May 2017 HL P1

Section A

1. Describe the use of beta testing. [2]

It is when an application/program is tested/ to get feedback (for errors/improvements);
By releasing it to the general public/users/stakeholders before final release;

2. Identify **two** causes of data loss. [2]

*Award up to **[2 max]**.*
Natural disaster/power failure;
Accidental deletion;
Malicious activities (hacking/theft/viruses);
Hardware/software/system failure;
Bad integration/migration of systems/data;
Transmission error;
Lossy compression when the original no longer available only the modified version;

3. Identify **two** reasons for releasing a software update. [2]

*Award up to **[2 max]**.*
Natural disaster/power failure;
Accidental deletion;
Malicious activities (hacking/theft/viruses);
Hardware/software/system failure;
Bad integration/migration of systems/data;
Transmission error;
Lossy compression when the original no longer available only the modified version;

4. Explain why cache memory can speed up the processing within a computer. [2]

Cache memory can be accessed/is faster than RAM;
It is used to hold common/expected/frequently used data/operations;
Closer to CPU than RAM/situated between RAM and CPU/on same board as CPU/with faster read/write speed;
Cache memory is static RAM and this memory doesn't need to be constantly refreshed;

5. One of the functions of an operating system is memory management.

Describe how this function prevents the system from crashing when more than one program is run at the same time. [2]

The OS allocates (and deallocates) specific sections of memory to each program/process/module;
This ensures that the memory assigned to one program is not overwritten;
Uses secondary/virtual memory to allow more processes to run simultaneously;
Note: *Do not accept vague reasons.*

6. Identify any **two** of the layers of the OSI model. [2]

*Award up to **[2 max]** for any two of the following:*
Physical layer;
Data link layer;
Network layer;
Transport layer;
Session layer;
Presentation layer;
Application layer;

7. Explain how compression of data may lead to negative consequences. [3]

Some compression methods (lossy) discard data;
Decompression will not return the complete file/some detail will have been removed;
Which in some cases *eg* audio/video may be unacceptable;
If original not saved/lost there is no way to recover it;

8. Explain how the use of media access control (MAC) addresses can improve security. [3]

*Award up to **[3 max]**.*
The MAC address identifies a specific device (network card/controller); MAC address checked against list of approved addresses/whitelist If not on list access to network is denied;
Prevents unauthorised access/makes access more difficult/(unless the NIC is cloned)/providing an extra layer to authentication process;
data sent to a specific MAC address can only be accessed on that device;

9. Identify the components of a node in a doubly linked list. [3]

Data;
A pointer/reference to the previous node;
A pointer/reference to the next node;

10. Outline the reason why recursive solutions can be memory intensive. [2]

*Award up to **[2 max]**.*
A recursive call involves the use of stacks;
For storing/pushing on/popping out data/ return addresses/return values *etc*;
If many recursive calls are made, the memory usage can be very large;

11. Identify **two** features of an autonomous agent acting within a larger system. [2]

(Have sensors that) can react to external stimuli/ changes in its environment;
without needing to contact any central/outside source/by making its own decisions/acts independently;

Section B

12. A large company has taken over another business. This takeover has required various changes to be made. One of the changes requires data migration.

(a) (i) Define the term *data migration*. [1]

Data migration is the transfer of data from one system/storage device to another; (Importing the data/database from the company to the company system)

(ii) Describe **two** problems, concerning data migration, which the company may have to overcome. [4]

*Award **[1]** for identifying a problem that may arise and **[1]** for an expansion / elaboration up to **[2 max]**.*

Mark as 2 and 2

Data loss; Due to transmission faults/lack of adequate storage;

Incompatible file formats;
Which could lead to incomplete or incorrect data transfer;

Different file structures;
Which will result in a mismatch of data, for example in customer records;

Validation rules differ between companies;
Which could lead to inconsistent/incorrect results;

Different character sets might be used;
Which could lead to inconsistent/incorrect results;

Different languages might be used;
Leading to translation issues;

Data corrupted when transferring (data) files;
And not usable at destination;

***Note:** Accept data loss as an issue unless the expansion is a repeat of the second issue identified.* **[4**

(b) Other than data migration, describe two aspects of change management that may arise from this takeover. [4]

*Award **[1]** for identifying each aspect of change management that may arise and **[1]** for an expansion / elaboration up to **[2 max]**.*

Mark as 2 and 2

Workforce issues;
Such as redundancy/retraining;

The time frame involved;
In merging the two systems;

