UNIVERSITY OF DUBLIN

MA1262-1

TRINITY COLLEGE

Faculty of Science

school of mathematics

JF Maths/TP/TSM **Trinity Term 2014**

Mathematical Computation II

Saturday, May 10 14:00 — 16:00

Prof. Colm Ó Dúnlaing

Attempt 3 questions.

1. There are four mistakes in the following code.

   (a) Try to identify and correct four mistakes.

   (b) Describe what is printed, after correction.

```
#include <iostream>
#include <vector>
#include <algorithm>
int main()
{ vector <double> v;
  double x, s;
  while ( ! cin.eof () )
  { cin >> x;
    s += x;
    v.push_back ( x );
  }
  sort ( v.begin(), v.end() );

  cout << v.size() << ' ' << v[0] << ' ' << s/v.size() << ' '
    << v[v.size()/2] << ' ' <<  v[v.size()] << endl;
  return 0;
}
```

2. Given

```
double a[5][10]; int b[90]
```

   suppose that `a` begins at address 1500 and `b` begins immediately after `a`.

   (a) What are the sizes of `a` and `b` in bytes, given that `double` occupies 8 bytes, and `int` 4 bytes?

   (b) Evaluate the address of `a[1][2]` and of `b[5]`.

(c) The array entry `a[6][1]` is outside the bounds of array `a` but within the bounds of `b`: i.e., it coincides with `b[`i`]` for some i, $0 \leq i < 90$. Calculate i.

(d) One of the following routine prototypes is valid, one is not. Say which one is, and why.

```
void aa ( int n, double x[][10] );
void bb ( int n, double y[10][] );
```

3. (a) Carefully simulate the following, showing what gets printed

```
#include <iostream>
using namespace std;
double xxx ( double x, int n )
{ int i;
  for (i=0; i<n; ++i)
  { x = x*x; }
  return x;
}
int main()
{ cout << "xxx(2,3) is " << xxx(2,3) << endl;
  return 0;
}
```

(b) What does `xxx ( x, n)` return in general, given $n \geq 0$?

(c) Carefully evaluate the following expressions, being careful to produce the answer in the correct type (`int, double, char`). Indicate doubles with decimal points.

```
  (i) 1/2/3
 (ii) 1/2/3+0.5
(iii) 1/2.0/3 + 0.5
 (iv) 1-2-3-4
  (v) (3.2 + 4)/3+3
 (vi) (char) ('h'-'a'+'A')
```

4. (a) Complete the coding of the following class (which is obviously intended to implement complex numbers).

```
typedef class Complex {
  public:
    Complex();
    Complex( double x, double y );
    Complex operator + ( Complex other );
    Complex operator * ( Complex other );
    void print();

    double re, im;
} Complex;
```

(b) Write a full C++ program which reads lines of input and prints those lines which contain the word `"finished"`. It should use <`string`>. Note that `string::npos` is returned if `string::find()` does not find an occurrence of a given string.