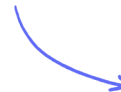


8.2 Algorithms for the Main Data Structures

Stacks / Queues / Linked Lists / Trees

Scan here to return to the course
or visit [savemyexams.com](https://www.savemyexams.com)



Total Marks

/11

- 1 A computer uses a stack data structure, implemented using an array, to store numbers entered by the user.

The array is zero based and has 100 locations

The main program initialises a new object of type stack with the identifier `mathsStack`.

Write **pseudocode** or **program code** to declare the object.

.....

.....

(2 marks)

- 2 Lucas writes a program that makes use of a circular queue. The queue stores the data entered into the program. An array is used to represent the queue.

The program needs two pointers to access and manipulate the data in the queue.

State the **purpose** of the two pointers and give an appropriate **identifier** for each.

.....

.....

.....

.....

(4 marks)

- 3 A program stores data in a linked list.

The current contents of the linked list are shown in **Fig. 3**, along with the linked list pointers.

headPointer	1
freeListPointer	4

location	data	pointer
0	"blue"	6
1	"red"	0
2	"green"	8
3	"orange"	NULL
4		5
5		7
6	"grey"	2
7		9
8	"purple"	3
9		NULL

The function `findNode` will search the linked list and return either the position of the node that contains the data item, or -1 if the data item is not found.

The data held in a node at location `x` can be accessed with `linkedList[x].data`. The pointer of the node at location `x` can be accessed with `linkedList[x].pointer`.

For example, using the linked list shown in **Fig. 3**: `linkedList[2].data` returns green. `linkedList[2].pointer` returns 8.

Complete the function, using **pseudocode** or **program code**.

```
function findNode(toFind, headPointer, linkedList)

    currentNode = .....

    while(currentNode != ..... )

        if linkedList[currentNode]. ..... == toFind

            return currentNode

        else

            currentNode = linkedList[.....].pointer

        endif

    endwhile
```

return

endfunction

(5 marks)