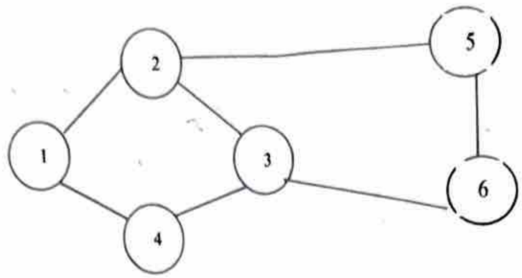
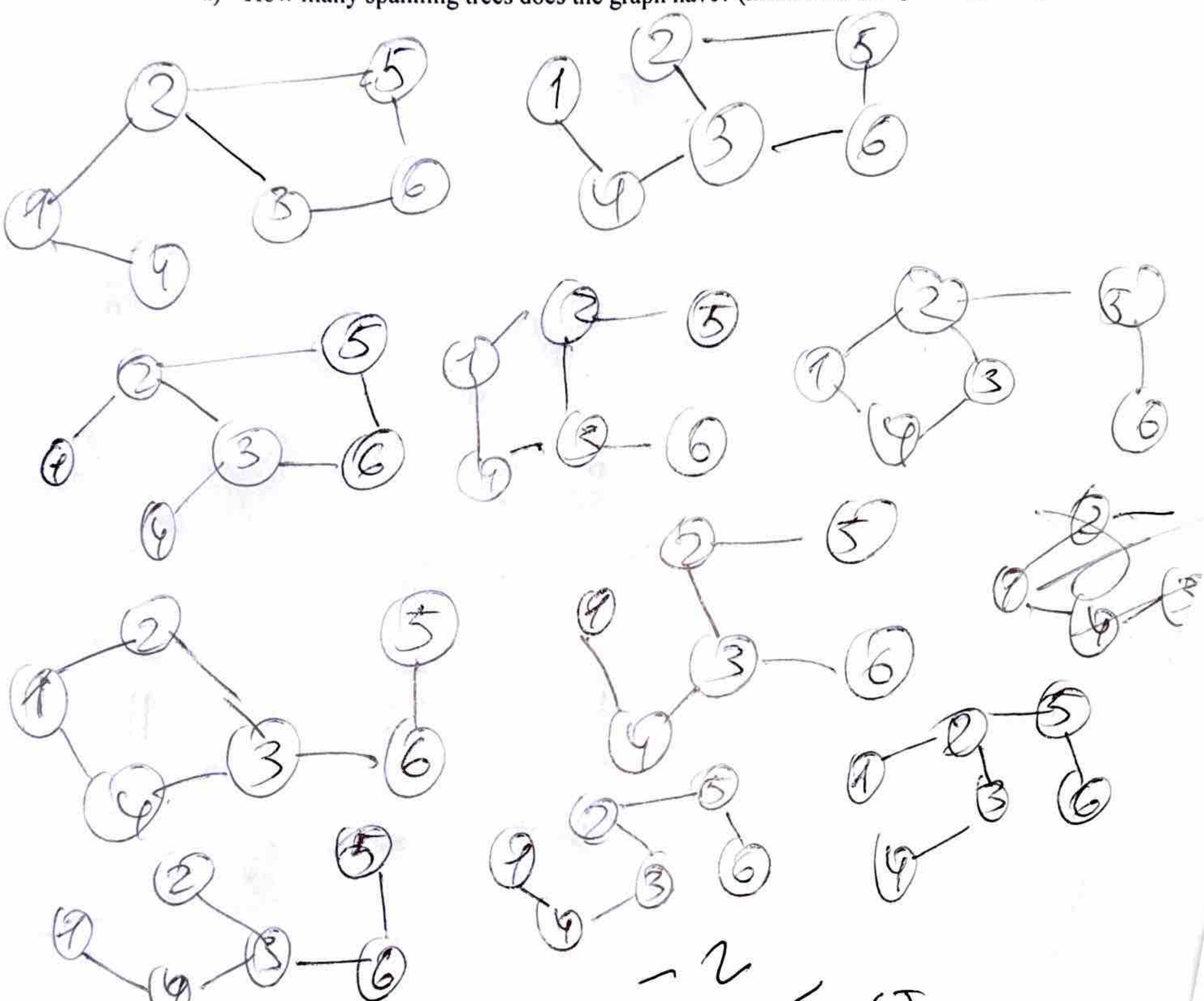
Questions

