Answer to the question number 1 ----------

A —>

Data - Data is a collection of raw, unorganized facts and details like text, observations, figures, symbols and descriptions of things etc. In other words, data does not carry any specific purpose and has no significance by itself. Moreover, data is measured in terms of bits and bytes – which are basic units of information in the context of computer storage and processing.

Information - Information is processed, organised and structured data. It provides context for data and enables decision making. For example, a single customer's sale at a restaurant is data – this becomes information when the business is able to identify the most popular or least popular dish.

Part 02 —>

Basically data structures are divided into two categories,

1. **Linear data structure**
2. **Non linear data structure**

LDS - In LDS, The data items are arranged in sequential order, one after the other.All the items are present on the single layer. The memory utilization is not efficient.The time complexity increases with the data size.Example: Arrays, Stack, Queue

NLDS - In NLDS, The data items are arranged in non-sequential order (hierarchical manner).The data items are present at different layers.Different structures utilize memory in different efficient ways depending on the need.Time complexity remains the same.Example: Tree, Graph, Map.

B —>

Control Structure - Control Structures are just a way to specify flow of control in programs. Any algorithm or program can be more clear and understood if they use self-contained modules called as logic or control structures.

The basic control structure in programming language are,

1. **Conditionals (or selection)**
2. **Loops (or iteration)**

Conditionals - Conditionals is that they allow you to control the flow of the code that is executed based on different conditions in the program. Example if statements, if-else statements.

Loops - "Loop statements" are nothing more than the automation of multi-step processes by organizing sequences of actions, and grouping the parts that need to be repeated. Example For loops, while loops and repeat loops.

C —>

Same as 1 - c (Autumn 2018) and 1 - d (Spring 2018)

Answer to the question number 2 ----------

A —>

Same as 2 - a (Spring 2018)

B —>

Same as 2 - b (Autumn 2018)

C —>
i) Same as 2 - C (Autumn 2018) , (ii) er pic down below

2or(c)

(iii)

B(3:10, 1:15, 10:20)

$L_1 = 10-3+1$; $L_2 = 15-1+1$; $L_3 = 20-10+1$

$L_1 = 8$; $L_2 = 15$; $L_3 = 11$

Base (B) = 400

~~B[5~~, B[5, 10, 15]

$\therefore E_1 = 5-3$; $E_2 = 10-1$; $E_3 = 15-10$

$E_1 = 2$; $E_2 = 9$; $E_3 = 5$

∴ row major order

$E_1 L_2 = 2 \times 15 = 30$

$E_1 L_2 + E_2 = 30 + 9 = 39$

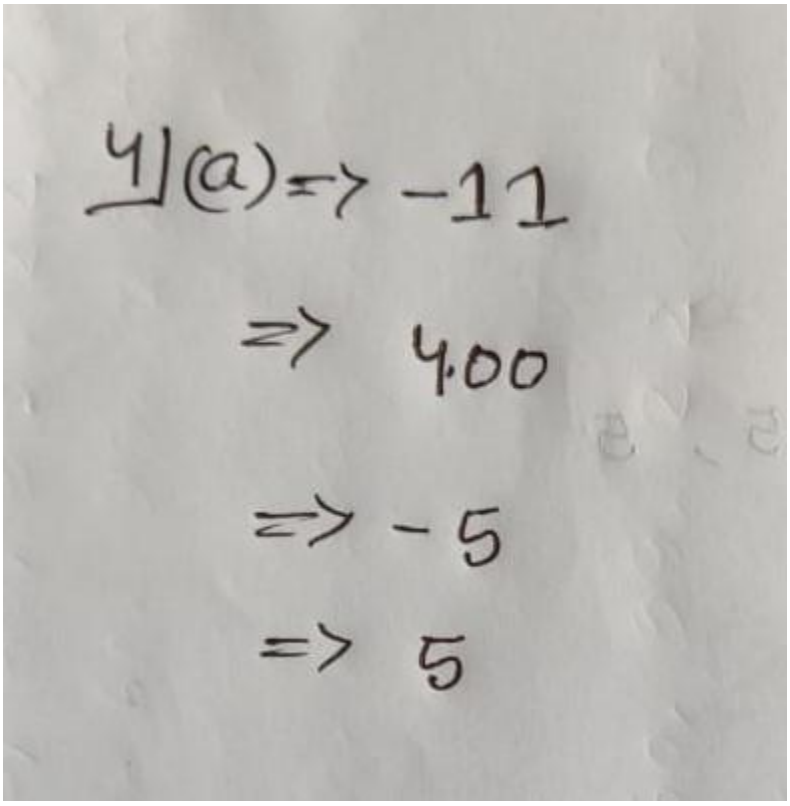
$(E_1 L_2 + E_2) \times L_3 = 39 \times 11 = 429$

$(E_1 L_2 + E_2) \times L_3 + E_3 = 429 + 5 = 434$

$\therefore LOC(B[5, 10, 15]) = 400 + 4 \times (434)$

$= 2136$

Answer to the question number 4 ---------- Picture given below



4(c)

(c) Sort the following list of numbers using bubble sort algorithm.

⇒ Here DATA is an array with N ~~element~~ elements. This algorithm Sorts the elements in DATA.

1. Repeat Steps 2 and 3 for K=1 to N-1
2. Set PTR := 1
3. Repeat while PTR <= N-K:
   (a) If DATA[PTR] > DATA[PTR+1], then Interchange DATA[PTR] and ~~DATA[PTR]~~ DATA[PTR+1]
   (b) Set PTR := PTR+1
4. Exit.

Sort

Step-1

44 33 11 55 77 90 40 60 99 22 88 66

33 44 11 55 77 90 40 60 99 22 88 66

33 11 44 55 77 90 40 60 99 22 88 66

33 11 44 55 77 90 40 60 99 22 88 66

33 11 44 55 77 90 40 60 99 22 88 66

33 11 44 55 77 90 40 60 99 22 88 66

33 11 44 55 77 40 90 60 99 22 88 66

33 11 44 55 77 40 60 90 99 22 88 66

33 11 44 55 77 40 60 90 99 22 88 66

33 11 44 55 77 40 60 90 22 99 88 66

33 11 44 55 77 40 60 90 22 88 99 66

33 11 44 55 77 40 60 90 22 88 66 99

Step-2

33 11 44 55 77 40 60 90 22 88 66 99

11 33 44 55 77 40 60 90 22 88 66 99

11 33 44 55 77 40 60 90 22 88 66 99

11 33 44 55 77 40 60 90 22 88 66 99

11 33 44 55 77 40 60 90 22 88 66 99

11 33 44 55 ~~77~~ 40 77 60 90 22 88 66 99

11 33 44 55 40 60 77 90 22 88 66 99

11 33 44 55 40 60 77 90 22 88 66 99

11 33 44 55 40 60 77 22 90 88 66 99

11 33 44 55 40 60 77 22 88 90 66 99

11 33 44 55 40 60 77 22 88 66 99

last step

11 22 33 40 44 55 60 66 77 88 99