

## **INFORMATION AND COMMUNICATION TECHNOLOGY (ELECTIVE) 2**

### **1. GENERAL COMMENTS**

This paper is the second May/June ICT administered. The standard of the paper compared favourably with the maiden paper in the areas of content and level of difficulty.

Although the paper was supposed to have been well within reach of the candidates, the general performance happened to be just slightly better than the first.

On the whole, the performance was just average.

### **2. A SUMMARY OF CANDIDATES' STRENGTHS**

- (1) Most candidates responded to the question as demanded by the rubrics.
- (2) A few candidates exhibited good knowledge of the subject matter.
- (3) Some candidates expressed themselves very well in the English Language.
- (4) Candidates showed delve knowledge in the impact of Information Technology in everyday life.

### **3. A SUMMARY OF CANDIDATES' WEAKNESSES**

The candidates' weaknesses identified include:

- (1) Inability to appreciate the key requirements of the questions.
- (2) Inadequate preparations.
- (3) Poor communication skills.
- (4) Some of the candidates had bad handwriting.
- (5) Some candidates demonstrated in their answers that they have *little or no knowledge* of the examination syllabus.

### **4. SUGGESTED REMEDIES**

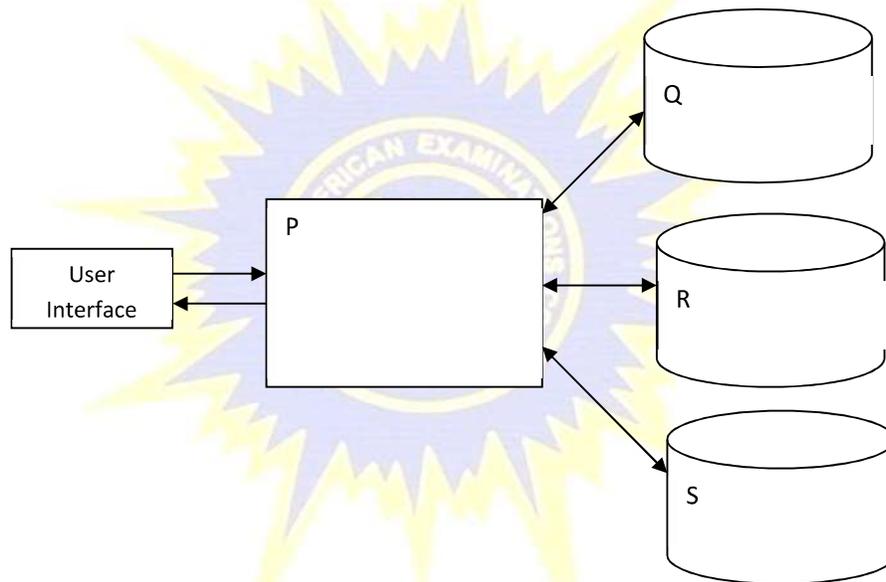
- (1) Candidates should carefully read through the questions, selecting those to be attempted and planning the answers before writing them out.
- (2) Candidates should learn with suitable textbooks and material on ICT and careful use of the Internet as a learning tool.
- (3) Candidates should avoid the use of communication styles on the various social media platforms.

- (4) Candidates should have time to read through their answers to correct any errors as well as add further details.
- (5) Candidates should be reprimanded for bad handwriting, to enable them improve on their handwriting.
- (6) Teachers using the ICT Elective syllabus should make it their first point of teaching in the schools.

5. **DETAILED COMMENTS**

**QUESTION 1**

- (a) What is meant by the term information system?
- (b) List three types of information systems.
- (c) The diagram below represents a decision support system. Identify the components labeled P, Q, R and S.



Majority of candidates did not provide correct answers to this question. Suggested solution is as follows:

- (a) An Information System is any formalized set of procedures that is capable of accepting data from any source, processing the data and making the results available to users.
- (b)
  - Transaction Processing Systems / Operation Support System
  - Office Automation Systems
  - Knowledge Work Systems
  - Decision Support Systems

- Management Information Systems / Management Support System.
- Executive Support Systems

(c) P – Planning Language/Dialogue System

Q – Corporate Database

R – User Database

S – Model Database

### QUESTION 2

Information Technology continuous to impact on everyday life. State two benefits of each of the following:

- (a) e-health;
- (b) e-learning;
- (c) e-banking;
- (d) telecommuting;
- (e) telesales.

Almost all candidates who attempted this question got full marks. Performance was excellent.