1) Consider the following graph



a) How many spanning trees does the graph have? (must draw all spanning trees)



there are 15 STs

CSC382 --- Exam 2 V1

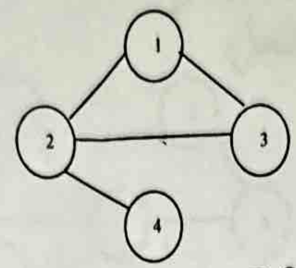
FALL 23

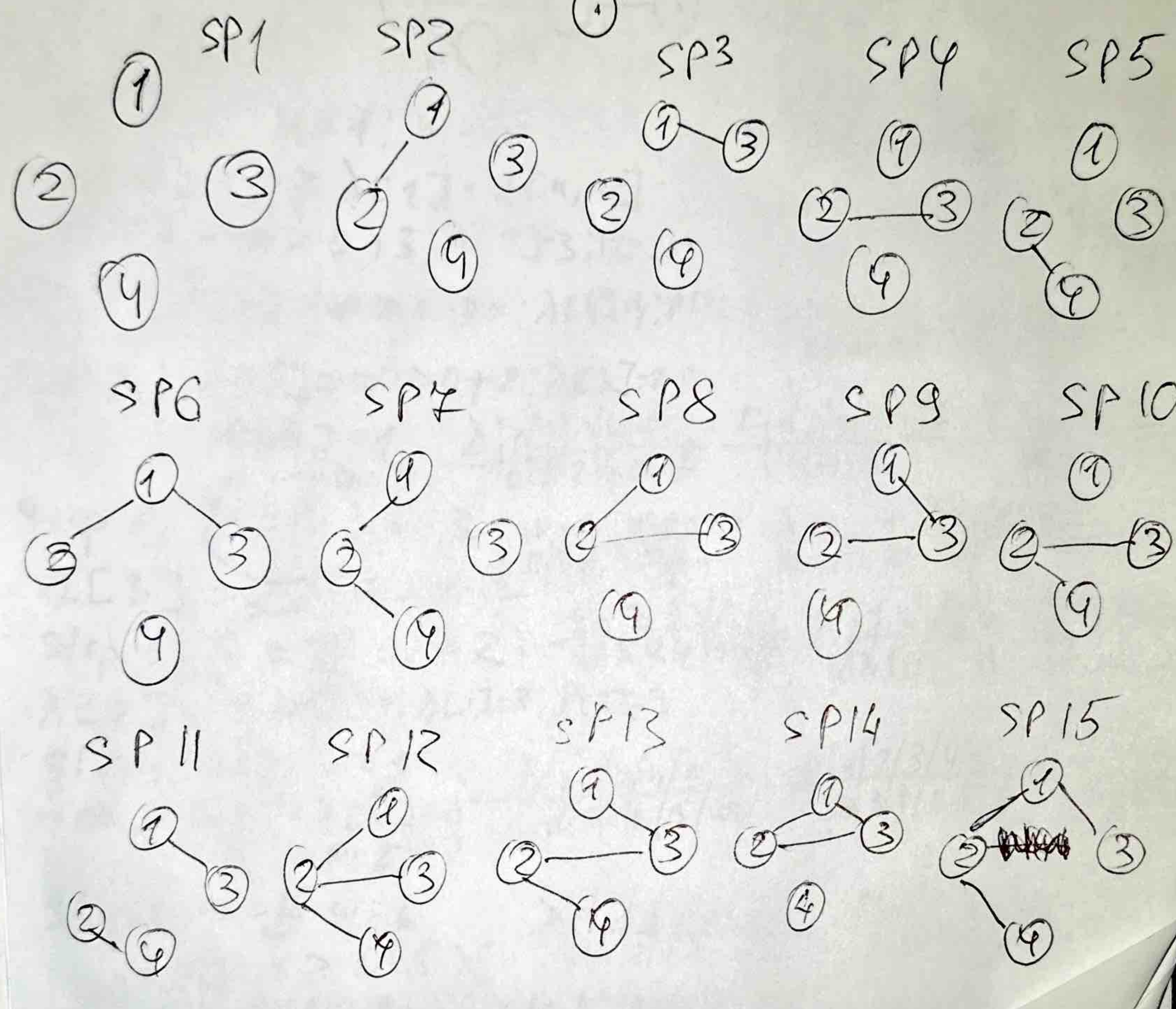
Name :

SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

2) Draw all the Spanning Subgraphs of the following graph. Which ones are spanning trees (mark them as "SP")





