CSP2348 Data Structures

Workshop Test 2: Data structures and algorithms (Total Marks: 10)

Student ID: **Name:** .

Tasks: Attempt all five questions below. Note: all work(ings) must be shown!

1. (2 marks)
   Consider the development of software using either arrays or linked-lists.
   What factors will determine an implementation using arrays versus linked-lists?

Answer:

Array:

- Static structure: fixed length (cannot be changed once declared)
- Access: using an index, contagious allocated memory, suitable to random access;
- Operations: search operation can be very efficient for sorted array, but delete/insert algorithms are inefficient.

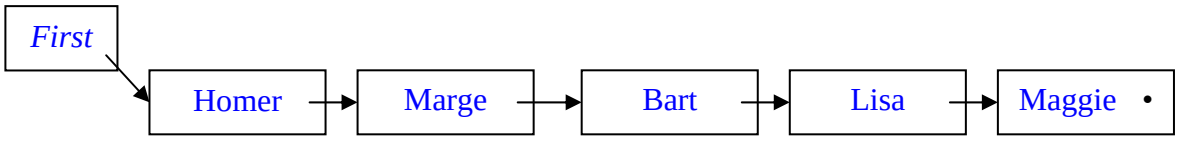
Linked list:

- Dynamic structure: the size of a linked list varies.
- Access: using link, unable to quickly locate/access elements (must traverse);
- Operations: efficient in data insertion/delation, suitable for storing data set that is frequently inserted/deleted, but infrequently in (value based) searching.

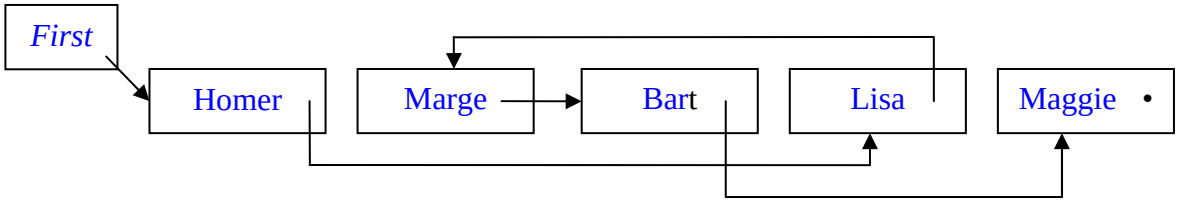
Therefore, **if the data set is dynamic and insert and delete operations occur frequently, it is better to use linked list, otherwise array data structure is better.**

2. (1 mark)
   Consider the following figure:
   What is the purpose of the SLL "Before" and "After" manipulation shown in the following diagrams? If we traverse the "After" SLL, what is the outcome of the visitations?

Before:



After:



Answer:

(a) To demonstrate how easy it is to restructure an SLL, simply by manipulating the links. There is no need to disturb the elements contained in the nodes. In this case, it moves "Lisa" to be the 2nd node (Note: it is not to swap two nodes with values "Marge" and "Lisa

(b) "Homer", "Lisa", "Marge", "Bart", "Maggie"

3. (1 mark)
What is the minimum possible depth of a binary tree with 357 nodes?

Answer:

A binary tree reaches minimum possible depth when it is balanced.

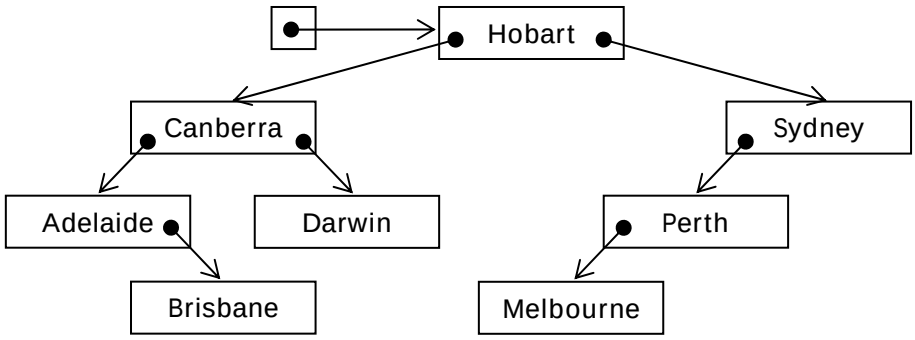
A balanced binary tree of depth *d* has at least 2^d nodes and at most $2^{d+1} - 1$ nodes, i.e. $2^d \le n \le 2^{d+1} - 1$, or $d \le \log_2 n < d+1$.

Let *n* =357. From above formula we have: d = *floor*(357) = 8

4. (3 marks)
Insert the following cities into an empty BST:
Hobart, Sydney, Perth, Canberra, Adelaide, Brisbane, Melbourne, Darwin.
After all the insertions, show the results of pre-order, in-order, and post-order, visitations of the BST.

Answer (reference only):

After insertion of all data,



Pre-Order: Hobart, Canberra, Adelaide, Brisbane, Darwin, Sydney, Perth, Melbourne

In-Order: Adelaide, Brisbane, Canberra, Darwin, Hobart, Melbourne, Perth, Sydney

Post-Order: Brisbane, Adelaide, Darwin, Canberra, Melbourne, Perth, Sydney, Hobart

5. (3 marks)
The family names of the most recent Prime Ministers of Australia are used as the keys of their data entries (see below). Use the hash function given below to construct a CBHT and an OBHT to represent these entries, respectively.

Whitlam, Fraser, Hawke, Keating, Howard, Rudd, Gillard
Hash(key) = key's initial letter – 'A'
table-size m = 26

Notes: This question was outsides the scope of the test 02 for this semester, and thus is not required.