Nov 2016 HL P1

Section A

1. State **three** potential usability issues with cell phones. [3]

*Award up to **[3 max]**.*
Has a small (touch-sensitive) screen;
Uses batteries for power;
No hard disk drive / small memory;
Reliability / Network coverage issues;
Over in warm weather;
Too many steps to access a particular feature;
etc.

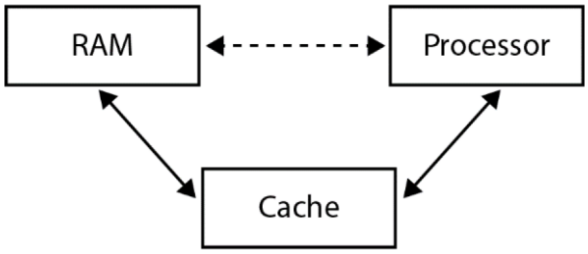
2.

(a) State the purpose of cache memory. [1]

Is used to save time in accessing RAM;

(b) Draw a diagram to show the relationship between random access memory (RAM), the processor and cache memory.

*Award **[1]** for the cache between RAM and the processor.*



3. Outline **one** advantage and **one** disadvantage of wireless networks. [4]

*Award **[1]** for stating an advantage and **[1]** for an expansion.*
*Award **[1]** for stating a disadvantage and **[1]** for an expansion.*

Advantages and disadvantages such as:
- Ease of use for mobile users
- Connectivity between different locations
- Reliability
- Cost
- Security
- Change in working patterns
- Health issues

etc.

Example answer:

Advantage:
Ease of use for mobile users;
As they can work in many different locations;

Disadvantage:
Security issues;
As wireless transmissions are easily intercepted;

4. Construct a truth table for the Boolean expression `not (A xor B) and C`.
Use the following headings in your table.

A	B	C	A XOR B	NOT (A XOR B)	NOT (A XOR B) AND C

[4]

A	B	C	A XOR B	NOT (A XOR B)	NOT (A XOR B) AND C
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	1	0	1	1

*Award up to **[4 max]** as follows.*
*Award **[1]** for all 8 sets of input values correct.*
*Award **[1]** for correct A XOR B column.*
*Award **[1]** for correct NOT (A XOR B) column.*
*Award **[1]** for correct NOT (A XOR B) AND C column.*

5. Many different people and organizations upload scientific materials to the internet.
A student uses data from the internet in a science project.

Outline **two** ethical issues concerning this use of the internet. [4]

*Award **[1]** for an ethical issue, **[1]** for an explanation, for two issues up to **[4 max]**.*

Points to be discussed:
The data/information is deliberately incorrect;
The data/information has not been validated;
Intellectual property issues;
Plagiarism;

6. Consider the following recursive algorithm *FUN(X, N)*, where *X* and *N* are two integers.

```
FUN(X, N)
if N<=0 then
  return 1
else
  return X*FUN(X, N-1)
end if
```

The *return* statement gives the value that the algorithm generates.

(a) Determine how many times multiplication is performed when this algorithm is executed. [1]

N;

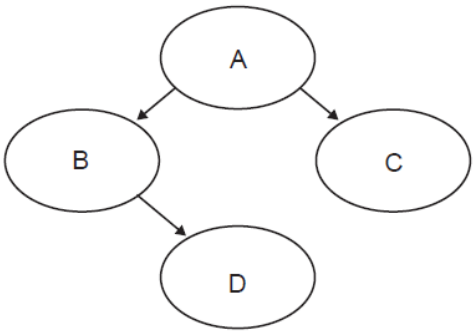
(b) Determine the value of *FUN(2,3)*, showing all of your working. [3]

```
FUN(2,3)=
=2 * FUN(2,2) ;
=2 * 2*FUN(2,1) ;
=2 * 2 * 2 * FUN(2,0);
=2*2*2*1= 8;
```

(c) State the purpose of this recursive algorithm. [1]

Calculates X^N;
Note: *DO NOT accept vague answers that may suggest the understanding of* N^X *or use incorrect terminology*

