



Department of Electronic and Computer Engineering

B142L Introduction to Computing

Semester 2 Examination

1.5 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.

Additional Information:

This is an open book exam

Provided:

NIL

Calculators:

Casio FX 85 Series or Casio FX 83 Series

Examiners:

Mr C Nguyen

External Examiner:

Dr A Rossiter

Student ID Number

SOLUTIONS

QUESTION 1

Consider the following source code. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary.

```
01: #include <stdio.h>
02:
03: void countDown( int );
04:
05: int main( void )
06: {
07:     int input;
08:
09:     scanf( "%d", &input );
10: }
```

- a) Add source code to **define the countDown() function which displays numbers starting at the input value and decreasing by one until it reaches zero**. For example, the function would display the following input when the input value is 6.

6 5 4 3 2 1 0

[8 Marks]

Add after line 10:

```
void countDown( int input ) [2]
{
    int counter; [1]

    for ( counter = input; counter >= 0; counter-- ) [4]
        printf( "%d ", counter ); [1]
}
```

- b) Modify the source code to **call the countDown() function in the program to display the output**.

[2 Marks]

Add between lines 9 and 10:

```
countDown( input ); [2]
```

QUESTION 2

- a) Place an "X" in the box next to **3 terms** that are most directly related to **selection and decisions** in the C language. [3 Marks]

	continue
	break
1	else
	when
1	if

1	and
	then
	test
	scanf
	printf

- b) Write a C program to **ask for an input integer and display output messages to indicate whether the input is a positive number that is a multiple of 3, 4 or 5**. Show or use all the terms selected in part (a). [7 Marks]

Similar to:

```
#include <stdio.h>

int main( void )
{
    int input, multiple;

    scanf( "%d", &input ); [1]

    /* Test condition uses AND logic operator */
    for ( multiple = 3; (input > 0) && (multiple <= 5); multiple++ ) [3]
    {
        if ( ( input % multiple ) == 0 ) printf( "Multiple of %d\n", multiple ); [2]
        else printf( "Not a multiple of %d\n", multiple ); [1]
    }
}
```

QUESTION 3

Consider the following source code. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary.

```
01: #include <stdio.h>
02:
03: int main( void )
04: {
05:     char input[ 25 ] = { '\0' };
06:     int counter = 3;
07:
08:     scanf( "%s", input );
09:
10:     while ( input[ counter ] != NULL )
11:     {
12:         printf( "%c", input[ counter ] );
13:         counter++;
14:     }
15: }
```

a) List the program output when the **input value is the string "mountain"**. [4 Marks]

ountain

[4]

b) Describe **the longest possible output display** using this program. [6 Marks]

String containing **21 or 22 characters**

[6]

QUESTION 4

Modify the following source code to correct all errors. The program is intended to ask for an input string and test if it is a valid password format. A valid password must be 4-16 characters in length and must not contain spaces. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary. **[10 Marks]**

```
01: #include <stdio.h>
02: #include <ctype.h>
03:
04: int main( void )
05: {
06:     char password[ 65 ] = { '\0' };
07:     int valid = 1, index;
08:
09:     scanf( "%64c" );
10:     for ( index = 0; (valid == 1) || (index < strlen( password )); index++ )
11:     {
12:         if ( strlen( password ) > 4 ) valid = 0;
13:         else if ( index > 16 ) valid = 0;
14:         else if ( isspace( index ) ) valid = 0;
15:     }
16:
17:     if ( index == 1 ) printf( "Valid password" );
18:     else printf( "Invalid password" );
19: }
```

```
#include <stdio.h>
#include <ctype.h>

int main( void )
{
    char password[ 65 ] = { '\0' };
    int valid = 1, index;

    scanf( "%64c", password );
    for ( index = 0; (valid == 1) && (index < strlen( password )); index++ )
    {
        if ( strlen( password ) < 4 ) valid = 0;
        else if ( index >= 16 ) valid = 0;
        else if ( isspace( password[ index ] ) ) valid = 0;
    }

    if ( valid == 1 ) printf( "Valid password" );
    else printf( "Invalid password" );
}
```

QUESTION 5

a) Place an "X" in the box next to **3 terms** that are most directly related to the use of **iterative loops** in a C program. [3 Marks]

	repeat
	assign
1	while
	then
	main

	head
1	body
	tail
	declaration
1	initialization

b) Write a C program to **display the ASCII value of all uppercase letters, i.e. A-Z** (inclusive). Show or use all the terms selected in part (a). [7 Marks]

Similar to:

```
#include <stdio.h>
```

```
int main( void )
```

```
{
```

```
    int letter = 'A'; /* initialization */
```

[2]

```
    while ( letter <= 'Z' )
```

[3]

```
    {
```

```
        printf( "%d ", letter++ ); /* body */
```

[2]

```
    }
```

```
}
```

QUESTION 6

Consider the following source code. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary.

```
01: #include <stdio.h>
02: #include <stdlib.h>
03: #include <time.h>
04:
05: int getRandomDayOfWeek( void );
06:
07: int main( void )
08: {
09:     srand( time( 0 ) );
10:     int dayOfWeek = getRandomDayOfWeek();
11: }
```

