

4CCS1DST – Data Structures

Lecture 4:

Exercises

Exercise 1

How would you modify the Parentheses Matching Algorithm if you wanted as output the pairs of positions of matched parentheses.

For example, for the input

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	
([(a	+	b)	*	c	+	d	*	e]	/	{	(f	+	g)	-	h	}))

the output should be:

(2,6) (1,13) (16,20) (15,23) (0,24)

Exercise 2

In the array-based implementation of Queue, “array location r is kept empty.” Consequently, when the queue is considered full, there is still one empty location in the array.

Could we put another element in that final empty location, and declare that the queue is full only if the size of the queue is equal to the size (length) of the array? What would be a problem with this approach and how that problem could be fixed?

Exercise 3

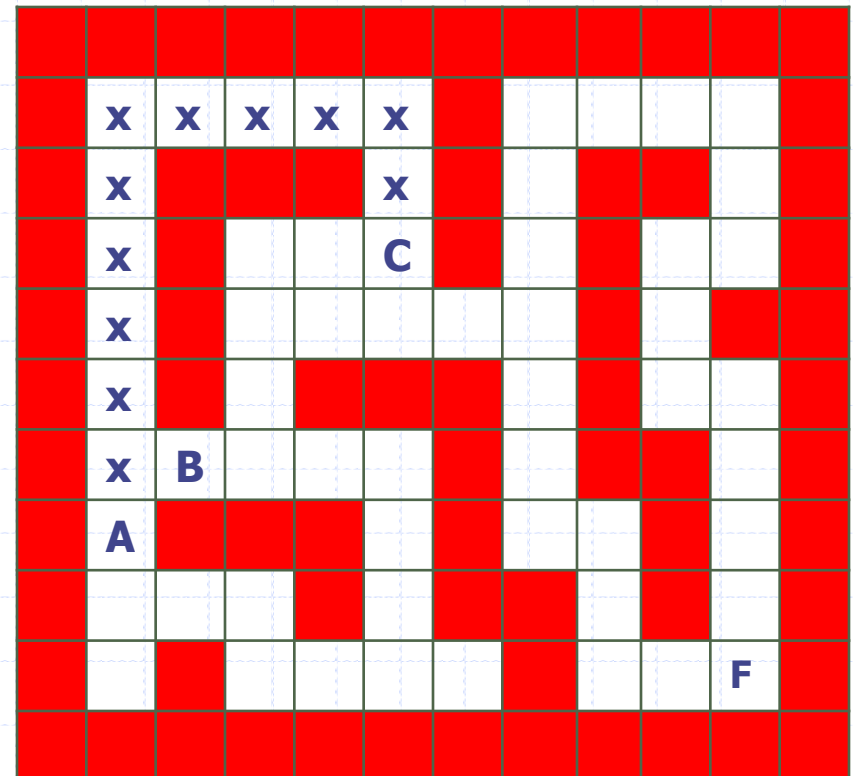
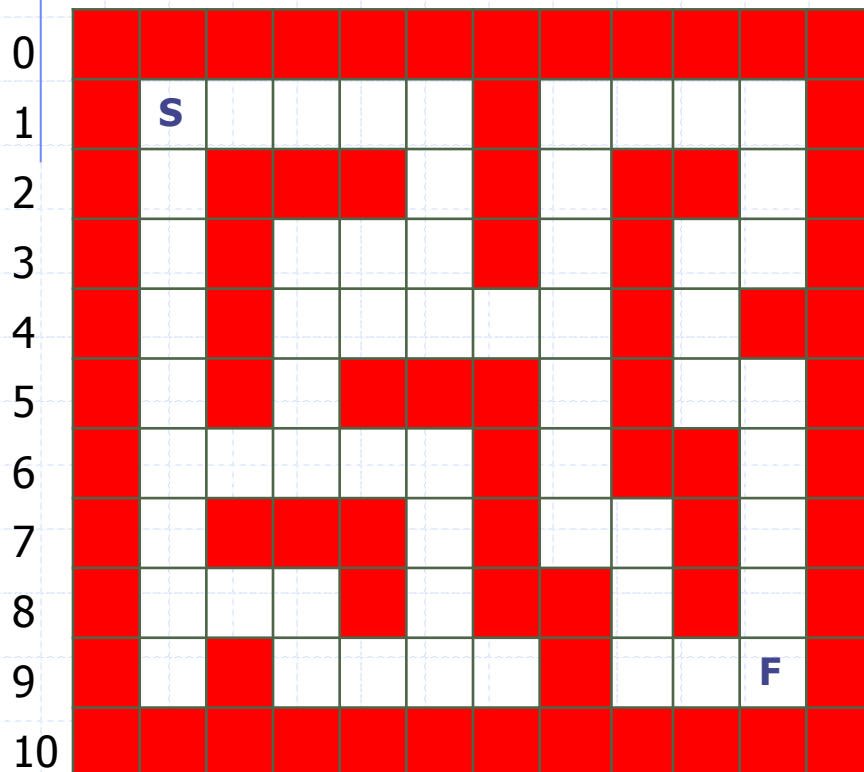
An example of a maze is shown on the next slide. Consider the following algorithm for exploration of such a maze to find out if the Finish location is reachable from the Start location. We can go from one location to a neighbouring location (two neighbouring locations share one side).

Algorithm:

- ❑ Each location is either unknown, discovered or explored.
- ❑ Initially Start is discovered and all other locations are unknown.
- ❑ All discovered locations are kept in a Queue.
- ❑ While Queue is not empty:
 - take (remove) a (discovered) location from Queue,
 - mark this location as explored,
 - mark all its unknown neighbouring locations as discovered and add them to Queue.
- ❑ Stop when Finish is discovered (Finish is reachable from Start) or when Queue becomes empty (Finish is not reachable).

Exercise 3 (cont.)

Show the status of each location (unknown, discovered, or explored) when Finish is discovered. Assume the neighbouring locations of the current location are considered in the order: S, E, N, W. The right diagram shows an intermediate state: the explored locations are marked with x; Queue is (A,B,C).



Exercise 3 (cont.)

What would happen, if we used Stack instead of Queue in this algorithm? Trace the computation showing the status of the locations (unknown, discovered, or explored) and the contents of the Stack. Assume the neighbouring locations of the current location are considered in the order: S, E, N, W.