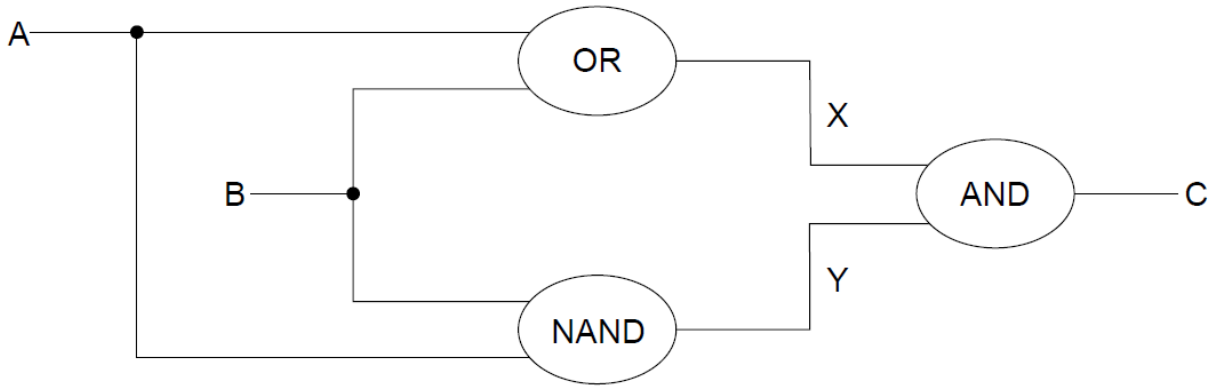
Testing;
Of the combined systems/new data;

Data entry;
If migration not possible;

Costs involved;
In the aligning of the two systems;

Changeover decisions;
Such as parallel running;

The company produces industrial chemicals. One of the chemical processes is represented by the following logic diagram.



(c) (i) Construct the truth table corresponding to this diagram. [3]

*Award up to **[3 max]**.*
If intermediate columns are not used:
If A, B and C are all correct, award 3 marks
If A, B and three outputs from C are correct, award 2 marks
If A and B are correct, award 1 mark

If intermediate columns are used:
If C is not correct and intermediate columns are used:
*Award **[1]** mark for Columns A and B being correct*
*Award **[1]** mark for either X or Y or both being correct*
If A, B and C are all correct, award 3 marks

A	B	NOT(A AND B)	A OR B	C=NOT(A AND B) AND (A OR B)
0	0	1	0	0
0	1	1	1	1
1	0	1	1	1
1	1	0	1	0

(ii) Identify the single logic gate that is equivalent to this diagram. [1]

XOR gate;
***Note**: Follow through from an incorrect truth table in part (c)(i).*
Accept A XOR B = C

(d) Outline how truth tables can be used to test that any two logic diagrams are equivalent. [2]

Two truth tables can be compared;
The same outputs from the same inputs, the circuits are equivalent;

13. (a) By copying the table below, trace the following algorithm using the data in the collection DATA. Note: B and C are also collections and are initially empty.

```
DATA = {2,4,-1,3}
loop while DATA.hasNext()
  A = DATA.getNext()
  if A >= 0 then
    if A mod 2 = 0 then
      B.addItem(A)
    else
      C.addItem(A)
    end if
  end if
end loop
```

DATA.hasNext() ?	A	A >= 0?	A mod 2 = 0?	Contents of B	Contents of C

[3]

*Award **[1]** mark if evident from the trace table that the loop executes exactly 4 times*
***[1]** mark for correct final contents of B (2 and 4)*
***[1]** mark for the correct final contents of C (3)*

DATA.hasNext() ?	A	A >= 0 ?	A mod 2 = 0?	Contents of B	Contents of C
true	2	true	true	2	
true	4	true	true	2, 4	
true	-1	false	*true	*2, 4	
true	3	true	false	*2, 4	3
false					

***Note:** These cells (*) could be left empty*

(b) Outline the steps involved in performing a binary search on an array of ascending numbers.
Note: you can assume that the search value is present in the array and that initially *LOW* is the index of the first value in the array and *HI* is the index of the last value. [4]

*Award **[4]** marks as follows:*

A position/index of the middle value calculated from HI and LOW;
If search value equals to value in the array at this position, then end;
Otherwise, change HI or LOW according to whether search value is above or below;
Repeat the process until search value is found;

Example answer 1:
1. Calculate MID (the midpoint of LOW and HI)
2. If ARRAY[MID] is the search value, end, else
3. If search value < than ARRAY[MID], HI = MID-1, else LOW = MID+1
4. Repeat steps 1, 2 and 3 until found

