THE UNITED REPUBLIC OF TANZANIA

NATIONAL EXAMINATIONS COUNCIL OF TANZANIA

ADVANCED CERTIFICATE OF SECONDARY EDUCATION EXAMINATION

136/2 **COMPUTER SCIENCE 2**

(For Both School and Private Candidates)

Time: 3 Hours **ANSWERS** **Year: 2016**

Instructions:

1. this paper consists of three questions.
2. Answer two questions including question number one
3. Submit printed codes and screenshots together with the softcopy of your work(s)

1. (a) Construct a program in C++ that prints a 'pyramid' shape of a specified height on the screen. The pyramid shape should be made of "*" characters. A program should prompt a user to enter the height of the pyramid. The height must be between 1 and 30.

Solution:

```
#include <iostream>
using namespace std;

int main() {
    int height;

    cout << "Enter the height of the pyramid (1 to 30): ";
    cin >> height;

    if (height < 1 || height > 30) {
        cout << "Invalid height! Please enter a value between 1 and 30." << endl;
        return 1;
    }

    for (int i = 1; i <= height; i++) {
        for (int j = 1; j <= height - i; j++) {
            cout << " ";
        }
        for (int k = 1; k <= 2 * i - 1; k++) {
            cout << "*";
        }
        cout << endl;
    }

    return 0;
}
```

This program creates a pyramid pattern by printing spaces and asterisks for the specified height. For example, a height of 5 will produce a pyramid with 5 levels.

(b) Generate a program in C++ that reads two integers from the user. A program must find the Greatest Common Factor of entered integers. Print with a suitable message.

Solution:

```
#include <iostream>
using namespace std;
```

```
int findGCD(int a, int b) {
    while (b != 0) {
        int temp = b;
        b = a % b;
        a = temp;
    }
    return a;
}

int main() {
    int num1, num2;

    cout << "Enter the first number: ";
    cin >> num1;
    cout << "Enter the second number: ";
    cin >> num2;

    int gcd = findGCD(num1, num2);

    cout << "The Greatest Common Factor (GCF) of " << num1 << " and " << num2 << " is: " << gcd << endl;

    return 0;
}
```

2. (a) Use HTML codes to construct the student identity card as shown below:

Page descriptions:
- The first table border =1, cellspacing="0", cellpadding="0", width="600", height="300".
- The second table border=0, cellspacing="0", cellpadding="0" width="100%", height="100%".
- Borders color=blue.
- Font color of the statement "MINISTRY OF EDUCATION" should be red and level of heading must be three (h3).
- Font color of the statement "STUDENT IDENTITY CARD" should be black and level of heading must be four (h4).
- Font color of school name and address/contact should be blue. Font size of school name and contact is (h4) and (h5) respectively.
- Insert any image of your choice from the pictures library.
- Font color of italicized statement must be red.
- Replace Syyy/xxx with your Examination Number.

Solution:

~~~
```html
<!DOCTYPE html>
<html>
<head>
    <title>Student Identity Card</title>
    <style>
        table {
            border-collapse: collapse;
            width: 600px;
            height: 300px;
            margin: 0 auto;
            border: 1px solid blue;
        }
        td {
            text-align: center;
            vertical-align: middle;
        }
        .header h3 {
            color: red;
        }
        .subheader h4 {
            color: black;
        }
        .school-details h4, .school-details h5 {
            color: blue;
        }
        .italic {
            color: red;
            font-style: italic;
        }
    </style>
</head>
<body>
    <table>
        <tr>
            <td class="header" colspan="2">
                <h3>MINISTRY OF EDUCATION</h3>
                <h4>STUDENT IDENTITY CARD</h4>
            </td>
        </tr>
        <tr>
            <td colspan="2" class="school-details">
                <h4>DODOMA SECONDARY SCHOOL</h4>
~~~

```
        <h5>P.O.BOX 200, DODOMA-TANZANIA</h5>
        <h5>Tel: +25512525880</h5>
      </td>
    </tr>
    <tr>
      <td>
        <p>Form: Five 2012/13</p>
        <p>Six 2014</p>
        <p>Exam No: Syyy/xxx</p>
      </td>
      <td>
        <img src="your_image.jpg" alt="Student Image" width="100" height="100">
      </td>
    </tr>
    <tr>
      <td colspan="2" class="italic">
        <p>If found please return to the above address</p>
      </td>
    </tr>
  </table>
</body>
</html>
```

2. (b) Create a JavaScript page which can load a prompt box that asks a user to enter a number. A script should provide a message through an alert box to inform a user the status of the number entered. The script must test whether the input entered is a number, positive, or negative. Provide appropriate message for each category of possible inputs.

Note
- The heading of your page should be "My first JavaScript page webpage" with blue color and aligned at center.
- Choose any suitable background color for your page.

Solution:

```
<!DOCTYPE html>
<html>
<head>
  <title>My First JavaScript Page</title>
  <style>
    body {
      background-color: lightgray;
```