7. Consider the following binary tree.



(a) Identify all leaf nodes in this binary tree. [1]

D and C;

(b) For this binary tree, state the result of:

(i) inorder tree traversal, [1]

B D A C;

(ii) postorder tree traversal. [1]

D B C A;

Section B

8. A book shop has a computer at each point of sale, and also a central computer.

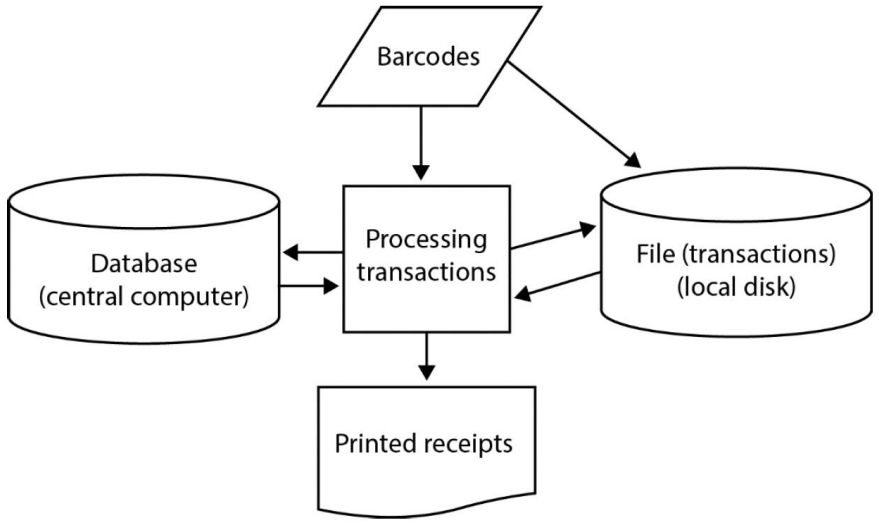
   When a customer buys a book in the book shop, the salesperson at the point of sale uses a scanning device to input a barcode from the book.

   The barcode is sent to the central computer where the barcode of each book and the corresponding price are held in a database on a disk.

   When the price is found, it is sent to the point of sale computer where all necessary calculations are performed, details of the transaction are stored on a local disk and a receipt is printed out.

   (a) Construct a system flow chart for the system described above. [5]

*Award **[1]** for all correct labels or symbols.*
(Accept meaningful, consistent symbols.)
*Award **[1]** for correct input flow.*
*Award **[1]** for correct output flow.*
*Award **[1]** for correct internal processes.*
*Award **[1]** for correct dataflow.*
*Answers given as a process flowchart may still be awarded **[1]** or **[2]**.*



At the point of sale there are peripheral devices other than the scanning device and printer.

(b) Outline the purpose of one other possible peripheral device in this scenario. [2]

*Award **[1]** for identifying a peripheral device, **[1]** for stating its purpose.*

Example answers:

Keyboard;
To type in some additional data;
Or to type in barcode data when it is not possible to scan;

Magnetic card reader;
Used when a credit card is used;

Microphone;
To call the next customer;
To call manager;

Monitor;
So the salesman can see the information/data on the screen;

Visual display;
So the customer can read the information/data on the display;

Speakers;
For customers to hear information;
For shop assistants to bring another item the customer may wish to buy;

The customers can also buy books online. A customer can select a book, and then enter their name, address and credit card number. This data is stored on the book shop's central computer in a database of customer orders.

(c) Outline the purpose of protocols in transferring this data. [2]

*Award up to **[2 max]**.*
Protocols are sets of rules for transmitting data correctly;
They ensure that data is sent from a customer's computer and received by the shop's computer;
To create a secure transmission of data from the client to the server through the use of the Hypertext Transfer Protocol (HTTPS) *ie* the customer can pay for the books securely (using TLS or SSL).

