King's College London

UNIVERSITY OF LONDON

This paper is part of an examination of the College counting towards the award of a degree. Examinations are governed by the College Regulations under the Authority of the Academic Board.

M.Sci. EXAMINATION

CP/4731 The C and C++ Programming Languages

Summer 2003

Time allowed: THREE Hours

Candidates must answer any THREE questions. No credit will be given for attempting a further question.

The approximate mark for each part of a question is indicated in square brackets.

Good answers to questions will include plans and explanations in addition to sections of C or C++ code.

Some questions specify C or C++. Marks will be lost if you use C++ syntax in the solutions to questions that specify C.

You must not use your own calculator for this paper.
Where necessary, a College calculator will have been supplied.

TURN OVER WHEN INSTRUCTED

Answer THREE questions

1) Given a general cubic equation

$$ax^3 + bx^2 + cx + d = 0 \qquad (1.1)$$

where *a, b, c,* and *d* are constants, write a program in C which reads in the coefficients, and prints out the three roots.

[20 marks]

Use this method:
The first step is to evaluate the discriminant, *D*:

$$D = \left(\frac{p}{3}\right)^3 + \left(\frac{q}{2}\right)^2$$
$$\text{where } p = \frac{1}{3}\left(\frac{3c}{a} - \frac{b^2}{a^2}\right), \qquad q = \frac{1}{27}\left(\frac{4b^3}{a^3} - \frac{9bc}{a^2} + \frac{27d}{a}\right) \qquad (1.2)$$

If $D > 0$, there is one real root and two complex roots. If $D = 0$, there are three real roots of which at least two are equal. If $D < 0$, there are three real roots.

If $D \geq 0$, define:

$$u = \left(\frac{-q}{2} + D^{1/2}\right)^{1/3}, \quad v = \left(\frac{-q}{2} - D^{1/2}\right)^{1/3}$$
$$y_1 = u + v, \quad y_{2,3} = \frac{-(u+v)}{2} \pm i\frac{(u-v)\sqrt{3}}{2} \qquad (1.3)$$

If $D < 0$, define:

$$\phi = \arccos\left(\frac{-q}{2}\left[\frac{p}{3}\right]^{-3/2}\right)$$
$$y_1 = 2\left(\frac{|p|}{3}\right)^{1/2}\cos\frac{\phi}{3}, \quad y_{2,3} = -2\left(\frac{|p|}{3}\right)^{1/2}\cos\frac{\phi \pm \pi}{3} \qquad (1.4)$$

Finally, the three roots of equation 1.1, $x_1, x_2,$ and x_3 are given by:

$$x_n = y_n - \frac{b}{3a}. \qquad (1.5)$$

2) Simpson's method is used to evaluate integrals numerically.

$$I = \int_a^b f(x)dx \qquad (2.1)$$

For any function, $f(x)$, the interval over which the integral is to be evaluated ($a<x<b$) is divided into N small elements, where N is large. The integrand must be evaluated at each of those elements ($x_0 = a, x_1, x_2, x_3, \cdots, x_N = b$). Now define:

$$\Delta I_i = \left(\tfrac{1}{3} f(x_{2i}) + \tfrac{4}{3} f(x_{2i+1}) + \tfrac{1}{3} f(x_{2i+2})\right)\frac{(b-a)}{N} \qquad (2.2)$$

$$I = \sum_{i=0}^{N/2-1} \Delta I_i \qquad (2.3)$$

It is possible to use this method to estimate the integral of a function $f(x)$ which is stored digitally as a series of $N+1$ equally spaced values.

Such a digital function is stored in a file called `data.d` as a series (initially of unknown length) of ASCII floating point numbers separated by spaces, preceded by the values of the first and last x values. (That is, the values stored will be: $a, b, f(x_0 = a), f(x_1), f(x_2), \cdots, f(x_N = b)$)

Write a program in C or C++ which reads in and stores the numbers, allocating just sufficient space for them, and calculates, using Simpson's method, the integral over the given range.

[20 marks]

3) Explain why a C (or C++) function with prototype `int factorial(int n);` intended to calculate $n!$ would work for only a limited range of values of n.

[3 marks]

The binomial coefficients $\binom{l}{m}$ with $l > m$ are given by:

$$\binom{l}{m} = \frac{l!}{m!(l-m)!} \qquad (3.1)$$

Write a function in C which calculates the binomial coefficients $\binom{l}{m}$.

[7 marks]

An expression for the Legendre polynomial, $P_l(x)$ is:

$$P_l(x) = 2^{-l} \sum_{m=0}^{l/2} (-1)^m \binom{l}{m} \binom{2l-2m}{l} x^{l-2m} \qquad (3.2)$$

Write a further function in C which calculates $P_l(x)$ for any values of l and x.

[10 marks]

4) A C++ class of matrices has the following definition:

```
class matrix
{
private:
      int N, M;
      float **ptr;
public:
      matrix(int, int);
      matrix(const matrix&);
      ~matrix();
      void setElement(int,int,float);
      float getElement(int,int);
};
```

Explain what is done by each of the functions declared in the class definition. Also explain what the terms `private:` and `public:` mean.

[5 marks]

Write the C++ code for the constructor function, such that it sets up and allocates space for an N × M matrix (default 2×2) and sets all the elements to zero initially.

[5 marks]

Write the C++ code for the copy constructor, such that its operation is consistent with the constructor.

[3 marks]

Set up a new class of matrices, derived from class matrix, such that both functions `setElement` and `getElement` (in the new class) check that the indices of the required element are not outside the array bounds of the matrix object accessed. Include the code for these two new functions.

[7 marks]

[You are required to write the code only for the functions specified.]

5) Explain what is meant by the C++ term "overloading". How are different overloaded functions distinguished by the C++ compiler?

[3 marks]

Assuming that classes of 3x3 matrices and vectors (of length 3) are already defined, and called `matrix` and `vector` respectively, write the code for the overloaded operator `<<` to allow the elements of a matrix or a vector to be printed out. State any assumptions that need to be made.

[7 marks]

Write C++ code to overload the `*` operator to allow matrix multiplication between a matrix and a (column) vector with the result expressed as a (column) vector.

[10 marks]