

✔ The code recomputes the factorial each time. We should refactor this logic so this computation is only done once

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

- SCORES
- SKILL RATINGS
- SCORECARD
- CHALLENGE SOLUTIONS
- MULTIPLE CHOICE ANSWERS

32. What does the following code block output?

```
x = 1
def f():
  x = 2
f()
print(x)
```

✔ 1

33. Suppose we start with an empty max-heap and insert the following elements in the order given:

20, 10, 15, 25, 30, 22, 35.

What is the list representation of the resulting max-heap?

✔ [35, 25, 30, 10, 20, 15, 22]

34. Compute the DP matrix for the following source matrix.

```
matrix = [[2, 4, 3],
          [5, 2, 1],
          [6, 3, 2]]

dp = [[float(-inf) for _ in range(len(matrix[0]))] for _ in range(len(matrix))]

for r in range(len(matrix)):
  for c in range(len(matrix[0])):
    dp[r][c] = max(dp[r][c], matrix[r][c])
    if c > 0:
       dp[r][c] = max(dp[r][c], dp[r][c-1])
    if r > 0:
       dp[r][c] = max(dp[r][c], dp[r-1][c])

print(dp)
```

×

```
2 4 3 5 5 5 6 6 6 [[, , ], [, , ], [, , ]]
```

35. Which traversal is this?

```
class TreeNode:
  def __init__(self, val, left=None, right=None):
    self.val = val
    self.left = left
    self.right = right

def f(node: TreeNode) -> None:
  if node is None:
    return
  print(node.val)
  f(node.right)
  f(node.left)
```

✔ Preorder

36. Which traversal is this?

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

- SCORES
- SKILL RATINGS
- SCORECARD
- CHALLENGE SOLUTIONS
- MULTIPLE CHOICE ANSWERS

```
def f(node: TreeNode) -> None:
  a = deque()
  a.append(node)
  while a:
    node = a.popleft()
    print(node.val)
    if node.left:
      a.append(node.left)
    if node.right:
      a.append(node.right)
```

✔ Breadth First Search

37. What does this code do? Assume that `root` is an N-ary Tree Node

```
class NTreeNode:
    def __init__(self, val, children):
        self.val = val
        self.children = children

def f(root:NTreeNode):
    if root is None:
        return 0
    x = 0
    for child in root.children:
        x = max(x, f(child))
    return x + 1
```

✔ Finds the maximum depth of an N-ary tree

38. Assume that `graph` is a 2D boolean array.

```
def foo(graph):
    visited = [[False for j in range(len(graph))] for i in range(len(graph))]
    count = 0

    def dfs(i, j):
        if i < 0 or i >= len(graph) or j < 0 or j >= len(graph) or not graph[i][j] or visited[i][j]:
            return

        visited[i][j] = True
        dfs(i + 1, j)
        dfs(i - 1, j)
        dfs(i, j + 1)
        dfs(i, j - 1)

        for i in range(len(graph)):
            for j in range(len(graph)):
                if graph[i][j] and not visited[i][j]:
                    count += 1
                    dfs(i, j)

    return count
```

Which of the following best describes `foo`?

✔ Counts the connected components in graph

39. Consider the following problem: Given a binary tree, find the number of paths from the root to the leaves that sum given value.

Which of the following is the correct recursive solution for this problem?

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023

Completed in 2 hours and 43 minutes

SCORES

SKILL RATINGS

SCORECARD

CHALLENGE SOLUTIONS

MULTIPLE CHOICE ANSWERS

41. How do I find out if a linked list has a cycle in the most optimal space and time complexity?

✔ Maintain a fast and slow pointer. Advance the fast pointer 2 nodes at a time and the slow pointer 1 node at a time. they ever are equal, the linked list has a cycle.

42. Fullness is defined as the number of nodes in the tree divided by the the maximum possible number of nodes for a of that height. Specifically, for a tree at height h, the fullness is given by

size(tree) / (2^h - 1)

```
        A
      /   \
     B     C
   /     /
  D     E
```

What is the "fullness" of each subtree (including the entire tree itself)?

Note: The above tree has a height of 3.

✔ A: 5/7, B: 2/3, C: 2/3, D: 1/1, E: 1/1

43. What's the missing line of code for the solution to the following problem?

Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.
An input string is valid if:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.
3. Every close bracket has a corresponding open bracket of the same type.

```
def is_valid(s: str) -> bool:
  # TODO: finish this line!
  mapping = ________________________________
  openings = []
  for c in s:
    if c not in mapping:
      openings.append(c)
    else:
      if openings and mapping[c] == openings.pop():
        continue
      return False
  return not openings
```

✔ { ')': '(', '}': '{', ']': '['}

44. What are the two ways I can check if two strings are anagrams of one another?

✔ Sort the strings and check equality; Count the characters in each string and check equality of the counts

45.

```
Given an list of integers nums sorted in non-decreasing order,
find the starting and ending position of a given target value.

If target is not found in the list, return [-1, -1].

Example 1:
Input: nums = [5,7,7,8,8,10], target = 8
Output: [3,4]
```

Coachable

You can share this report by copying the URL above

Assessment Report

Week 8: DS&A Final

Shiva Reddy

shivajreddy@outlook.com

Joined on June 26, 2023
Completed in 2 hours and 43 minutes

SCORES
SKILL RATINGS
SCORECARD
CHALLENGE SOLUTIONS
MULTIPLE CHOICE ANSWERS

Given this problem description above, can I fundamentally solve it using binary search? Why or why not?

✔ Yes, but you'll need to run binary search twice. Once to find the left bound and one to find the right bound.

46. A rotated sorted array is an array that has been sorted in ascending order and then rotated some number of times the right. This means that the elements in the array are still sorted, but the starting point of the sorted order has chan

```
arr = [3, 4, 5, 1, 2]
```

For example, `arr` is a rotated sorted array since 5 is a pivot where after and before it is sorted.

Suppose you were given a rotated sorted array. Can you still search for elements in O(log n) time just like a regular bin search?

✔ Yes. We can find the pivot point and then performing binary search on the left and right subarrays.

47. Which of the following recurrences represents f(n) = 2^n ?

✔ f(n+1) = 2*f(n)

48. Suppose you have an initially empty hash map with a fixed size of 10 and you use linear probing to resolve collisior hash function is simply the modulo operation with the size of the hash map. You insert the following elements in the c given:

12, 44, 13, 88, 23, 94, 11, 39, 20, 16.

What is the underlying array representation of the hash map after all the elements have been inserted?

✔ [20, 11, 12, 13, 44, 23, 94, 16, 88, 39]

49. What could the heap-as-a-list look like for the below heap-as-a-tree?

```
        2
     /     \
    3       6
  /   \    /
 8     10  15
```

✔ [2, 3, 6, 8, 10, 15]

Help Center Blog Privacy Terms Contact Changelog