(d) (i) Identify two sources of risk to personal data in this online system. [2]

*Award up to **[2 max]**.*
Protocols are sets of rules for transmitting data correctly;
They ensure that data is sent from a customer's computer and received by the shop's computer;
To create a secure transmission of data from the client to the server through the use of the Hypertext Transfer Protocol (HTTPS) *ie* the customer can pay for the books securely (using TLS or SSL).

(ii) State two measures that the book shop can take to address the risks identified in part (d)(i). [2]

*Award up to **[2 max]**.*
All private information must be encrypted;
Transmission channel must be protected by encryption;
Logging on to the system must be secured (to prevent intruders);
Dual data back-up system in case of accidental deletion;

(iii) Outline the consequences to the customer if their data is not adequately protected. [2]

*Award up to **[2 max]**.*
Details stolen;
Used for fraudulent purposes;

Contact details could be shared;
Used for junk mail/fraud;

Personal details stolen;
For identity theft;

9. A new higher level programming language is being developed.

(a) Identify **two** reasons why consistent grammar and syntax should be essential features of a higher level programming language. [2]

*Award up to **[2 max]**.*
Easy to learn/use;
Otherwise time may be wasted learning the new language/writing programs in this HLL;
There will be no/less compilation errors;
There will be no/less logical errors;
(Reduction of time to create software;)
Future maintenance/development is possible by other programmers;

(b) Identify **two** features of a user interface that will allow application programmers to interact more easily with the programming language. [2]

*Award up to **[2 max]**.*
GUI;
Toolbars;
Menus;
Built in commands for inputting from touch screens;
Predicted text so that typing a class name followed by a full stop will bring up a list of methods/attributes;
Automatically use a colour to represent keywords/variables and improve readability

(c) State one method of providing user documentation. [1]

*Award **[1 max]**.*
Help files;
Online support;

Application programmers who use this programming language will be able to choose to use either an interpreter or a compiler.

(d) (i) Outline the need for an interpreter or a compiler. [2]

*Award up to **[2 max]**.*
Must be translated from a higher level language understandable by humans/not understood by machines;
Must be translated into machine code;
For the CPU to execute it;

(ii) Describe one advantage to application programmers of having both an interpreter and a compiler available. [2]

*Award up to **[2 max]**.*
Must be translated from a higher level language understandable by humans/not understood by machines;
Must be translated into machine code;
For the CPU to execute it;

One of the predefined sub-programs in the new language is *sumOdd()*. It accepts an integer *N* as input. If *N<=0* it outputs *-1*, otherwise it outputs the sum of the first *N* odd numbers.

For example:

sumOdd(4) outputs *16*, because 4 is not less than 0, and 1 + 3 + 5 + 7 = 16.

sumOdd(−3) outputs *−1*, because −3 is less than 0.

(e) Construct, in pseudocode, the algorithm for *sumOdd()*. [4]

Award marks as follows:
*Award **[1]** for branch of if-then-else leading to correct computation of S=–1;*
*Award **[1]** for the correct loop (boundaries);*
*Award **[1]** for correctly calculating the sum;*
*Award **[1]** for the output;*

Example algorithm 1:

```
if N<=0 then
     S=-1
else
     S=0
     loop for K=1 to N
          S=S+2*K-1
     endloop
end if
output S
```

Example algorithm 2:

```
if N>0 then
     S=0
     loop for K=1 to 2*N
          if K mod 2==1 then
            S=S+K
          end if
     endloop
else
     S=-1
end if
output S
```

(f) Outline the need for predefined sub-programs and collections. [2]

*Award **[1]** for any of the benefits listed below, **[1]** for an expansion (ie when/why/who will need it?).*

Reusability;
Modularity;
Reliability / All predefined sub-programs are tested and reliable;
etc.

Example answer:
Predefined sub-programs and collections are reusable;
And this reduces the cost/time needed to develop a large program;

10. The temperature, humidity, light levels and automatic watering of plants inside the greenhouses (glasshouses) of a garden centre are centrally monitored and controlled.

