# UNIVERSITY OF DUBLIN

## TRINITY COLLEGE

Faculty of Engineering, Mathematics & Science
School of Computer Science & Statistics

**B.A.(Mod.) Computer Science** **Trinity Term 2011**
Senior Freshman Examination

## Systems Programming I and II (CS2014/5)

**Tuesday 10th May 2011** **RDS Main Hall** **14:00 – 17:00**

**Dr David Gregg**

---

**Instructions to Candidates:**

- Answer 4 out of the 6 questions
- All questions are marked out of 25
- All program code should be commented, indented and use good programming style

**Materials permitted for this examination:**

- Calculator.

1. Bad programming often makes programs difficult to understand, and unfortunately we often have to understand poorly written code. Describe what the following pieces of C code do, and write new versions that provide the same functionality but with simple programming style, appropriate function and variable names, and suitable indentation.

```
/* code section A */
void e (char * a, char * b) { while (*(a++) = *(b++)); }
```

[5 marks]

```
/* code section B */
int f (int * a, int b, int item) {
        if ( b == -1 ) return 0;
        else return ( a[b] == item) + f(a, b-1, item);
}
```

[5 marks]

```
/* code section C */
int g (int * y, unsigned size, int z) {
int tmp = y[0];
int * b = y + size;
y[0] = z;
while ( 1 ) if ( *(--b) == z) { y[0] = tmp; return b – y; } ;
y[0] = tmp;
if ( tmp == z ) return 0;
else return -1;
}
```

[7 marks]

```
/* code section D */
void h (int ** a, int b) {
for (int x=1; x <= b; x++) for (y=1; y <= b; y++ ) a[x-1][y-1] = (i/j) * (j/i);
}
```

[8 marks]

2. Run length encoding is a common and simple way of compressing binary data. It is particularly successful at compressing monochrome images such as a fax. Given an arbitrary sequence of binary data, run length encoding counts the "runs" of ones and zeroes, and outputs a sequence of integers representing the lengths of the runs.

   For example, given the following binary stream:
   000000110000000000011111000
   The output of run length encoding will be:
   6, 2, 10, 5, 3
   This would be interpreted as:
   6 zeroes, followed by two ones, followed by ten zeroes, etc.

   Write a C function that implements run length encoding and has the following prototype:
   void runLengthEncode(unsigned char * bitstream, int size, int * result);

   Where *bitstream* is the sequence of bits to be encoded, and *size* is the length of the stream in bytes. Note that each element of the *bitstream* array is an unsigned character containing 8 bits, and you need to encode all the bits in this array. Your function should write the sequence of integers in the run length encoding into the array *result*. The end of the encoding should be marked by the integer -1. You may assume that the *result* array is always sufficiently large to store the encoding.

   [25 marks]

3. The most commonly encountered logics use two states: true and false. However, fuzzy logics can also be useful for dealing with approximate situations where we deal with the *degree of truth* of a statement. Typically, the degree of truth is represented by a floating point value in the range zero to one (inclusive). Given a fuzzy logic value *X*, the expression *truth(X)* returns this value between 0 and 1.

   Write a C++ class to represent fuzzy logic values. Your class should provide && (and), || (or) and ! (not) operators. You should also overload the << and >> operators for input and output. The three main fuzzy logic operators behave as follows:

   X && Y = minimum(truth(X), truth(Y))

   X || Y = maximum(truth(X), truth(Y))

   !X = 1 – truth(X)

   The resulting value in the range 0 to 1 from the operator becomes the truth value of the fuzzy logic expression returned by the operator.

   [25 marks]

4. The C++ Standard Template Library (STL) provides a set of standard container classes for use in C++ programs. One of the most important of these is the list class, which implements a doubly-linked list. The following shows the broad outline of a simplified version of the STL list class.

```
template <class T>
class mylist{
private:
        // add your own private variables and methods here
public:
        mylist( );              // create new, empty list
        ~mylist( );             // destructor
        void push_back(T item);    // add new item onto end of list
        void push_front(T item);   // add a new item to front of list
        T pop_back( );          // remove and return last item of list
        T pop_front( );         // remove and return first item of list
        T& at(int  i);          // return the i'th element of the list
        void sort( );           // sort the elements of list
};
```

Add the remaining declarations to this class, and provide the bodies of methods to implement each of the methods listed above. You may use the basic building blocks of the C++ language (arrays, classes) to construct your class, but you may not use the STL in your code.

[25 marks]

5. Write a C++ function to compute the mass of a molecule, given its chemical formula. Your function should have the following prototype:
int compute_mass(string formula, map<string, int> weights);

The first parameter is a C++ string containing the chemical formula of the compound. For example, $H_2SO_4$ would be represented by the string "H2SO4". Similarly, NaCl would be represented with the string "NaCl".

The second parameter is an STL map, mapping strings to integers. For each element, this map records its corresponding atomic weight. Note that the first letter in the name of an element is always upper case, and subsequent letters are always lower case. Therefore, it is always possible to know whether the next character is part of the name of the current element (lower case) or the start of the name of a new element (upper case).

[25 marks]

6. (a) The data memory of a program is usually divided into (i) the stack, (ii) the heap, and (iii) the static/global variable areas. Describe the different types of variables that occupy these areas, the lifetimes of these variables, and the way in which memory is allocated and de-allocated for each of these types of variables. Your description should cover the different types of variables in each of the programming languages C, C++ and Java. Your answer should include simple pictures, where appropriate.

[13 marks]

(b) Write egrep regular expression commands to find all lines in a file described by each of the following:

(i) Lines containing the string 'while'.

[1.5 marks]

(ii) Lines containing any vowel.

[1.5 marks]

(iii) Lines which end with a full stop.

[1.5 marks]

(iv) Empty lines in the file.

[1.5 marks]

(v) Lines containing the same character repeated twice in a row. E.g. "AA".

[1.5 marks]

(vi) Lines containing violations of the "i after e rule", that is lines containing the sequence "ei", where the "ei" is not directly preceded by 'c'. (So the word "vein" is a violation of the rule, but "ceiling" is not).

[1.5 marks]

(vii) Lines containing a number with digits both before and after the decimal point. E.g. something like "3.5" or "18.293".

[1.5 marks]

(viii) Lines containing the same word more than once.

[1.5 marks]