



UNIVERSITI TUN HUSSEIN ONN MALAYSIA

**FINAL EXAMINATION
SEMESTER I
SESSION 2018/2019**

COURSE NAME : DOT NET PROGRAMMING
COURSE CODE : BIE 33103
PROGRAMME CODE : BIP / BIW
EXAMINATION DATE : DECEMBER 2018 / JANUARY 2019
DURATION : 3 HOURS
INSTRUCTION : ANSWER ALL QUESTIONS

THIS QUESTION PAPER CONSISTS OF SEVEN (7) PAGES

SECTION A

Determine whether each of these statements is **TRUE** or **FALSE**.

- Q1** ASP.NET is a sub of .NET framework. (1 mark)
- Q2** We can split between all markups and the codes using code behind method. (1 mark)
- Q3** `<%@ page language="C#" %>` inside an .aspx file is an example of inline code within a code render block. (1 mark)
- Q4** `Response.Redirect` can be used to display output to users. (1 mark)
- Q5** Encapsulation is an approach to display necessary methods or properties and hiding important methods or properties. (1 mark)
- Q6** Content page helps the website to have a consistent look and behavior across all the pages in the web application. (1 mark)
- Q7** In C#, `SqlDataSource` class represents an SQL database to data-bound controls. (1 mark)
- Q8** `ViewState` mode is available for both ASP.net and HTMLcontrol. (1 mark)
- Q9** In a web application, there are two level events, page and control level. (1 mark)



Q10 XML is the only format that .NET framework supports for serialization. (1 mark)

SECTION B

Answer **ALL** questions.

Q11 (a) Describe the execution of client side source code in ASP.NET. (3 marks)

(b) Write a C# code segment to print the output as shown in **Figure Q11(b)** using `while` statement.

```
I am excellent!  
I am excellent!  
I am excellent!
```

Figure Q11(b) (2 marks)

(c) Write a C# code segment to check whether a number is even or odd using `if-else` statement. (5 marks)

Q12 (a) Differentiate between `Int32.Parse (string s)` and `Convert.ToInt32 (string s)` methods. (4 marks)

(b) Figure **Q12(b)** shows a console application written in C#.

(i) Explain what the codes would like to achieve. (4 marks)

(ii) Write sample output using your own input. (2 marks)



```

using System;

class ProgramQ12b
{
    static void Main(string[] args)
    {
        Console.WriteLine(" Input any positive number : ");
        int num = Convert.ToInt32(Console.ReadLine());
        long f = Calc(num);
        Console.WriteLine(" The output of calculation of {0}
is : {1} ", num, f);
        Console.ReadKey();
    }

    private static long Calc(int num)
    {
        if (num == 0)
        {
            return 1;
        }
        return num * Calc (num-1);
    }
}
    
```

Figure Q12(b)

Q13 Based on Figure Q13,

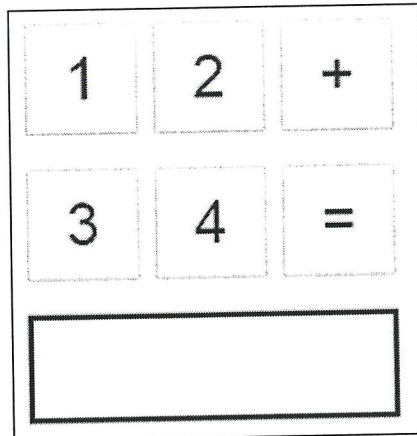


Figure Q13

- (a) Write <body></body> code segment for the .aspx file to generate the interface. Assume the size of the button “1” is 60px × 60px. (13 marks)
- (b) Assume that ID for button “1” is Button1, while ID for the Textbox is t. Write .aspx.cs file code segment for Button1_Click(). (7 marks)



Q14 Based on **Figure Q14**,

- (a) Produce a form in design view. (14 marks)

- (b) Explain validation control used. (12 marks)

- (c) Write `ValidationSummary` validation control for the codes. (4 marks)

- END OF QUESTION -



FINAL EXAMINATION

SEMESTER / SESSION : SEM I 2018/2019
 COURSE NAME : DOTNET PROGRAMMING

PROGRAMME CODE : BIP
 COURSE CODE : BIE 33103

```
<form id="form1" runat="server">

    <table style="width: 66%;">

        <tr>
            <td class="style1" colspan="3" align="center">
                <asp:Label ID="lblmsg"
                    Text="President Election Form : Choose your president"
                    runat="server" />
            </td>
        </tr>

        <tr>
            <td class="style3">
                Candidate:
            </td>

            <td class="style2">
                <asp:DropDownList ID="ddlcandidate" runat="server"
                    style="width:239px">
                    <asp:ListItem>Please Choose a Candidate</asp:ListItem>
                    <asp:ListItem>Adam</asp:ListItem>
                    <asp:ListItem>Scott</asp:ListItem>
                    <asp:ListItem>Siva</asp:ListItem>
                    <asp:ListItem>Johan</asp:ListItem>
                </asp:DropDownList>
            </td>

            <td>
                <asp:RequiredFieldValidator ID="rfvcandidate"
                    runat="server" ControlToValidate ="ddlcandidate"
                    ErrorMessage="Please choose a candidate"
                    InitialValue="Please choose a candidate">
                </asp:RequiredFieldValidator>
            </td>
        </tr>

        <tr>
            <td class="style3">
                House:
            </td>

            <td class="style2">
                <asp:RadioButtonList ID="rblhouse" runat="server"
                    RepeatLayout="Flow">
                    <asp:ListItem>Red</asp:ListItem>
                    <asp:ListItem>Blue</asp:ListItem>
                    <asp:ListItem>Yellow</asp:ListItem>
                    <asp:ListItem>Green</asp:ListItem>
                </asp:RadioButtonList>
            </td>

            <td>
                <asp:RequiredFieldValidator ID="rfvhouse" runat="server"
                    ControlToValidate="rblhouse" ErrorMessage="Enter your house">
            </td>
        </tr>
    </table>
</form>
```



```

name" >
    </asp:RequiredFieldValidator>
    <br />
</td>
</tr>

<tr>
    <td class="style3">
        Class:
    </td>

    <td class="style2">
        <asp:TextBox ID="txtclass" runat="server"></asp:TextBox>
    </td>

    <td>
        <asp:RangeValidator ID="rvclass"
            runat="server" ControlToValidate="txtclass"
            ErrorMessage="Enter your class (6 - 12)" MaximumValue="12"
            MinimumValue="6" Type="Integer">
        </asp:RangeValidator>
    </td>
</tr>

<tr>
    <td class="style3">
        Email:
    </td>

    <td class="style2">
        <asp:TextBox ID="txtemail" runat="server" style="width:250px">
        </asp:TextBox>
    </td>

    <td>
        <asp:RegularExpressionValidator ID="remail" runat="server"
            ControlToValidate="txtemail" ErrorMessage="Enter your
email"
            ValidationExpression="\w+([-.']\w+)*@\w+([-.\w+([-
.\w+)*"]>
        </asp:RegularExpressionValidator>
    </td>
</tr>

<tr>
    <td class="style3" align="center" colspan="3">
        <asp:Button ID="btnsubmit" runat="server"
onclick="btnsubmit_Click"
            style="text-align: center" Text="Submit"
style="width:140px" />
    </td>
</tr>
</table>
</form>

```

Figure Q14