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

Data represented by a continuous variable;
***Note:** Do not accept "not in digital format" or just examples.*

(b) With reference to sensors, transducers and the processor, explain the control process that takes place in the greenhouse (glasshouses). [5]

*Award **[1]** for outlining the purpose of each device, for all three devices.*
*Award **[2]** for explaining the importance of feedback in this relationship;*

Example answer:
Sensor: converts an inputted physical quantity (temperature, light, *etc*) into an electrical signal;
Processor: executes a set of instructions (programs) which control the whole process;

Transducer: converts electrical signals into other forms of energy (heat, light, *etc*);

Feedback: input signals (information about what is happening to a particular process in the greenhouses) is monitored;
And fed back to the processor where they can be used to make decisions whether to change/modify the climate in the greenhouses or not;

(c) Outline the role of the operating system specific to this scenario. [4]

Operating system is a set of programs for this (dedicated) system;
Responsible for input devices (reading sensor data);
Responsible for sending to the output;
And reacting to inputted data in (predetermined) periods of time (to ensure the correct climate in the greenhouses);
***Note:** Correct answers must be specific to the scenario in question 10.*

(d) Describe the difference between polling and interrupt in the event that some of the sensors malfunction. [3]

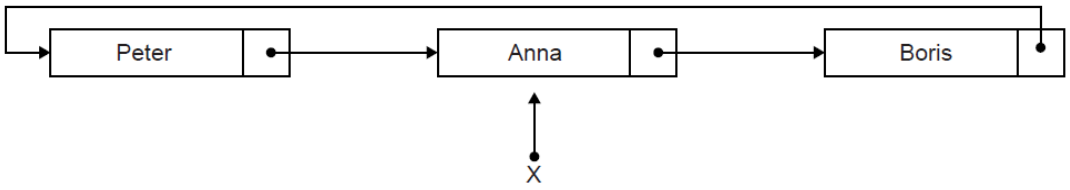
Example answer:
Polling:
The CPU visits/checks each sensor in turn to see if there is some input data;
It will know that the sensor has malfunctioned;

Interrupt:
Each sensor sends data as required;
It will not know that the sensor has malfunctioned (unless a timer is set with a limit on the time between expected interrupts by a given sensor);

(e) Compare a centrally controlled system with a distributed system. [2]

One computer/processor controls all the greenhouses;
Whilst in distributed system each of the greenhouses is monitored and controlled by its own computer;

11. The diagram shows a list of names held in a circular linked list. The end of the list is pointed to by an external pointer, X.



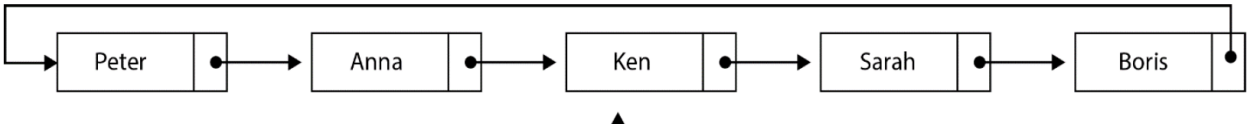
(a) State the first name in this circular list. [1]

Boris;

Two operations are performed on the list in the following order:
1. A node containing the name Sarah is inserted at the beginning of the list.
2. A node containing the name Ken is inserted at the end of the list.

(b) Sketch a diagram showing the resulting circular linked list. [3]

*Award up to **[3 max]**.*
For the diagram showing all nodes and links;
Ken inserted after Anna AND Sarah placed after Ken;
Node containing Ken is pointed to by X/Ken is currently at the end of the list;



(c) Describe how the number of names held in this list could be determined. [4]

Use a variable (counter) to keep track of/increment the number of nodes;
Use a temporary pointer;
Follow the pointers from the beginning of the list/from the node pointed to by pointer `X.next`;
Until the pointer to the end of the list (pointer X) is encountered;
***Note:** Accept methods that start from the end of the list (X).*

(d) Explain how a stack could be used to output, in reverse order, all names held in the linked list. [4]

