UNIVERSITY OF PORTSMOUTH

FACULTY OF TECHNOLOGY

Department of Electronic and Computer Engineering

B142L – Introduction to Computing

U13746

Date: **July 2007** **Time:** **2 hours**

INSTRUCTIONS

Write your student ID number clearly on page 2.

Write your answers to all 10 questions within the spaces provided in this examination paper.

Handwritten notes are permitted with this examination.

Calculators permitted are:

Casio FX 85WA

Casio FX 83WA

Casio FX 85MS

Examiner:

Mr Chi Nguyen

Student ID Number

Consider the following source code to answer Question 1. Line numbers have been provided at the start of each line for reference. **Indicate specific line numbers** in your answers when necessary.

```
01:  int main( void );
02:  {
03:    char model[] = ‘desktop’;
04:    int price = 400.50;
05:    long quantity = “11”;
06:
07:    printf( "There are %d %s computers on sale for %.2f each.",
quantity, price, model );
08:  }
```

QUESTION 1. Modify the source code to correct all errors. [10 marks]

```
int main( void )
{
  char model[] = "desktop";
  float price = 400.50;
  long quantity = 11;

  printf( "There are %d %s computers on sale for %.2f each.", quantity,
model, price );
}
```

Consider the following source code to answer Question 2. Line numbers have been provided at the start of each line for reference. **Indicate specific line numbers** in your answers when necessary.

```
01:  int main( void )
02:  {
03:    int input = -100;
04:
05:    scanf( "%d", &input );
06:
07:    if ( input > 100 ) printf( "Orange" );
08:      else printf( "Apple" );
09:    if ( input < 200 ) printf( "Grape" );
10:  }
```

QUESTION 2A. Describe all valid input values that would cause the following output to be displayed: [5 marks]

OrangeGrape

Integer values greater than 100 and less than 200.

QUESTION 2B. List all valid program outputs. [5 marks]

There are 3 possible outputs:

Orange

OrangeGrape

AppleGrape

QUESTION 3A. Place an "X" in the box next to **3 terms** that are **most directly related to the use of for loops** in a C program. [3 marks]

	selection	X	initialization
X	iteration		assignment
	output		declaration
	input	X	test condition
	format		error condition

QUESTION 3B. Write a C program **that uses a for loop to display all positive odd integers less than 700. Show or use in your program all of the terms you selected in Question 3A.** [7 marks]

```
int main( void )
{
  int counter;

  /* counter is set to 1 during initialization */
  /* the test condition checks that counter is less than 700 */
  for ( counter = 1; counter < 700; counter += 2 )
    printf( "%d\n", counter ); /* Executes at each iteration */
}
```

QUESTION 4. Write a C program that **uses a while loop to collect integer inputs and ends when a zero value is entered. The program must display the minimum and maximum of all input values.** [10 marks]

```
int main( void )
{
  int max = 0;
  int min = 0;
  int input = 0;

  scanf( "%d", &input );
  max = input;
  min = input;

  while ( input != 0 )
  {
    if ( input > max ) max = input;
    if ( input < min ) min = input;
    scanf( "%d", &input );
  }

  printf( "\n Min: %d \t Max: %d \n", min, max );
}
```

Consider the following source code to answer Question 5. Line numbers have been provided at the start of each line for reference. **Indicate specific line numbers** in your answers when necessary.

```
01:  int main( void )
02:  {
03:    int marks[ 5 ] = { 72, 54, 61, 49, 67, 69 }
04:    int count;
05:    for ( counter == 0; counter <= 6; counter++ )
06:    {
07:      if ( marks[ counter ] >= 70 ) { printf ( "A \n" ); continue; }
08:      if ( marks[ counter ] >= 60 ) { printf ( "B \n" ); continue; }
09:      if ( marks[ counter ] >= 50 ) { printf ( "C \n" ); continue; }
10:      if ( marks[ counter ] >= 40 ) { printf ( "D \n" ); continue; }
11:    }
12:  }
```

QUESTION 5. Modify the source code to correct all errors. [10 marks]

```
int main( void )
{
  int marks[ 6 ] = { 72, 54, 61, 49, 67, 69 };
  int counter;
  for ( counter = 0; counter < 6; counter++ )
  {
    if ( marks[ counter ] >= 70 ) { printf ( "A \n" ); continue; }
    if ( marks[ counter ] >= 60 ) { printf ( "B \n" ); continue; }
    if ( marks[ counter ] >= 50 ) { printf ( "C \n" ); continue; }
    if ( marks[ counter ] >= 40 ) { printf ( "D \n" ); continue; }
  }
}
```