CSC382 --- Exam 2 V1 FALL 23

Name : SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

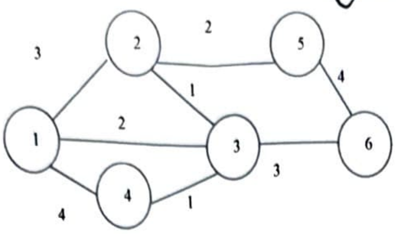
3) Apply Dijkstra's Algorithm using all the Data Structures λ, V, Parent and recreate the paths from the source (vertex labeled '1').

the order of the chosen vertices is incorrect -4

λ	1	2	3	4	5	6
	0	∞	∞	∞	∞	∞

F	1	2	3	4	5	6

T	1	2	3	4	5	6
	y	y	y	y	y	y



$if(\lambda[w] \geq \lambda[u] + d(u,w)$

$\lambda[w] = \lambda[u] + d(u,w)$

$P[w] = u$

Step 1; u = 1; w = 2

$\lambda[w] = \infty \geq \lambda[1] + d[u,w]$

$\lambda[2] = \infty > 0 + 3$; $\lambda[2] = 3$; $P[2] = 1$

λ	1	2	3	4	5	6
	0	3	∞	∞	∞	∞

F	1	2	3	4	5	6
		1				

T	1	2	3	4	5	6
	X	y	y	y	y	y

w = 4; $\lambda[4] = \infty > 0 + 4$ ✓; $\lambda[4] = 4$; $P[4] = 1$

λ	1	2	3	4	5	6
	0	3	∞	4	∞	∞

F	1	2	3	4	5	6
		1		1		

w = 3; $\lambda[3] = \infty > 0 + 2$, $\lambda[3] = 2$; P

$P[3] = 1$

λ	1	2	3	4	5	6
	0	3	2	4	∞	∞

F	1	2	3	4	5	6
		1	1	1		

T	1	2	3	4	5	6
	X	y	y	y	y	y

u=3

Step 2; u = 4 ; w = 3

$\lambda[3] = 2 \geq 4 + 1$ X

w = 6 ∞ > 2 + 3 $\lambda[6] = 5$

$P[6] = 3$

F	1	2	3	4	5	6
		3	1	1		3

λ	1	2	3	4	5	6
	0	3	2	4	∞	5

Step 3: u = 3 : w = 2

λ	1	2	3	4	5	6
	0	3	2	4	∞	5

F	1	2	3	4	5	6
	1	3	1	1		3

T	1	2	3	4	5	6
	X	y	X	y		

$\lambda[2] = 3 \geq 2 + 1$, $\lambda[2] = 3$; $P[2] = 3$

Step 4: u = 2 : w = 5

$\infty > 3 + 2$ $\lambda[5] = 5$

$P[5] = 2$

λ	1	2	3	4	5	6
	0	3	2	4	5	5

F	1	2	3	4	5	6
		3	1	1	2	3

T	1	2	3	4	5	6
	X	X	X	y	y	

Step 5 : u = 5 w = 6

$\lambda[6] = 5 \geq 5 + 5$ X

λ	1	2	3	4	5	6
	0	3	2	4	5	5

F	1	2	3	4	5	6
		3	1	1	2	3

T	1	2	3	4	5	6
	X	X	X	X	X	

Shortest Path: From 1 to 6: 1 → 3 → 6

From 1 to 5: 1 → 2 → 5 From 1 to 2 1 → 2

From 1 to 4: 1 → 4 From

From 1 to 3: 1 → 3

CSC382 --- Exam 2 V1

FALL 23

Name :

SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

4) Define the following graph-theoretic terms

a) A graph G=(V,E). is an ordered pairs such that V is a finite set whose elements' vertices and E is a set of unordered pairs of distinct vertices of V, call edges

b) A spanning subgraph of a graph G=(V,E). is subgraph of G, such that V'=V (it containing all original vertices of G.

c) A spanning tree of a graph G=(V,E). is a tree V' that contains all the vertices of G (V'=V), but no cycle

d) A trail between vertices *u* and *v* of G=(V,E). is a walk without repetition of edges

e) The degree of a vertex *u* of a graph G=(V,E). is number of edges incident at U

