UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER I
SESSION 2013/2014

COURSE NAME	:	COMPUTER PROGRAMMING
COURSE CODE	:	BEC 10102 / BEE 1212
PROGRAMME	:	BEJ
EXAMINATION DATE	:	JANUARY 2014
DURATION	:	2 HOURS
INSTRUCTIONS	:	1. ANSWER **ALL** QUESTIONS IN **SECTION A** AND **ONE (1)** QUESTION IN **SECTION B.** 2. ANY ANSWER WRITTEN IN PENCIL WILL **NOT BE EVALUATED.** 3. STUDENT IS **NOT ALLOWED** TO BRING OUT THE QUESTION PAPER.

THIS QUESTION PAPER CONSISTS OF **SEVEN (7)** PAGES

SECTION A (50 MARKS)

INSTRUCTION: Answer ALL questions. Any answer written in pencil will not be evaluated.

Q1 (a) Write a program that ask for user input from 5 to 9. Your should:

(i) show the sequence control structure using psedo code.

(4 marks)

(ii) write a complete C++ program with comments to calculate the average

(6 marks)

(b) The electric company charges customers according to the following rate schedule for kilowatt-hours (kWh) used:

- 8 cents a kWh for the first 300 kWh
- 6 cents a kWh for the next 300 kWh (up to 600 kWh)
- 5 cents a kWh for the next 400 kWh (upto1000 kWh)
- 3 cents a kWh for all electricity used over 1000 kWh

Write a complete C++ program with comments to calculate the total charge for a customer. Your program should:

(i) show the sequence control structure using flowchart.

(5 marks)

(ii) obtain the kilowatt-hours used by a customer.

(4 marks)

(iii) calculate and the charge for electricity.

(3 marks)

(iv) display the charge in dollars with 2 digits after the decimal point.

(3 marks)

(c) Determine which of the statement in (i) to (v) is TRUE or FALSE.

(i) A variable name indirectly references a value, whereas a pointer directly references a value.

(ii) The * operator is referred as dereferencing operator.

(iii) A pointer can be initialised using 0, NULL or an address.

(iv) The && operator returns the memory address of its operand.

(v) *void Display (int *sum)* shows the *Display* function is called using pass-by-reference mechanism.

(5 marks)

Q2 (a) Consider the following function declarations:

```
double volume(double l, double w, double &h); // function 1
double volume(double);                        // function 2
```

State whether the following statements are TRUE or FALSE in (i) to (v).

(i) The parameter list for function 2 will cause a compiler error. Because the parameter name is missing.

(ii) Parameter h in function 1 can be used to modify the value of the corresponding argument in a function call.

(iii) The return type of function 2 is *double*.

(iv) Call by reference is used to provide a value for parameter w in function.

(v) You cannot have two functions named volume in the same program.

(5 marks)

(b) What is the difference between an ARRAY and a LIST?

(5 marks)

(c) Write a function that swaps the values of two integers, using int* as the argument type.

Explain how to check a linked list is circular.

(10 marks)

SECTION B (10 MARKS)

INSTRUCTION: Answer ONE question only. Any answer written in pencil will not be evaluated.

Q3 (a) Analyse the following program of *Program Q3(a)*. (*Note that line numbers have been added to you identify certain parts of the program.*)

```
 1  /*Program Q3(a)*/
 2  #include<iostream.h>
 3  int   balance_owed;     //amount owed
 4
 5  main()
 6   {
 7       cout<<"Enter number of dollars owed:";
 8       cin>>balance_owed;
 9
10       if (balance_owed= 0)
11           cout<<"You owe nothing.\n";
12       else
13          cout<<"You owe"<<balance_owed<<
14          "dollars.\n";
15
16       return(0);
17   }
```

For some strange reason, the given *Program Q3(a)* thinks that everyone owes a balance of 0 dollars. Why?

(5 marks)

(b) Analyse the following program of *Program Q3(b)*. (*Note that line numbers have been added to you identify certain parts of the program.*).
The keep-it-simple system of programming prevents us from using the increment (++) and decrement (--) operators except on a line by themselves.
When used in an expression, they are considered side effects, and this can lead to unexpected results as illustrated in *Program Q3(b).*

```
 1  /*Program Q3(b)*/
 2  #include <iostream.h>
 3  #define SQR(x) ((x)* (x))
 4  main()
 5  {
 6          int counter;            /*Counter for loop*/
 7          counter= 0;
 8          while (counter<5)
 9                          cout<<"x    <<counter+ 1 <<
10                          " x squared"<<SQR(++counter)<<'\n';
11          return(0);
12  }
```

(i) What will be the expected result?

(ii) Explain how the output in *Q3(b)(i)* is produced.

(5 marks)

Q4 Analyse the program of *Program Q4(a)*. (*Note that line numbers have been added to you identify certain parts of the program.*)

```
 1. /*Program Q4(a)*/
 2. #include<iostream.h>
 3. /*
 4.  This program produces a Celsius to Fahrenheit
    conversion
 5. */
 6. //The current Celsius temperature we are working
    with int celsius;
 7. main(){
 8. For (celsius= 0;celsius<=100;++celsius);
 9. cout<<"Celsius:" <<celsius<<
10. " Fahrenheit:" <<((celsius* 9)/ 5 + 32)<<'\n';
11. return(0);
12. }
```

(a) Investigate the output of the program. The *Program Q4(a)* contains an error.

When run, this program prints out: Celsius:101 Fahrenheit:213
and nothing more. Why?

(5 marks)

(b) The *Program Q4(b)* below reads a list of five numbers and counts the number of threes and sevens in the data.

```
/*Program Q4(b)*/
include <iostream.h>
int seven_count;  /* Number of sevens in the data int data [5];      The data
to count 3 and 7 in int three_count;  Number of threes in the data int index;
Index into the data

main(){
seven_count= 0;
three_count= 0;

cout<<"Enter5 numbers\n";
cin>>data[1]>>data[2]>>data[3]>> data[4]>>data[5];

for  (index= 1;index<=5;++index)
if (data[index]==3)
++three_count;
If (data[index]==7)
++seven_count;
}

cout<<"Threes" <<three_count<<" Sevens" <<seven_count<<'\n';
return(0);
}
```

Why does it give us the wrong answers?

(5 marks)

- END OF QUESTIONS -