Example answer 2:
Find/calculate the midpoint/median value between HI and LOW;
If the array value at midpoint equals the search value; then the search value is found;
If search value is greater than the array value at midpoint, set the midpoint as a new LOW and repeat the process - if search value is less than the array value at midpoint, set midpoint as a new HI;
Repeat until search value is found; **[4]**

(c) A collection called *NUMBERS* is to be searched to see if it contains a specified value.
Construct an algorithm in pseudocode to perform the following:

- input the number, *S*, to be searched for
- read in the values from the *NUMBERS* collection into the array *D*.
  Note: you can assume that the array is large enough and that the collection is **not** empty
- perform a linear search for *S* on the array *D*
- output the message "*found*" or "*not found*" as appropriate. [5]

*Award up to **[5 max]** as follows:*
Example answer 1:
***[1]** mark for a loop through collection NUMBERS*
***[1]** mark for the assignment into the array*
***[2]** marks for a correct linear search, **[1]** mark for the loop and 1 mark for the comparison and use of flag*
***[1]** mark for correct input of search value and output message*

```
input S // search item
COUNT = 0
loop while NUMBERS.hasNext()        // or … while NOT
                                    //NUMBERS.isEmpty()
  D[COUNT] = NUMBERS.getNext()
  COUNT = COUNT + 1
end loop
FOUND = false
X = 0
loop while (X<COUNT) and (FOUND == false) //accept for loop
  if S == D[X]
    FOUND = true
    X = X + 1
  end if
end loop
if FOUND == true
  output "found"
else
  output "not found"
end if
```

Example answer 2:

(in this example assignment into array done in same loop as reading from NUMBERS)

1 mark for a loop through collection NUMBERS
1 mark for the assignment into the array(**do not award this mark if flag is used to terminate/break the while loop)**
2 marks for correct linear search (through D), 1 mark for correct comparison, 1 mark for flag
1 mark for correct input of S and output message

```
input S // search item

COUNT = 0
FOUND = false
loop while NUMBERS.hasNext()      // or … while not NUMBERS.isEmpty()
  D[COUNT] = NUMBERS.getNext()
  if   S == D[COUNT]
    FOUND = true
  end if
  COUNT = COUNT + 1
end loop
if FOUND
  output "found"
else
  output "not found"
end if
```

[5]

A binary search can be performed on the array *D*, if the values in *D* are in ascending order.
As the values are being read from *NUMBERS* into *D* they are checked to see if they are in order.

(d) Without writing pseudocode, suggest how this check could be performed. [3]

Award up to ***[3 max]***.
[1] *mark for setting and changing a flag*
[1] *mark for starting from the 2nd entry and checking all values as being entered*
[1] *mark for comparing the current value with the previous one*

Note*: Do not accept pseudocode/code.*

14. A laptop computer supplements its primary memory by making use of virtual memory.

(a) Outline the use of paging in relation to virtual memory. [2]

Award up to ***[2 max]***.
Paging allows similar-sized sections of data;
identified by specific addresses;
To be swapped between RAM and secondary memory/storage;

Example answer:
Main memory is divided into page files/swap files/sections so that data from these sections can be transferred to and from secondary storage. The addresses of these sections in main memory are linked to virtual addresses in secondary memory and the links stored in a table. When a page is needed, the table is accessed to recall the page into the correct locations in main memory.

The laptop has 1GB of random access memory (RAM) and a single processor.
The laptop is using one of the latest operating systems to run multimedia gaming programs.

(b) Explain the limitations and consequences of using the laptop for this purpose. [3]

Award up to ***[3 max]***.
The listed software would require a lot more RAM to run efficiently;
Paging/VM would be regularly used;
In order to load the required modules;
Thus slowing down the operation of the laptop;
The laptop does not have a dedicated graphics processor;
Which would be required for a multimedia game;
This would result in low quality graphics;

One of the laptop's game applications stores the data relating to the different actions of one of its characters in a stack.

(c) Suggest **one** reason why the character's actions might be stored in a stack. [2]

A stack is a LIFO data structure;
In the game it allows the "undo" function to work / movements/steps can be retraced;
Note*: Allow any reasonable answer for the second marking point.*

At specific moments during the game this data is read from the stack (*S*) into a queue (*Q*).