### QUESTION 3

- (a) List two steps of the machine execution cycle in the processing of data.
- (b) Explain two of the steps listed in 3(a).
- (c) Differentiate between
  - (i) device driver and language translator;
  - (ii) word size and bus size.

This is the most unpopular question among candidates. The few who attempted had very low marks. Suggested solution is as follows:

- (a) -Fetching
- Decoding
- Executing
- Storing
- (b) **Fetching**

This is the process of obtaining a program instruction or data item from memory.

### **Decoding**

This is the process of translating the instruction into commands that the computer understands.

### **Executing**

This is the process of carrying out the commands and executing.

### **Storing**

This is the process of writing the results to memory or register.

- (c) (i) A device driver is software that allows high-level computer programs to enable hardware devices function appropriately, while a language translator is software that converts programmers' source code into object code.
- (ii) A word size of a computer is the number of bits that the CPU can process simultaneously, while the bus size determines how many bits are transmitted at a time.

### **QUESTION 4**

- (a) **State one function each of the following computer network devices:**
- (i) **bridge;**
- (ii) **repeater;**
- (iii) **router.**
- (b) **Name the layer of the OSI model at which each of the devices listed in 4(a) operates.**
- (c) **List the two types of network transmission media.**

The performance of candidates' on (a) and (c) was average but could not provide correct answer to the (b) part. The expected answer is:

- (a) (i) **Bridge:**
- They connect networks
  - Used in dividing network segments
  - Used to increase performance on a high-traffic segment.
- (ii) **Repeater:**
- They boost up signals to their original strength
  - They retime and retransmit signals to another segment.
  - They essentially enable a number of cables segments to be treated as a single cable.
- (iii) **Router:**

- They connect multiple networks.
- They determine the best path for sending data.
- They filter signals
- They repeat signals

(b) **Bridge-** They operate in the data link layer of the OSI model.

**Repeater-** They operate at the physical layer of the OSI model.

**Router-** They operate at the network layer of the OSI model.

- (c) - Wireless  
- Wire/Cable

### QUESTION 5

**Construct a hexadecimal addition table.**

Most candidates attempted this question but was the worst answered without any correct table.

Candidates should add in hexadecimal and fill the various cells, noting that in hex, the decimals 10, 11, 12, 13, 14 and 15 are denoted by A, B, C, D, E and F respectively.

### INFORMATION AND COMMUNICATION TECHNOLOGY (ELECTIVE) 3

#### **1. GENERAL COMMENTS**

The standard of the paper and that of the previous year's examination is the same. It was noted that, candidates' performed very much better than the previous year.

It has, however, been observed that performances were localized, i.e. excellent performances are concentrated at schools while bad performances are also concentrated at certain schools.

The variance of performances at localities is insignificant. It is either entirely good or entirely bad.

Candidates' general performance was good.

#### **2. A SUMMARY OF CANDIDATES' STRENGTHS**

- (1) Majority of the candidates were able to present CDs with data.

- (2) Most candidates responded to the demands of the questions as demanded by the rubrics.
- (3) Most of the candidates were able to code in HTML.
- (4) Candidates were able to enter data.

### **3. A SUMMARY OF CANDIDATES' WEAKNESSES**

1. A number of candidates used Microsoft Word for the database application.
2. Candidates did not name objects properly.
3. Usage of the header facility in the table was wrongly effected.
4. Candidates were unable to create the database relationships.
5. Candidates were unable to produce the database relationships reports.
6. Candidates shied away from programming aspects – especially QBASIC.

### **4. SUGGESTED REMEDIES**

1. Teachers should try and start teaching ICT elective from programming perspective. Candidates are more likely to be able to learn the use of application better than they can do in learning programming on their own.
2. Teachers must cultivate logical reasoning skills in candidates to help in the development of programming skills.
3. Teachers must pay attention to the curriculum requirements and should not underrate the expectations of the ICT curriculum. They must stress on technical approach in teaching ICT.
4. Candidates must be encouraged and assisted to pick up personal ICT projects structured in a manner which will compel them to eventually be practical in their approach to the subject.

### **5. DETAILED COMMENTS**

#### **QUESTION 1**

##### **HTML**

The question required candidates to create an html web page. It required the use of a *Text Editor*. This time round, candidates used text editors in the coding HTML.

The solution to the question is expected to follow the pattern explained below.