Traverse the list from beginning to end;
Pushing each data value from the list onto the stack;
While stack is not empty;
Popping an element from the stack and output the stack element;

(e) Compare the use of static and dynamic data structures. [3]

Static data structure has a predetermined number of elements but number of elements in dynamic data structure does not have to be defined in advance;
Static data structure has limited size, the amount of memory available is the only limit in size of dynamic data structure, size varies;
In static data structure elements can be directly accessed, in a dynamic data structure access is sequential (which is slower);

12. A two-dimensional array, *A*, has *N* rows and *N* columns, where *N* is a positive integer.
The following algorithm is written to fill array *A* with the numbers 1, 2, 3,…, *N2*.

```
N=input('Enter an integer greater than zero')
K=N*N
loop for ROW=0 to N-1
  loop for COLUMN=0 to N-1
       A[ROW][COLUMN]=K
       K=K-1
    end loop
end loop
```

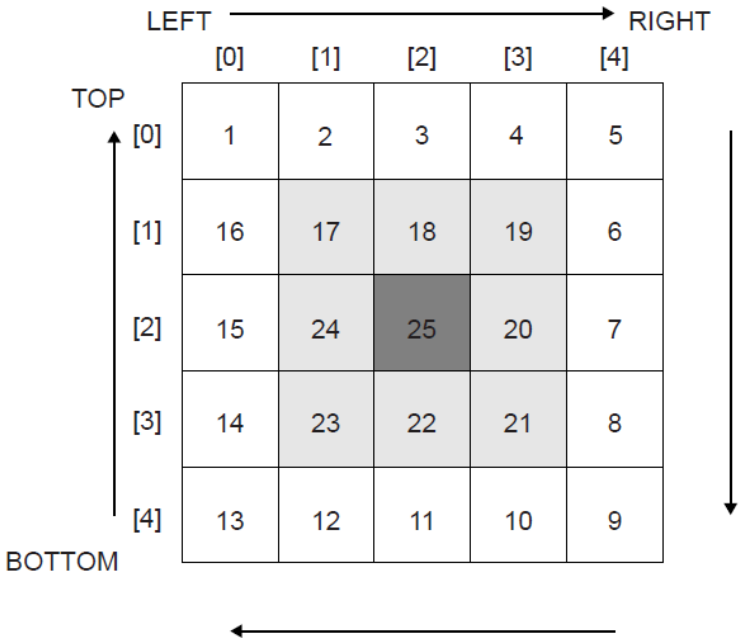
(a) Trace the algorithm, with an input of *N=3*, to show the contents of array *A* after the algorithm has been executed. [3]

*Award **[1]** for each correct row, up to **[3 max]**.*

	[0]	[1]	[2]
[0]	9	8	7
[1]	6	5	4
[2]	3	2	1

Accept answers that transpose the table.

There are many different ways of placing the numbers 1 to N^2 into an $N \times N$ two-dimensional array. The following two-dimensional array, with dimensions 5 × 5 has been filled in a circular (spiral) pattern with numbers 1 to 5^2.



The general process of filling an $N \times N$ two-dimensional array, in a circular (spiral) pattern, with numbers from 1 to N^2 could be described as follows:

- initialize *Z=1*,
- initialize *TOP*, *BOTTOM*, *LEFT* and *RIGHT*,
- iterate until the whole array is filled,
- each time *Z* is placed correctly increase the value of *Z* by 1,
- fill the elements of the *TOP* row starting from *LEFT* to *RIGHT*,
- increase *TOP* by 1 before filling the elements of the *RIGHT* column,
- fill the elements of the *RIGHT* column starting from *TOP* to *BOTTOM*,
- decrease *RIGHT* by 1 before filling the elements of the *BOTTOM* row,
- and continue filling the *BOTTOM* row and *LEFT* column in a similar way, adjusting *TOP*, *RIGHT*, *BOTTOM* and *LEFT* accordingly.