```
            text-align: center;
            font-family: Arial, sans-serif;
        }
        h1 {
            color: blue;
        }
    </style>
    <script>
        function checkNumber() {
            let input = prompt("Enter a number:");

            if (input === null || input === "") {
                alert("No input provided!");
                return;
            }

            if (isNaN(input)) {
                alert("The input is not a number!");
            } else {
                let number = parseFloat(input);
                if (number > 0) {
                    alert("The number is positive.");
                } else if (number < 0) {
                    alert("The number is negative.");
                } else {
                    alert("The number is zero.");
                }
            }
        }
    </script>
</head>
<body>
    <h1>My First JavaScript Page Webpage</h1>
    <button onclick="checkNumber()">Click to Enter a Number</button>
</body>
</html>
```

Explanation:

1. The page has a heading "My first JavaScript page webpage" styled in blue and centered.
2. A button is provided that triggers the JavaScript function `checkNumber()`.
3. The function uses `prompt()` to get user input and validates it:
   - Alerts if the input is not a number.
   - Checks if the number is positive, negative, or zero and provides an appropriate alert message.

3. (a) (i) Use Microsoft Access to create a database called DBOS. Design a table named “Student” which consists of the following fields: Student ID, Student Name, Age, Mark1, Mark2, Mark3, Total Marks, Average, and Grade.

(ii) Construct a form named “StudentForm”: Make your form active so that Total Marks, Average, and Grade can be calculated automatically on it. Use the following conditions to assign grades:
- If average > 80, "A"
- If average > 60, "B"
- If average > 40, "C"
- If average > 20, "D"
- If average <= 20, "F"

(iii) Add ten records to the table through the form created in (ii) to test your system.

Table Descriptions:
- The highest marks entered must be 100. If a user enters marks greater than 100, a message "Sorry the highest mark is 100" should be displayed.
- Student age should be greater than 7 and less than 20. A system should display a message "The entered age is not allowed" when a user enters the age below or above the limit.

Form Descriptions:
- The form background color should be dark blue.
- Font type of fields should be Bodoni MT Black.

Solution:
1. Create the "Student" Table in Microsoft Access:
  - Fields:
    - Student ID: Text (Primary Key)
    - Student Name: Text
    - Age: Number
    - Mark1: Number
    - Mark2: Number
    - Mark3: Number
    - Total Marks: Calculated ([Mark1] + [Mark2] + [Mark3])
    - Average: Calculated ([Total Marks] / 3)
    - Grade: Text (Based on conditions below)
  - Validation Rules:
    - Mark1, Mark2, Mark3: `<=100`
    - Age: `>=7 AND <=20`
  - Validation Text:
    - Marks: "Sorry the highest mark is 100."
    - Age: "The entered age is not allowed."

2. Create the "StudentForm":
   - Add fields to the form and format them as follows:
     - Background color: Dark Blue
     - Font type: Bodoni MT Black
   - Add a button to save records and a validation script in the form design:
     - If marks exceed 100 or age is out of range, display the appropriate error message.

3. Add Automatic Calculations for Total Marks, Average, and Grade:
   - Set "Total Marks" and "Average" as calculated fields in the table.
   - Use an expression builder in the form to assign grades dynamically:

```
IIf([Average]>80,"A", IIf([Average]>60,"B", IIf([Average]>40,"C", IIf([Average]>20,"D","F"))))
```

4. Test the Form:
   - Input ten sample records with varying ages and marks. Ensure validation messages appear for invalid data.

3. (b) Use Visual Basic program to construct the interface below:

Form Descriptions:
- The background color of the form should be `&H00C0FFC0&`.
- File menu consisting of Open, Save as, and Exit items with shortcuts Ctrl + O, Ctrl + S, and F1 respectively.
- Edit menu consisting of Paste, Cut, and Copy items.
- Format menu consisting of Background color, Font, and Bold items.
- Use Microsoft FlexGrid control to create East Africa minerals.

Solution:

```
Private Sub Form_Load()
    ' Set background color
    Me.BackColor = &H00C0FFC0&

    ' Add File menu
    Dim FileMenu As Menu
    Set FileMenu = Me.Controls.Add("VB.Menu", "File")
    FileMenu.Caption = "&File"
    FileMenu.MenuItems.Add "&Open", "Ctrl+O"
    FileMenu.MenuItems.Add "&Save As", "Ctrl+S"
    FileMenu.MenuItems.Add "E&xit", "F1"

    ' Add Edit menu
    Dim EditMenu As Menu
    Set EditMenu = Me.Controls.Add("VB.Menu", "Edit")
    EditMenu.Caption = "&Edit"
    EditMenu.MenuItems.Add "&Cut", ""
    EditMenu.MenuItems.Add "&Copy", ""
    EditMenu.MenuItems.Add "&Paste", ""

    ' Add Format menu
    Dim FormatMenu As Menu
    Set FormatMenu = Me.Controls.Add("VB.Menu", "Format")
    FormatMenu.Caption = "&Format"
    FormatMenu.MenuItems.Add "Background &Color", ""
    FormatMenu.MenuItems.Add "&Font", ""
    FormatMenu.MenuItems.Add "&Bold", ""
    ' Add FlexGrid control
    Dim FlexGrid As MSFlexGrid
    Set FlexGrid = Me.Controls.Add("MSFlexGridLib.MSFlexGrid", "FlexGrid")
    FlexGrid.Left = 100
    FlexGrid.Top = 100
    FlexGrid.Width = Me.ScaleWidth - 200
    FlexGrid.Height = Me.ScaleHeight - 200
    FlexGrid.Cols = 4
    FlexGrid.Rows = 5
    FlexGrid.TextMatrix(0, 0) = "Country"
    FlexGrid.TextMatrix(0, 1) = "Athletes"
    FlexGrid.TextMatrix(0, 2) = "Gold"
    FlexGrid.TextMatrix(0, 3) = "Silver"
    FlexGrid.TextMatrix(0, 4) = "Bronze"
End Sub
```