HTML is a standard and the layout follows a specific structure to allow for correct interpretation for presentation. The structure of an html document is as follows:

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>
</html>
```

It must be noted that the title is part of the head tag. Placing it outside the head is not a correct structuring even though you can have the title correctly displayed.

The body tag is not part of head tag as some have sort to do. Even though an example of indentation was given in the question, candidates still failed to indent properly.

Indentation is not considered critical for the structure tags i.e. html, head, title, and body. Rather, it is the lines coded between the opening and closing sets of any of the tags that are critical.

The arrangement :

```
<p>
    <u>Items</u>
</p>
```

has been given in the question as *an example*. Yet candidates did not perform indentation. `<p>....</p>` is a set of paragraph tags. Within this paragraph a content of the paragraph is entered as *Items*. This content is underlined using the `<u>....</u>` set of tags.

All that candidates should have done is to copy this example given and change the tags to suit the exact work required of them.

A sample answer is as shown below:

```
<!DOCTYPE html>
<html>
<head>
<title>
```

Candidates' name and Index Number goes here

```
</title>
```

```
</head>
```

```
<body>
```

```
<p>
```

These are my to-do list in no particular order:

```
</p>
```

```
<!-- List the items using Unordered HTML Lists as implied in the  
line above. -->
```

```
<ul>
```

```
<li>Do assignment</li>
```

```
<li>Wash</li>
```

```
<li>Watch a movie</li>
```

```
</ul>
```

```
</body>
```

```
</html>
```

Performance of most candidates' was good.

## QUESTION 2

### **DATABASE**

The requirement is for candidates to use a database application to design a database for a distribution company and name it ***ClientOrders*** in the folder they will create in drive C:\.

The very exact naming of the database is critical. Its placement in the folder created is critical. You can manually search for a document on the computer through various techniques even if you forgot the name. However, during the execution of a program, the name and its location must be ***exact*** otherwise the program cannot find it.

Some candidates used Microsoft Word to answer this question. Microsoft Word is not a database application.

Three tables were required to be created defining the fields appropriately:- tblClients, tblOrders, tblStock.

Defining the fields appropriately implies that the field names must be correct and their data types must be correct. A table with a wrong field data types is not a correct table.

Some candidates did not name the tables properly. Others defined all fields as type *Text*. This is not correct in some instances.

The created tables are to be used to create the relationships to enable a user to design a form to show *Client Name, Location, Telephone, Item Order Date*.

To finally link the relationships, select the primary key from one table and drag it to the same field in another table. For the relationship between *tblClients* and *tblOrders*, select the primary key *ClientID* from table *tblClient* and drag it to *ClientID* in the *tblOrders* table. Also establish relationship between *tblOrders* and *tblStock* using the same procedure.

To finally link the relationships, select the primary key from one table and drag it to the same field in another table. For the relationship between *tblStock* and *tblOrders*, select the primary key *ItemID* from table *tblStock* and drag it to *ItemID* in the *tblOrders* table.

Select the form field from the tables that contain them.

As you change the table selection in the *Table/Queries* tab the *Available Fileds* also will change accordingly. From the fields that display, select the field you want from it. Continue to change the tables and select desired fields until all fields for the form as display in the *Selected Fields* tab.

The next stage is the insertion of full name and system date into the footer. This is system date and not date manually entered by candidate.

Create space for the form footer section either by dragging the borders or setting height option in the *FormFooter* toolbar on the right side of the screen.

Now insert a label or labels by selecting *Ab* from the Design Menu bar and opening up a sized label in the footer for inserting name. You may clean off the entry in the form header since it is not required.

Edit the labels of the form to read exactly what is in the question. Finish by saving your work. The name must be *frmOrders*.

A relationship report is required to be produced and saved as *R\_report* in the database. This is achieved by selecting from the Database tools menu *Relationships* and then select *Relationship Reports*.

Performance was average.

### QUESTION 3

#### **QBASIC**

Candidates were not able to carry out this work except for an exceptional few. The following is a basic outline pseudo code with some sample codes added to assist test coding for solving the problem.

```
DECLARE VARIABLES  
DIM Number AS INTEGER
```

```
DIM Count AS INTEGER
```

```
DIM Counter AS INTEGER
```

```
DECLARE A TEMPORAL STORE OF DATA e.g. ARRAY
```

**Note:** In the formatting of the output, it must be noted that the first column is the original data while the second column is the sorted data. Hence both the sorted data and the unsorted data must be available at output time.

Thus, the declared variables must include at two arrays, one for the raw data and one for the sorted data. The two of them must hold the same information before sorting starts. Let us call these two arrays as UNSORTED\_ARRAY\_LIST and SORTED\_ARRAY\_LIST for this work.

