*Answer **all** the questions.*

**1.** Two of the most common computer operations are *sorting* and *searching*.

(a) Explain what is meant by sorting. *[2 marks]*

(b) Explain what is meant by searching. *[2 marks]*

(c) State **one** example of internal sort method and state its efficiency in *BigO notation*. *[2 marks]*

(d) State **one** example of search method and state its efficiency in BigO notation. *[2 marks]*

(e) Sorts are time consuming and it may be a good policy to avoid them where possible. Explain how this could be done. *[2 marks]*

**2.** The linked list is held in memory in a table, which has room for 100 entries. The first item on the list is pointed to by the pointer start.
All free locations in the table are linked and nextAvailable is a pointer to the next free location in the table.
Each node consists of a student's name and a pointer to the next item in the list. Pointer –1 is the *sentinel value*.
The table currently holds four entries in such a way that they can be retrieved in alphabetical order

start [0]

nextAvailable [4]

| | STUDENT'S NAME | NEXT STUDENT |
|---|---|---|
| 0 | Adams, Jose | 3 |
| 1 | Turner, Ivor | –1 |
| 2 | Kliss, Mary | 1 |
| 3 | Brown, Charlie | 2 |
| 4 | | 5 |
| 5 | | 6 |
| 6 | | 7 |
| . | | |
| . | | |
| 98 | | 99 |
| 99 | | –1 |

(a) Determine the new state of the table and pointers start and nextAvailable after name Lohy, Ann has been inserted in the list given above. *[3 marks]*

(b) Determine the new state of the table and the pointers start and nextAvailable after name Kliss, Mary has been deleted from the **original** list. *[3 marks]*

(c) Describe, by means of diagrams, or otherwise, how this list can be held in a memory as a dynamic data structure. *[4 marks]*

**1.** (a) Putting a list;
in (ascending or descending) order; ***[2 marks]***

(b) Locating a specific value (target);
in a list of values; ***[2 marks]***

(c) *Award **[2 marks]**, **[1 mark]** for name, **[1 mark]** for BigO notation.*
possible answers (sorts listed in the syllabus, accept other sorts):
bubble sort $O(n^2)$;
selection sort $O(n^2)$;
quick sort $O(n \log_2 n)$; ***[2 marks max]***

(d) *Award **[2 marks]**, **[1 mark]** for name, **[1 mark]** for BigO notation.*
possible answers(algorithms listed in the syllabus);
linear/sequential search O(n);
binary search $O(\log_2 n)$; ***[2 marks]***

(e) *Award **[2 marks]** for correct explanation, **[1 mark]** for vague answer.*
example answer:
sorts can be avoided by maintaining the data in correct order all the time (example linked list)
or
sorts can be avoided by keeping the items in a data structure such as a BST. ***[2 marks]***

**2**. (a) Award ***[1 mark]*** for correct column STUDENT'S NAME, ***[2 marks]*** for correct column NEXTSTUDENT (***[1 mark]*** for each changed link), ***[1 mark]*** for correct pointer nextAvailable. ***[3 marks max]***

start 0

nextAvailable **5**

| | STUDENT'S NAME | NEXT STUDENT |
|---|---|---|
| 0 | Adams, Jose | 3 |
| 1 | Turner, Ivor | -1 |
| 2 | Kliss, Mary | **4** |
| 3 | Brown, Charlie | 2 |
| 4 | **Lohy, Ann** | **1** |
| 5 | | 6 |
| 6 | | 7 |
| . | | |
| . | | |
| 99 | | -1 |

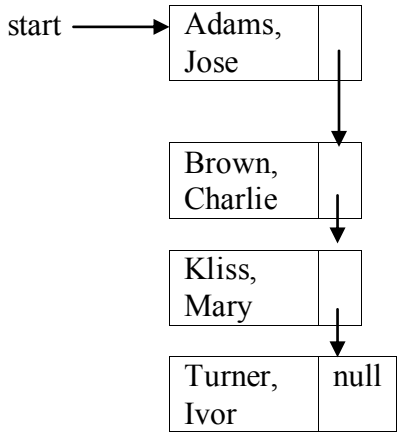
(b) Award ***[1 mark]*** for correct column STUDENT'S NAME (name Kliss Mary may not appear on the list).
Award ***[2 marks]*** for correct column NEXTSTUDENT (award ***[1 mark]*** for each changed link).
Award ***[1 mark]*** for correct pointer nextAvailable). ***[3 marks max]***

start 0

nextAvailable **2**

| | STUDENT'S NAME | NEXT STUDENT |
|---|---|---|
| 0 | Adams, Jose | 3 |
| 1 | Turner, Ivor | -1 |
| 2 | Kliss, Mary | **4** |
| 3 | Brown, Charlie | **1** |
| 4 | | 5 |
| 5 | | 6 |
| 6 | | 7 |
| . | | |
| . | | |
| 99 | | -1 |

(c)




*Award **[1 mark]** for any.*
pointer to the first node in the list;
correct node contents (data and link);
correct order of nodes;
all correct links and null;
although pointer nextAvailable is not needed in dynamic representations, marks are to be awarded for the following statements.
stating that nextAvailable could be a a pointer to last node;
or if ADT linked list is used-add a node at the end of the list is available method so pointer nextAvailable is not needed; ***[4 marks max]***