- a) Add source code to **define the `getRandomDayOfWeek()` function** that **generates a random number in the range of 1-7** (inclusive). [5 Marks]

Similar to:

```
int getRandomDayOfWeek( void ) [2]
{
    return ( ( rand() % 7 ) + 1 ); [3]
}
```

- b) Modify the **`main()`** function to use a **switch statement** that displays the **name of the day of the week** based on value of the variable **`dayOfWeek`**, where **1 is Sunday** and **7 is Saturday**. [5 Marks]

Similar to:

Add between lines 10 and 11:

```
switch ( dayOfWeek ) [2]
{
    case 1 : printf( "Sunday" ); break;
    case 2 : printf( "Monday" ); break;
    case 3 : printf( "Tuesday" ); break;
    case 4 : printf( "Wednesday" ); break; [3]
    case 5 : printf( "Thursday" ); break;
    case 6 : printf( "Friday" ); break;
    case 7 : printf( "Saturday" );
}
```

QUESTION 7

Consider the following source code. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary.

```
01: int main( void )
02: {
03:     int counter;
04:     float prices[ 20 ] = { 0.0 };
05: }
```

a) Add source code to **accept and store 20 input prices in the array.**

[4 Marks]

Similar to:

Add between lines 4 and 5:

```
for ( counter = 0; counter < 20; counter++ )           [2]
{
    scanf( "%f", &(amp; prices[ counter ] ) );         [2]
}
```

b) Add source code to **reduce each price in the array by 10. However, do not change the price if the reduced price would be zero or less.**

[6 Marks]

Similar to:

Add between lines 4 and 5 after the input loop:

```
for ( counter = 0; counter < 20; counter++ )           [2]
{
    if ( ( prices[ counter ] - 10.0 ) > 0 )             [2]
        prices[ counter ] = prices[ counter ] - 10.0; [2]
}
```

QUESTION 8

Consider the following source code. Line numbers have been provided for reference. Indicate specific line numbers in your answers when necessary.

```
01: #include <stdio.h>
02:
03: int main( void )
04: {
05:     int counter = 100;
06:
07:     do
08:     {
09:         if ( ( counter + 5 * 4 - 2 ) > 0 ) printf( "%d ", counter );
10:         else printf( "%d ", counter * (-1) );
11:
12:         counter++;
13:     } while ( counter < 200 );
14: }
```

a) Describe the program output.

[5 Marks]

Similar to:

Integer numbers 100 to 199.

[5]

b) Modify the program to use a **for** loop instead of the **do/while** loop.

[5 Marks]

Similar to:

```
#include <stdio.h>
```

```
int main( void )
```

```
{
```

```
    int counter = 100;
```

```
    for ( counter = 100; counter < 200; counter++ )
```

[3]

```
    {
```

```
        if ( ( counter + 5 * 4 - 2 ) > 0 ) printf( "%d ", counter );
```

[1]

```
        else printf( "%d ", counter * (-1) );
```

[1]

```
    }
```

```
}
```

QUESTION 9

- a) Place an "X" in the box next to **3 terms** that are most directly related to the use of **text strings** in a C program. [3 Marks]

	scope
	local
1	length
	global
	automatic

	zero
1	char
1	array
	binary
	boundary

- b) Write a C program that **keeps asking for an input string until the length of the string is an even number**. Show or use all the terms selected in part (a). [7 Marks]

Similar to:

```
#include <stdio.h>
#include <string.h>
```

```
int main( void )
```

```
{
  char input[ 256 ] = { '\0' }; /* array length is 256 */ [2]
```

```
  do
```

```
  {
    scanf( "%s", input ); [1]
```

```
  } while ( ( ( strlen( input ) ) % 2 ) == 1 ); [4]
```

```
}
```

QUESTION 10

Write a C program that **accepts 200 input integers** and displays the following output: **[10 Marks]**

- **Count** of **positive** input values
- **Subtotal** of all **positive** input values
- **Count** of **negative** input values
- **Subtotal** of all **negative** input values

Similar to:

```
#include <stdio.h>

int main( void )
{
    int input, counter, positiveCounter, positiveSum, negativeCounter, negativeSum;

    input = positiveCounter = positiveSum = negativeCounter = negativeSum = 0;
    for ( counter = 0; counter < 200; counter++ )           [1]
    {
        scanf( "%d", &input );                               [1]
        if ( input > 0 )                                     [1]
        {
            positiveCounter++;                               [1]
            positiveSum = positiveSum + input;               [1]
        }
        else if ( input < 0 )                               [1]
        {
            negativeCounter++;                               [1]
            negativeSum = negativeSum + input;               [1]
        }
    }

    printf( "Found %d positive numbers and their subtotal is %d\n",
            positiveCounter, positiveSum );                  [1]
    printf( "Found %d negative numbers and their subtotal is %d",
            negativeCounter, negativeSum );                  [1]
}
```