```
DIM UNSORTED_ARRAY_LIST(6) AS INTEGER
```

```
DIM SORTED_ARRAY_LIST(6) AS INTEGER
```

There are six (6) known items to be sorted and therefore we input data six times only:

```
Number=6
```

```
CLEAR SCREEN
```

```
CLS
```

```
START LOOP TO READ INPUTS VARIABLES FOR NUMBER OF RECORDS GIVEN  
FOR Count = 1 TO Number
```

```
INPUT A RECORD INTO A TEMPORAL STORE CALLED tmp  
Input "Enter a Number";tmp
```

```
STORE RECORD IN UNSORTED_ARRAY_LIST
```

```
UNSORTED_ARRAY_LIST(Count)= tmp
```

```
STORE RECORD IN SORTED_ARRAY_LIST
```

```
SORTED_ARRAY_LIST(Count)=tmp
```

```
LOOP UNTIL LAST RECORD IS READ
  Next Count
```

At the end of the above loop, UNSORTED\_ARRAY\_LIST and SORTED\_ARRAY\_LIST will both contain the same values.

**NOTE:** It is possible to do this in many other ways. Bear in mind that simply saying UNSORTED\_ARRAY\_LIST = SORTED\_ARRAY\_LIST as a way of putting the values of one array into the other has its implications that has to be properly resolved, else depending on the environment you may end up having them always containing the same values throughout the program execution.

This is not what is wanted in this instance. We want to be able to change the values in one array without affecting the values in the second array. Thus, we store values in them separately.

#### **PERFORM SORTING**

To perform the sorting on the SORTED\_ARRAY\_LIST and leave the UNSORTED\_ARRAY\_LIST untouched!

#### **Using Bubble sort algorithm in its beginner form, sort *only* SORTED\_ARRAY\_LIST**

```
FOR Count = 1 TO Number
  FOR Counter = 1 TO Number
    IF SORTED_ARRAY_LIST (Counter) > SORTED_ARRAY_LIST
      (Count) THEN SWAP SORTED_ARRAY_LIST (Count), SORTED_ARRAY_LIST
      (Counter)
    NEXT Counter
  NEXT Count
```

#### **‘OUTPUT HEADER TO SCREEN**

```
PRINT
```

```
PRINT
```

```
PRINT “NUMBER”, “SORTED”
```

```
PRINT “*****”, “*****”
```

```
PRINT
```

The comma (,) provides the spacing. Using semi colon (;) will make the “NUMBER” and “SORTED” to be joined together leaving no space.

FORMAT *OUTPUTDATA* and OUTPUT *OUTPUTDATA* TO SCREEN Row by Row.

Start a LOOP to output the 6 lines

```
FOR n = 1 TO 6
```

## FORMAT OUTPUTDATA

Note: In formatting the output, the values on the rows for the left NUMBER column will come from the UNSORTED\_ARRAY\_LIST while the values for the right SORTED column will come from the SORTED\_ARRAY\_LIST

The values will be matched one-to-one. That is, the  $n^{\text{th}}$  value of UNSORTED\_ARRAY\_LIST will be outputted together with the  $n^{\text{th}}$  value of the SORTED\_ARRAY\_LIST.

## OUTPUT OUTPUTDATA TO SCREEN

The following QBasic statement formats will output the result all together. The *PRINT* is for output while *UNSORTED\_ARRAY\_LIST(n)*, *SORTED\_ARRAY\_LIST(n)* is the formatting.

```
PRINT UNSORTED_ARRAY_LIST(n), SORTED_ARRAY_LIST(n);
```

The *print* command sends the output to screen. The *UNSORTED\_ARRAY\_LIST(n)*, *SORTED\_ARRAY\_LIST(n)* is what goes out together with its formatting. *UNSORTED\_ARRAY\_LIST(n)* is the  $n^{\text{th}}$  value for the left side under the column heading "NUMBER" the comma (,) formats the space between the columns. *SORTED\_ARRAY\_LIST(n)* is the matching  $n^{\text{th}}$  value to the right under column named "SORTED".

Continue if it is not yet done 6 times

```
    NEXT n
```

```
END
```

```
CLEAR SCREEN
```

```
CLS
```

```
EXIT
```