(d) Using appropriate access methods for stacks and queues, construct an algorithm that reads the data from the stack and enters it into the queue. You should assume that the queue structure exists and that both structures are of a fixed size. [6]

*Award up to **[6 max]** as follows.*
Example answer 1:
*Award **[1]** for while loop;*
*Award **[2]** for the correct condition in while statement (stack is not empty **and** queue is not full) award **[1]** in case of the minor error;*
*Award **[1]** for pop. (**Note:** Do not allow alternatives)*
*Award **[1]** for enqueue; (**Note**: Allow alternatives)*
*Award **[1]** for the output only if stack is not empty after the loop;*

```
loop while NOT S.isEmpty() and NOT Q.isFull()
  N = S.pop()
  Q.enqueue(N)
end loop
if NOT S.isEmpty() //accept NOT S.isEmpty()and Q.isFull()
  output "incomplete process"
end if
```

Example answer 2:
*Award **[1]** for the correct use of flag throughout.*
*Award **[1]** for the while loop*
*Award **[1]** for checking `!(S.isEmpty())`*
*Award **[1]** for `pop`. (**Note:** Do not allow alternatives.)*
*Award **[1]** for checking if Q is full*
*Award **[1]** for enqueue; (**Note**: Allow alternatives.)*
*Award **[1]** for output*

```
FLAG = true
loop while !(S.isEmpty())
  N = S.pop()
  if !(Q.isFull()
    Q.enqueue(N)
  else
    FLAG = false
    break
    // can be missed out or alternative used
  end if
end loop
if (FLAG = false)
  output "incomplete process"
end if
```

(e) Outline **one** advantage of making the queue dynamic. [2]

*Award **[2]** marks, **[1]** for advantage, **[1]** for expansion.*

A dynamic queue would not have predetermined/fixed size;
Memory is allocated as required/efficient use of memory/flexible size//there is always sufficient memory to accommodate the queue;

15. The collection *WEATHER* contains the temperatures that have been measured for one city over the course of **one week**, starting on Monday and ending on Sunday. Each day, 24 readings were taken, one each hour, the first being at 00:00, the second at 01:00 and so on. The data is stored in chronological order with the data for Monday stored in the collection first, followed by Tuesday and so on.

(a) State the total number of readings that were taken during this week. [1]

168; **[1]**
***Note:** Award **[1]** for "24 x 7" seen.*

(b) Construct the algorithm to read this data into a 2D array, *A*, that would allow the temperature on a specific day at a specific time to be accessed directly. [4]

*Award up to **[4 max]** as follows:*
Example answer 1:
*Award **[1]** for any nested loop.*
*Award **[1]** for the correct nested loop.*
*Award **[1]** for the correct assignment to `A`.*
*Award **[1]** for the correct retrieval from `WEATHER`.*

```
loop for DAY from 0 to 6
  loop for HOUR from 0 to 23
     A[DAY][HOUR] = WEATHER.getNext()
         // A[DAY][HOUR] = WEATHER.getData()
  end loop
end loop
```

Example answer 2:
*Award **[1]** for initialization of POS and correct increment within the loop*
*Award **[1]** for the correct loop.*
*Award **[1]** for the correct calculation of DAY*
*Award **[1]** for the correct calculation of HOUR*
*Award **[1]** for the correct assignment to A(correct retrieval from WEATHER)*

```
POS=0
loop while WEATHER.hasNext() //accept not WEATHER.isEMPTY()
   DAY=POS div 24
   HOUR=POS mod 24
   POS=POS+1
   A[DAY][HOUR]=WEATHER.getData() // WEATHER.getNext()
end loop
```

(c) Construct the algorithm that will output the day, as a word (for example Tuesday), on which the highest temperature was recorded. [6]

*Award up to **[6 max]** as follows:*

Example answer 1:
*Award **[1]** for initialization of HIGHEST*
*Award **[1]** for initialization of NAMES*
*Award **[1]** for the correct outer loop*
*Award **[1]** for the correct inner loop*
*Award **[1]** for the correct comparison and the assignment of HIGHEST*
*Award **[1]** for the assignment of `MAX_DAY` within if statement*
*Award **[1]** for the correct output statement*

```
HIGHEST = A[0][0] //accept for example HIGHEST = -100
NAMES=["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"]
// array NAMES may contain other names, like "Monday", "Tuesday", etc.
loop for DAY from 0 to 6
  loop for HOUR from 0 to 23
    if A[DAY][HOUR] > HIGHEST
       HIGHEST = A[DAY][HOUR]
       MAX_DAY = DAY
    end if
  end loop
end loop
output(NAMES[MAX_DAY])
```

Example answer 2:
*Award **[1]** for initialization of HIGHEST.*
*Award **[1]** for the correct outer loop*
*Award **[1]** for the correct inner loop*
*Award **[1]** for the correct comparison and assignment of HIGHEST*
*Award **[1]** for the assignment `MAX_DAY` within if statement*
*Award **[1]** for the if statement after the outer loop (accept switch statement)*
*Award **[1]** for the correct output statement (may be written within if statement)*

```
HIGHEST = A[0][0] //accept for example HIGHEST=-1000
  loop for DAY from 0 to 6
    loop for HOUR from 0 to 23
       if A[DAY][HOUR] > HIGHEST
          HIGHEST = A[DAY][HOUR]
          MAX_DAY = DAY
       end if
    end loop
end loop
if MAX_DAY == 0 then
   D = "Monday"
else if MAX_DAY == 1 then
   etc….
end if
output D
```

The process described at the start of question 15 is extended so that each week the value and date of the highest temperature recorded that week are stored chronologically in a collection, *HIGHEST*.

