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 C++

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    #include <stdio.h>
   2    #include <conio.h>
   3    #include <string.h>
   4
   5    // skeleton program written by Examiner team
   6
   7    // Completed by ......................
   8    // Candidate Number ...........
   9    // Centre Number ..............
  10
  11    // some statements are incomplete and
  12    // therefore do not do what they should do
  13
  14    //  This program is designed to
  15    //     accept a user's menu choice to
  16    //       1 - read in a single hexadecimal number
  17    //           and display the binary equivalent
  18    //       2 - read in a text file of hexadecimal numbers and
  19    //           produce a text file of their binary equivalent
  20    //       3 - display the content of a text file
  21    //       4 - finish running the program
  22
  23    int choice, finished;
  24
  25    void ShowMenu();
  26    void GetResponse(int &response);
  27    char* Binary(char Hex[4]);
  28    void ConvertNumber(void);
  29    void ConvertFile();
  30    void DisplayFile();
  31
  32    void main(void)
  33    {
  34      finished = 0;
  35
  36      ShowMenu();
  37      GetResponse(choice);
  38      switch (choice)
  39      {
  40        case 1 : ConvertNumber(); break;
  41        case 2 : ConvertFile();break;
  42        case 3 : DisplayFile();break;
  43        case 4 : ;  // end program
  44      }
  45      getch();
  46    }     // end of main function
```

```
47    void ShowMenu()
48    {
49      printf("\nPlease choose an option\n");
50      printf("\n");
51      printf("    1 - Convert a Hex number to Binary\n");
52      printf("    2 - Convert a text file of Hex numbers\n");
53      printf("    3 - Display text file\n");
54      printf("    4 - exit program\n");
55      printf("\n");
56    }  // end of ShowMenu
57
58    void GetResponse(int &response)
59    {
60      printf(" Enter option number :  ");
61      flushall();
62      scanf("%i",&response);
63    }  //end of GetResponse
64
65    char* Binary(char Hex[6])
66    {
67      int HexDigit, NoOfHexDigits;
68      char ThisHexDigit, BinaryEquivalent[4], Result[24];
69      strcpy(Result,"");
70      NoOfHexDigits = strlen(Hex);
71      for(HexDigit=0;HexDigit<NoOfHexDigits;HexDigit++)
72      {
73        ThisHexDigit = Hex[HexDigit];
74        if((ThisHexDigit >='0') && (ThisHexDigit<='9')
75           || (ThisHexDigit>='A') && (ThisHexDigit<='F'))
76        {
77          switch (ThisHexDigit)
78          {
79            case '0' : strcpy(BinaryEquivalent,"");break;
80            case '1' : strcpy(BinaryEquivalent,"");break;
81            case '2' : strcpy(BinaryEquivalent,"");break;
82            case '3' : strcpy(BinaryEquivalent,"");break;
83            case '4' : strcpy(BinaryEquivalent,"");break;
84            case '5' : strcpy(BinaryEquivalent,"");break;
85            case '6' : strcpy(BinaryEquivalent,"");break;
86            case '7' : strcpy(BinaryEquivalent,"");break;
87            case '8' : strcpy(BinaryEquivalent,"");break;
88            case '9' : strcpy(BinaryEquivalent,"");break;
89            case 'A' : strcpy(BinaryEquivalent,"");break;
90            case 'B' : strcpy(BinaryEquivalent,"");break;
91            case 'C' : strcpy(BinaryEquivalent,"");break;
92            case 'D' : strcpy(BinaryEquivalent,"");break;
93            case 'E' : strcpy(BinaryEquivalent,"");break;
94            case 'F' : strcpy(BinaryEquivalent,"");break;
95          }
96        }
97        else
98        {
99        }
100       strcat(Result,BinaryEquivalent);
101       strcpy(BinaryEquivalent,"");
102     }  // end of for loop
103     return(Result);
104   }     // end of function Binary
105
106   void ConvertNumber(void)
107   {
108     char Hexadecimal[6], Converted[24];
109     printf("Enter a Hexadecimal number \n");
110     scanf("%s",Hexadecimal);
```

```
111       strcpy(Converted, Binary(Hexadecimal));
112       printf("%s\n",Converted);
113     }  // end of ConvertNumber
114
115     void ConvertFile()
116     {
117       char FileNameIn[30];
118       FILE *HexFile;
119       char HexNumber[6], BinaryNumber[24];
120       strcpy(FileNameIn,"Hexdata.dat");
121       HexFile = fopen(FileNameIn,"r");
122       fgets(HexNumber,6,HexFile);
123       while (!feof(HexFile))
124       {
125         strcpy(BinaryNumber,Binary(HexNumber));
126         printf("%s\n",BinaryNumber);
127         fgets(HexNumber,6,HexFile);
128       }
129       fclose(HexFile);
130     }  // end of ConvertFile
131
132     void DisplayFile()
133     {
134       char FileName[30], bit;
135       FILE *BinaryFile;
136       strcpy(FileName,"Binaryda.dat");
137       BinaryFile= fopen(FileName,"r");
138       bit = fgetc(BinaryFile);
139       while(bit !=EOF)
140       {
141         if ( bit == '\n')
142         {
143           printf("\n");
144         }
145         else
146         {
147           printf("%c",bit);
148         }
149         bit = fgetc(BinaryFile);
150       }
151       fclose(BinaryFile);
152     }    // end of DisplayFile
```

SECTION B – Test Data

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

END OF PRE-RELEASE MATERIAL