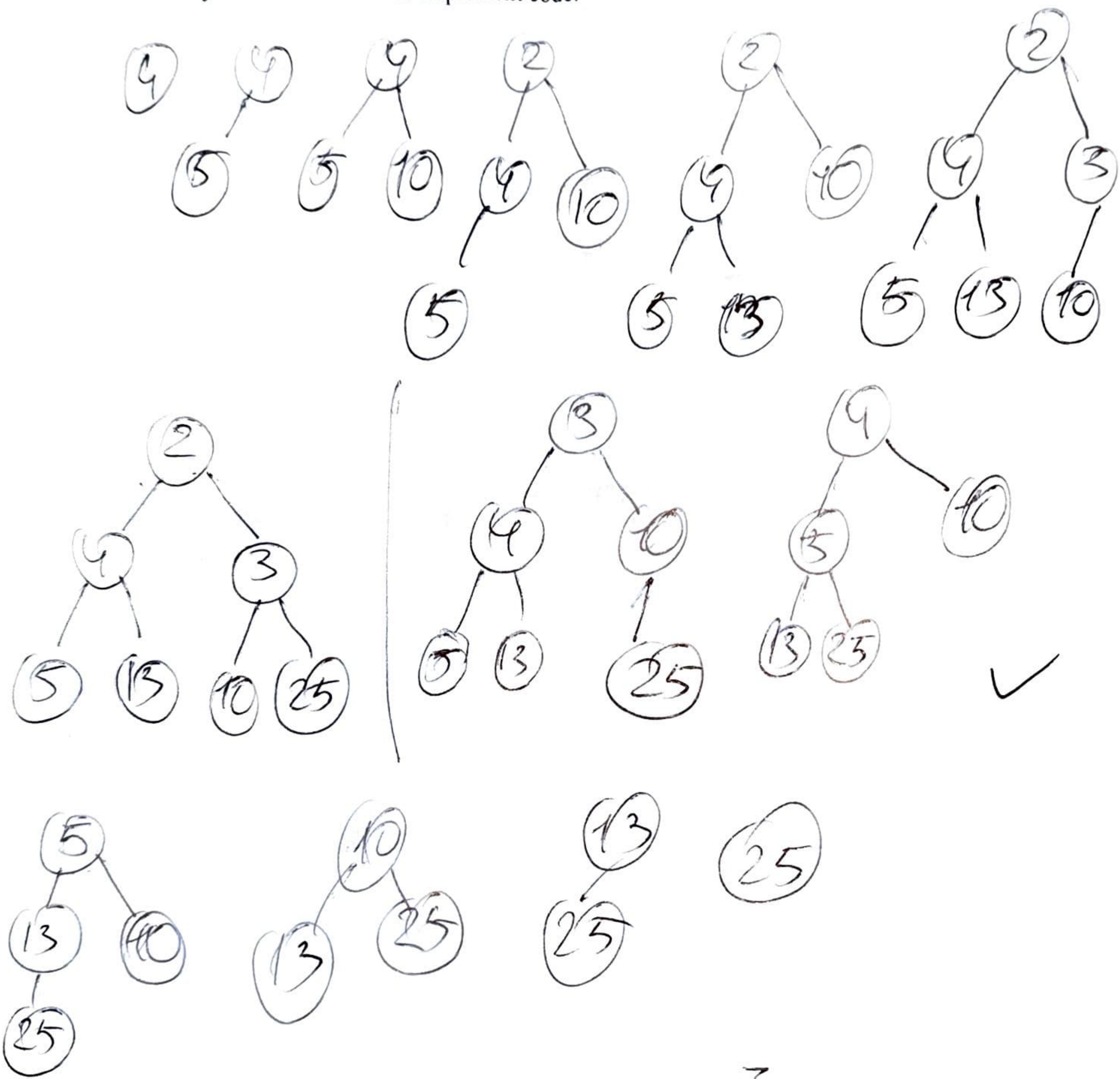
CSC382 --- Exam 2 V1 FALL 23

Name : SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

5)

a) Suppose that we have an array A={4,5,10,2,13,3,25}. Insert all the integers of the array into a Minheap one by one (using the *Insert* function of MinHeap explained in class) and delete the elements of Minheap one by one into A (using the *Delete* function of Minheap), so A is sorted. Do not use Heapify or any other internet version for Heapsort. Just draw the binary trees and no need to implement code.



A = {2, 3, 4, 5, 10, 13, 25}

CSC382 --- Exam 2 V1

FALL 23

Name :

SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

5b) Heapify is an approach that given an array, the (same) array get sorted. Heapify is equivalent to the Delete function of Min-heap (or Max-heap) as a priority queue. However, consider the situation where we have an Array, and we use Heapify to construct a Min-Heap (Max-Heap) from the array and later suppose that there are two more elements that need to be inserted into the resulting heap (same array); this scenario happens very often in Operating Systems. How would you go about that? From the answer explains why Heapify is not equivalent to applying the insert and delete functions of Max-heap as a priority queue.