Consider the following source code to answer Question 6. Line numbers have been provided at the start of each line for reference. **Indicate specific line numbers** in your answers when necessary.

```
01:  #include <string.h>
02:
03:  int main( void )
04:  {
05:    char buffer[ 128 ] = { '\0' };
06:    int counter;
07:
08:    scanf( "%s", buffer );
09:  }
```

QUESTION 6A. Modify the source code **to display a duplicate of each letter in the input buffer string.** For example, an input string "welcome" should be displayed as "wweellccoommee" on screen. [5 marks]

```
Add between lines 8 and 9:

  for ( counter = 0; counter < strlen( buffer ); counter++ )
    printf( "%c%c", buffer[ counter ], buffer[ counter ] );
```

QUESTION 6B. Modify the source code to **display the input buffer string with all 'b' characters replaced with 'B' characters.** [5 marks]

```
Add between lines 8 and 9:

  for ( counter = 0; counter < strlen( buffer ); counter++ )
  {
    if ( buffer[ counter ] == 'b' ) buffer[ counter ] = 'B';
    printf( "%c", buffer[ counter ] );
  }
```

QUESTION 7A. Place an "X" in the box next to **3 terms** that are **most directly related to the use of functions** in a C program. [3 marks]

	template	X	parameter
X	prototype		element
	extern variable		member
X	local variable		default
	global variable		break

QUESTION 7B. Write a C function **that accepts two input values. If the input character is 'C', the function converts the input floating point number to Fahrenheit and returns that value. If the input character is 'F', the function converts the input floating point number to Celsius and returns that value. Show or use in your function all of the terms you selected in Question 7A.** Use the formula below. [7 marks]

Celsius = 5 / 9 * (Fahrenheit – 32)

```
/* prototype */
float convert( char, float );

/* scale and input are parameters */
float convert( char scale, float input )
{
  /* output is a local variable */
  float output = input;
  if ( scale == 'C' ) output = 32.0 + ( input * 9.0 / 5.0 );
  if ( scale == 'F' ) output = 5.0 / 9.0 * ( input - 32.0 );
  return output;
}
```

QUESTION 8. Write a C function that **returns the number of times a specific character is found in a string. The first input value is the character to find and count. The second input value is a pointer to the string to be searched.** The prototype is provided below. [10 marks]

```
int findCharacter( char, char* );
```

```
int findCharacter( char target, char* input )
{
  int matches = 0;
  int counter;

  for ( counter = 0; counter < strlen( input ); counter++ )
    if ( input[ counter ] == target ) matches++;

  return matches;
}
```

QUESTION 9A. Place an "X" in the box next to **3 terms** that are **most directly related to the use of data files** in a C program. [3 marks]

	data array
	file array
X	file pointer
X	fopen()
	close()

	MODE_A
	MODE_R
	MODE_W
	END
X	EOF

QUESTION 9B. Write a C program to **read and display the integers in the data file until you find a -1 value or reach the end of file.** "C:/voltage.txt" is the file location. **Show or use in your program all of the terms you selected in Question 9A.** [7 marks]

```
#include <stdio.h>

int main( void )
{
  int input = 0;
  /* inputFile is the name of the file pointer variable */
  FILE* inputFile = fopen( "C:/voltage.txt", "r" );

  if ( inputFile != NULL )
  {
    while ( fscanf( inputFile, "%d", &input ) != EOF )
    {
      if ( input == -1 ) break;
      printf( "%d \n", input );
    }

    fclose( inputFile );
  }
}
```

Consider the following source code to answer Question 10. Line numbers have been provided at the start of each line for reference. **Indicate specific line numbers** in your answers when necessary.

```
01:  #include <stdio.h>
02:  #define FILENAME "C:/absolute.txt"
03:
04:  int main( void )
05:  {
06:    int output = 0;
07:  }
```

QUESTION 10A. Modify the source code **to accept integer inputs from the keyboard and write the values out to the data file. The program ends when a zero input value is entered.** Use the file name provided in the source code for output. [7 marks]

```
Add after line 6:

07:    FILE* outputFile = fopen( FILENAME, "w" );
08:    if ( outputFile != NULL )
09:    {
10:      while ( scanf( "%d", &output ) != EOF )
11:      {
12:        if ( output == 0 ) break;
13:        fprintf( outputFile, "%d\n", output );
14:      }
15:      fclose( outputFile );
16:    }
17:  }
```

QUESTION 10B. Modify the source code for Question 10A to **replace all negative input numbers with their absolute value before writing to the data file.** For example, an input value of -25 should be saved to file as number 25. [3 marks]

```
Add between lines 12 and 13 in solution for Question 10A:

      if ( output < 0 ) output = output * -1;
```