(b) (i) State the initial values for *TOP*, *BOTTOM*, *LEFT* and *RIGHT*. [1]

*Award **[1]** only for all correct values.*

```
TOP=0
BOTTOM=N-1
LEFT=0
RIGHT=N-1
```

(ii) State the consequence of not increasing *TOP* by 1 before starting to fill the *elements of the RIGHT column.* *[1]*

The array element at position `[TOP][RIGHT]` in which value of `Z` is already placed, will be overwritten by the value of `Z + 1`;
Not all of the numbers 1 to N^2 will be placed in the array because some will be overwritten;
The array will be filled with more than N^2 numbers/with numbers greater than N^2;

Accept answers from the sample 5×5 table, eg the value of `MATRIX[0][4]` which is already filled by 5, will be changed to 6.

(iii) In the algorithm described above, state the indices (subscripts) of the first and the last element to be filled in the *BOTTOM* row. [1]

The first element to be filled in `BOTTOM` row has indices (subscripts) `[BOTTOM][RIGHT]` and the last to be filled has indices (subscripts) `[BOTTOM][LEFT]`;

Accept answers from the sample 5 × 5 table. The first element to be filled in `BOTTOM` row has indices (subscripts) [4][3] and the last to be filled has indices (subscripts) [4][0].

(c) Construct, in pseudocode, an algorithm to fill an $N \times N$ two-dimensional array, in a circular (spiral) pattern, with numbers from 1 to N^2 as described above. [9]

*Award up to **[9 max]** as follows.*
*Award **[1]** for initializing `Z`.*
*Award **[1]** for initialization of the top and bottom rows, and left and right columns.*
*Award **[1]** for the outer loop (must be `while`).*
*Award **[1]** for the idea that four inner loops are needed (could be `for` or `while` loops).*
*Award **[1]** for each correct inner loop up to **[4 max]**.*
*Award **[1]** for assignment (current value of `Z` placed in `A`).*
*Award **[1]** for changing the value of `Z` after each assignment.*
*Award **[1]** for changing values of `TOP, BOTTOM, LEFT, RIGHT`.*

Example answer 1:

```
Z=1
TOP=0
BOTTOM=N-1
RIGHT=N-1
LEFT=0
loop while Z<=N*N
  COUNT1 = LEFT
  loop while COUNT1 <= RIGHT
    A[TOP][ COUNT1] = Z
    Z = Z+1
    COUNT1 = COUNT1+1
  end loop
  TOP = TOP+1
  COUNT2 = TOP
  loop while COUNT2 <= BOTTOM
    A[COUNT2][ RIGHT] = Z
    Z = Z+1
    COUNT2 = COUNT2+1
  end loop
  RIGHT = RIGHT-1
  COUNT3 = RIGHT
  loop while COUNT3 >= LEFT
    A[BOTTOM][ COUNT3] = Z
    Z = Z+1
    COUNT3 = COUNT3-1
  end loop
  BOTTOM = BOTTOM-1
  COUNT4 = BOTTOM
  loop while COUNT4 >= TOP
    A[COUNT4][ LEFT] = Z
    Z = Z+1
    COUNT4 = COUNT4-1
  end loop
  LEFT = LEFT+1
end loop WHILE
```

Example answer 2:

```
Z=1
TOP=0
BOTTOM=N-1
RIGHT=N-1
LEFT=0
loop while Z<=N*N
  loop for i from LEFT to RIGHT
    A[TOP][i]=Z
    Z=Z+1
  end loop
  TOP=TOP+1
  loop for i from TOP to BOTTOM
    A[i][RIGHT]=Z
    Z=Z+1
  end loop
  RIGHT=RIGHT-1
  loop for i from RIGHT downto LEFT
    A[BOTTOM][i]=Z
    Z=Z+1
  end loop
  BOTTOM=BOTTOM-1
  loop for i from BOTTOM downto TOP
    A[i][LEFT]=Z
    Z=Z+1
  end loop
  LEFT=LEFT+1
end loop WHILE
```

Note: *For both examples, assume that integer* `N` *is inputted and the space for the array* `A` *with dimensions* $N \times N$ *is allocated.*