No, It can not be done

why -3

Heapify is equal to delete function of priority queque

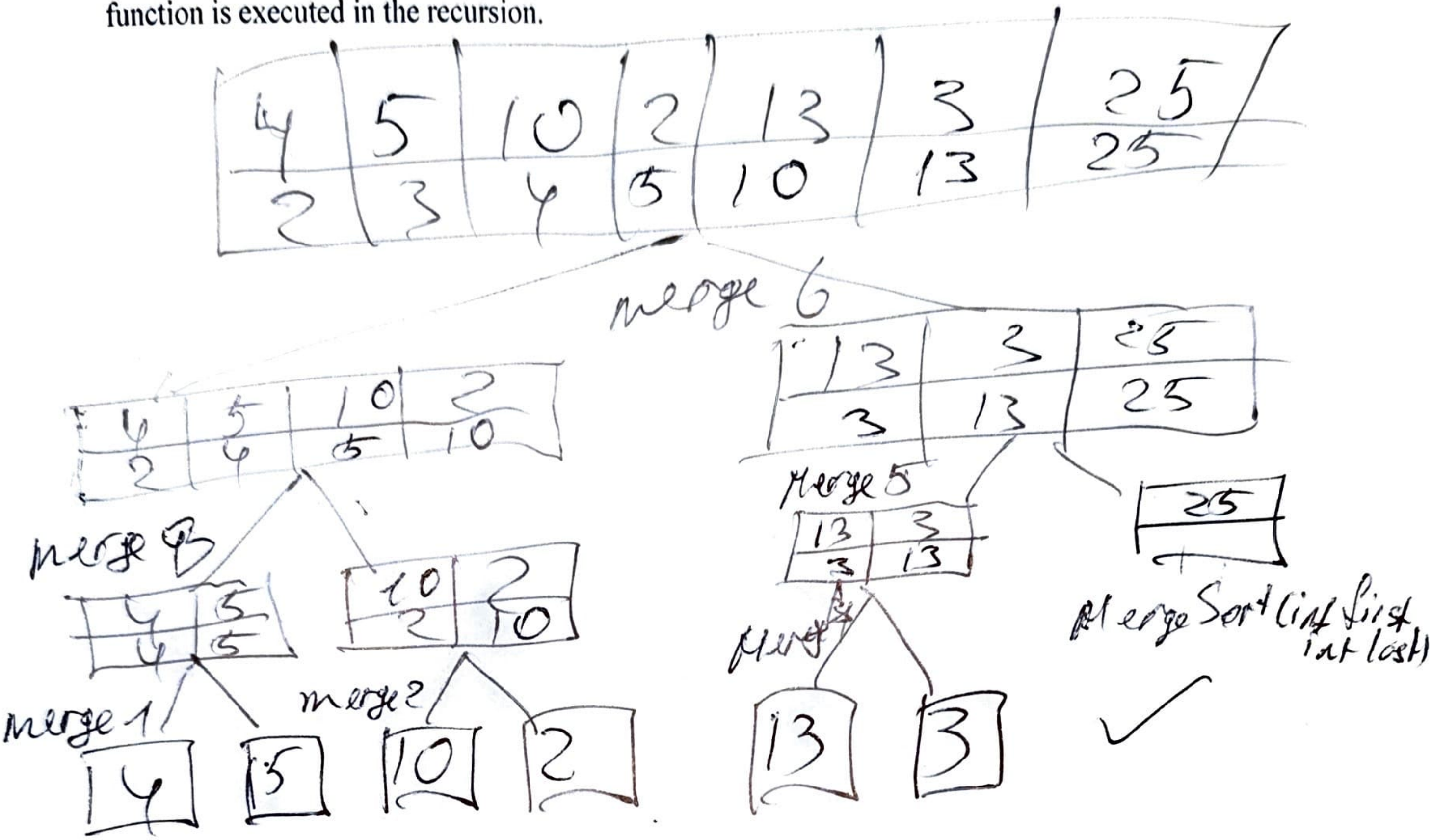
max heap from new array then sort jobs come later.

CSC382 --- Exam 2 V1 FALL 23

Name : SS#:

No Collaboration Permitted. Any similarities will represent a 0 for the exam's grade

6) a) Sort the array stated in question 5 using Mer-Sort and show the order in which the Merge function is executed in the recursion.



b) Give a brief explanation of the worst-case scenario of Merge-Sort

The worse case time complexity would be the worst case of a tree $\log_2 n$. Therefore the worst case if we have n elements to insert or delete, the time complexity becomes $O(n\log_2 n)$