Version 1.0

General Certificate of Education
Advanced Subsidiary
Specimen Pre-release material for 2009 and 2010
Month Year

COMPUTING

COMP1/PR Problem Solving, Programming, Data Representation and Practical Exercise

Date Time

Pre-release material – Instructions for Candidates: Program Language Visual Basic. NET

To be given to candidates on or after 1 April XXXX

Information

- There are 2 parts to this pre-release material:
  - Section A: Skeleton Program
  - Section B: Test Data.
- This material will be re-printed in the examination paper.
- You are advised to familiarise yourself with this material before the examination.
- You will use both the program and test data in the examination and your teacher will provide you with access to these electronically at the start of the examination.
- You can view/download this material on the AQA Website: www.aqa.org.uk/xxxxxx

SECTION A – Skeleton Program

```
 Line     Program statements
Number
   1     Module Module1
   2
   3       ' AQA Pre-release Material for COMP1 – Visual Basic.NET (Console mode)
   4       ' Written by the COMP1 Principal Examiner Team
   5       ' developed with Visual Basic.NET 2005 Express Edition
   6       '
   7       ' Completed by:
   8       ' Candidate No:
   9       ' Centre No:
  10       '
  11       ' Some statements are incomplete
  12       ' and therefore do not do what they should do
  13       '
  14       ' This program is designed to
  15       '   accept a users menu choice to
  16       ' 1 - Read in a single hexadecimal number
  17       '     and display the binary equivalent
  18       ' 2 – Read in a file of hexadecimal numbers and
  19       '     produce a text file of their binary equivalent
  20       ' 3 – Display the contents of a text file
  21       ' 4 – Finish running the program
  22
  23       Sub Main()
  24         Dim Choice As Integer
  25         Dim Finished As Boolean
  26         Finished = False
  27
  28         ShowMenu()
  29         GetResponse(Choice)
  30         Select Case Choice
  31           Case 1 : ConvertNumber()
  32           Case 2 : ConvertFile()
  33           Case 3 : DisplayFile()
  34           Case 4 ' end program
  35         End Select
  36         Console.ReadLine()
  37       End Sub ‘ of main
  38
  39       Sub ShowMenu()
  40         Console.WriteLine("Please choose an option")
  41         Console.Writeline()
  42         Console.Writeline("1 - Convert a Hex numnber to Binary")
  43         Console.Writeline("2 - Convert a text file of Hex numbers")
  44         Console.WriteLine("3 - Display text file")
  45         Console.WriteLine("4 - Exit program")
  46         Console.Writeline()
  47       End Sub ‘ of ShowMenu
  48
  49       Sub GetResponse(ByRef Response As Integer)
  50         Console.Write("Enter option number: ")
  51         Response = Console.ReadLine()
  52       End Sub ‘ of GetResponse
  53
  54       Function Binary(ByVal Hex As String) As String
```

```
55          Dim Result As String
56          Dim HexDigit As Integer
57          Dim NoOfHexDigits As Integer
58          Dim ThisHexDigit As String
59          Dim BinaryEquivalent As String
60          Result = ""
61          NoOfHexDigits = Len(Hex)
62          For HexDigit = 1 To NoOfHexDigits
63            ThisHexDigit = Mid(Hex, HexDigit, 1)
64            If InStr("0123456789ABCDEF", ThisHexDigit) <> 0 Then
65              Select Case ThisHexDigit
66                Case "0" : BinaryEquivalent = ""
67                Case "1" : BinaryEquivalent = ""
68                Case "2" : BinaryEquivalent = ""
69                Case "3" : BinaryEquivalent = ""
70                Case "4" : BinaryEquivalent = ""
71                Case "5" : BinaryEquivalent = ""
72                Case "6" : BinaryEquivalent = ""
73                Case "7" : BinaryEquivalent = ""
74                Case "8" : BinaryEquivalent = ""
75                Case "9" : BinaryEquivalent = ""
76                Case "A" : BinaryEquivalent = ""
77                Case "B" : BinaryEquivalent = ""
78                Case "C" : BinaryEquivalent = ""
79                Case "D" : BinaryEquivalent = ""
80                Case "E" : BinaryEquivalent = ""
81                Case "F" : BinaryEquivalent = ""
82              End Select
83            Else
84            End If
85            Result = Result + BinaryEquivalent
86          Next
87          Binary = Result
88        End Function ‘ of Binary
89
90        Sub ConvertNumber()
91          Dim Hexadecimal As String
92          Dim Converted As String
93          Console.Write("Enter a Hexadecimal number: ")
94          Hexadecimal = Console.ReadLine
95          Converted = Binary(Hexadecimal)
96          Console.WriteLine(Converted)
97        End Sub ‘ of ConvertNumber
98
99        Sub ConvertFile()
100         Dim HexNumber As String
101         Dim BinaryNumber As String
102         FileOpen(1, "C:\HexData.txt", OpenMode.Input)
103         Console.WriteLine()
104         Do While Not EOF(1)
105           HexNumber = LineInput(1)
106           BinaryNumber = Binary(HexNumber)
107           Console.WriteLine(BinaryNumber)
108         Loop
109         FileClose(1)
110         Console.ReadLine()
111       End Sub ‘ of ConvertFile
112
113       Sub DisplayFile()
114         Dim NextNumber As String
115         FileOpen(1, "C:\BinaryData.txt", OpenMode.Input)
116         Console.WriteLine()
117         Do While Not EOF(1)
```

```
118           NextNumber = LineInput(1)
119           Console.WriteLine(NextNumber)
120         Loop
121         FileClose(1)
122       End Sub ‘ of DisplayFile
123
124     End Module
```

SECTION B – Test Data

000000000000
000001110000
000010000100
000010001000
000001110000
001000100010
000100100100
000010101000
000001110000
000000100000
000000100000
000000100000
000001010000
000010001000
000100000100

END OF PRE-RELEASE MATERIAL