At any point in time, the data from the *HIGHEST* collection can be read into a suitable data structure that will allow the details of the highest temperatures recorded to be output in descending order. The structure is chosen in order to minimize processing.

(d) Explain how a suitable data structure can be constructed and used for this purpose. [4]

*Award up to **[4 max]**.*
***Note:** Do not give any marks if the proposed structure is not dynamic.*

A binary search tree could hold a series of nodes each of which contains the data for each reading;
and two pointers (to the left and right child);
Each node is inserted according to the temperature, higher temperature to the left, lower to the right;
An in-order traversal would output the data sorted in descending order of temperature;

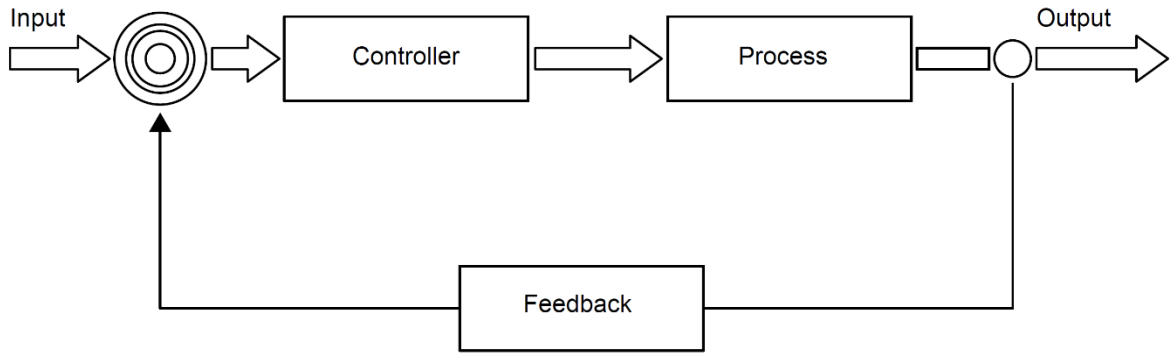
Note: Linked list would not minimise processing but should get credit.
*Award up to **[2 max]**, **[1]** mark for correct explanation on how it can be constructed and **[1]** mark for correct explanation on how it can be used to output data in descending order;*

16. An embedded system is used to control the speed of an electric motor.

(a) With reference to the example, above, define an embedded system. [2]

*Award up to **[2 max]**.*
An embedded system is a system that performs a specific/dedicated function;
For a larger system for which it is physically part of:
In this example, the control system is actually part of the electric motor;

The diagram shows the main components in a typical negative feedback system.



The control system for the electric motor consists of a negative feedback loop.

(b) Outline the desired outcome of any feedback system. [2]

To control the output (values);
By keeping it close to the reference/input value;

OR

To control the system performance/behaviour of the system/;
by routing back the output values as input values;
(which are used to make decisions on changing/modifying the performance of the system/behaviour of the system);

(c) List the steps involved in the feedback system for the electric motor. You should make use of the appropriate technical terms of the control system process. [6]

*Award up to **[6 max]**.*
The desired speed is set/inputted and the (speed) sensor / input transducer measures the (current output) speed;
This analog signal (from the speed sensor) is converted into digital;
And sent back to the controller/processor;
The error/difference between the output speed and desired/input speed is calculated and compared to pre-set values to determine whether speed should be adjusted;
If the speed/power should be adjusted by an amount dependent upon the error then signal is sent to the output (transducer);
The above process is repeated continually;

A more complex control system is used to monitor and control the functioning of a power station. This includes a dedicated operating system with sensors and output transducers (actuators) placed at various places around the power station.

(d) Explain the interaction between the components identified above, if interrupts are generated by the sensors. [5]

The sensor sends an interrupt to the operating system;
When it measures a value outside of the normal range;
Such as an abnormal temperature/pressure;
The operating system sends a signal to operate an actuator (output transducer);
In order to bring the system back to normal;