King's College London

UNIVERSITY OF LONDON

B.Sc. EXAMINATION

CP/1710 Computing for Physical Sciences

Summer 2001

Time allowed: THREE Hours

Candidates must answer SIX parts of SECTION A, and TWO questions from SECTION B.

Separate answer books must be used for each Section of the paper.

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

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

TURN OVER WHEN INSTRUCTED

Where necessary, you may assume that the following *C* functions are defined in the library `<math.h>`

```
double sqrt( double x); // returns the square root of x
double sin( double x);  // returns the sine of x (x in radians)
double exp( double x);  // returns the exponential of x
```

SECTION A – Answer SIX parts of this section

1.1) Distinguish between the rôles of the *pre-processor*, the *compiler* and the *linker* when preparing an executable *C* program.

[7 marks]

1.2) Inside a *C* function some variables are declared with the following statements:

```
short int a=4;
long int b=4;
const int c=4;
static int d=4;
```

Distinguish clearly the consequences of these different declarations for the way in which the value 4 is stored in each of the variables `a, b, c, d`.

[7 marks]

1.3) Describe the output produced by the following short *C* program.

```
int main()
{  int i;
   int a[6]={2,4,6,8,10,12};
   for (i=5; i >= 0; i--)
   {  printf( "a[%d] = %2d \n", i, a[i]);
   }
   return 0;
}
```

[7 marks]

1.4) Describe the difference in behaviour between the following two fragments of *C* code.

Fragment (i)

```
{  int i = -10;
   while (i>0) printf("Time to launch: %d s \n", i--);
}
```

Fragment (ii)

```
{  int i = -10;
   do printf("Time to launch: %d s \n", i--); while (i>0);
}
```

[7 marks]

SEE NEXT PAGE

1.5) Write a function in *C* with two integer arguments a and b that will return the value zero when b is not a divisor of a, and will return the value a/b when b is a divisor of a. Ensure that your function will return zero for the case $b = 0$.

[7 marks]

1.6) A structure of type `record_temp` is defined as shown in the *C* code fragment below.

```
struct record_temp
{  int date;
   char month[4];
   int year;
   char weekday[4];
   float temperature;
};
```

Provide a few additional lines of *C* code that will:

(i) declare a variable `london` that is a structure of this type, initialised so that the members of the structure store the values `22, "May", 2001, "Tue"` and `25.3` respectively,

(ii) print out the member values as a single line of text, in the format shown here.

```
Date: Tue 22 May 2001, Temperature: 25.3 C
```

[7 marks]

1.7) Explain briefly why it can be helpful to use pointer variables as the arguments to *C* functions.

[7 marks]

1.8) A *C* program initialises the variables `a, b, c` and `x` in the following way.

```
int a=3, *b;
b=&a;
float c=7.0, x=4.0;
```

What are the values of `x` after each of the *C* statements below have been executed?

```
(i)      x = 1/a + c;
(ii)     x -= c - a;
(iii)    x = c % (*b);
(iv)     x = ceil(c/3.0);
```

[7 marks]

Note: Treat each part of this question independently, the statements are not consecutive steps in the same program.

SECTION B – Answer TWO questions

2) In a counting experiment a set of integers, with values in the range 0 to 999, is recorded, and a statistical analysis of the data is required. Write a *C* program that will allow the user to carry out the tasks listed below. Ensure that your *C* code contains sufficient comments for the reader to understand how each of the requirements has been met.

(i) Repeatedly prompt the user to enter an integer reading of the count, until the user indicates the end of the data set by entering a negative integer.

[4 marks]

(ii) Issue a warning message if a data value of 1000 or greater is entered by the user, and ignore this value in any statistical analysis.

[2 marks]

(iii) Calculate the mean value $\bar{x}$ and standard deviation σ of the N valid data values entered. Use the formulae

$$\bar{x} = \frac{1}{N}\sum_{n=1}^{N} x_n, \qquad \text{and} \qquad \sigma = \sqrt{\left(\frac{1}{N}\sum_{n=1}^{N} x_n^2\right) - (\bar{x})^2}$$

where x_n denotes an individual reading.

[6 marks]

(iv) Count the number of input values that lie in each century range 0...99, 100...199, 200...299 and so on up to 900...999.

[6 marks]

(v) Print out to the computer screen a single-line statement giving the mean and standard deviation of the data, and the total number of readings used.

[2 marks]

(vi) Print out to the screen the frequency distribution of readings, using one line for each century range, ensuring that the numerical information is displayed in a neatly aligned column.

[4 marks]

Describe briefly any important choices you had to make in designing the program.

[6 marks]

3) The function

$$\texttt{sinc}(x) = \frac{\sin(x)}{x}$$

is encountered in mathematical physics. Write a function in *C* to return the value of $\texttt{sinc}(x)$ for any real value of x that is supplied as an argument to the function. Comment in detail on how any special cases of the argument x are handled by your function.

[10 marks]

It can be shown that the function $\texttt{Si}(x)$, the integral of $\texttt{sinc}(x)$, can be approximated by the first few terms of an infinite series of the form

$$\texttt{Si}(x) = \int_0^x \texttt{sinc}(u)\,du = x - \frac{x^3}{3 \times 3!} + \frac{x^5}{5 \times 5!} - \frac{x^7}{7 \times 7!} + \frac{x^9}{9 \times 9!} \dots$$

Write another *C* function that will use the first five terms in the series above to calculate an approximate value for $\texttt{Si}(x)$. Explain carefully the main features of the *C* function you have written.

[15 marks]

Calculate an approximate range of x values for which you would expect your *C* function to give the value of $\texttt{Si}(x)$ correct to 2 decimal places.

[5 marks]

4) Describe the **bisection** method for locating a real root of the equation $f(x) = 0$ for x in the interval from x_0 to x_1, where $x_0 < x_1$, and $f(x)$ is a continuous real function, with $f(x_0) < 0$ and $f(x_1) > 0$.

[6 marks]

Explain briefly why the bisection method is said to converge *linearly*. Calculate the number of iterations needed to locate the root of $f(x) = 0$ to an accuracy of 4 decimal places, assuming that $x_1 - x_0 = 10$.

[10 marks]

Determine a suitable interval over which to find the root of the equation

$$\exp(x) + x = 0$$

[2 marks]

Write a *C* program to find, and print out, the root of the equation to an accuracy of 4 decimal places.

[12 marks]

5) A set of experimental data is stored as formatted values in a plain text file called `rawdata.txt`. The data consist of 1000 real numbers, stored as 1000 lines of text, with one value on each line. The data represent measurements of a slowly-changing voltage that has been corrupted by rapidly-changing noise. Write a *C* program that will carry out the following tasks:

   (i) Read the noisy data from the specifed file into a one-dimensional array named `input`.

   (ii) Apply a moving-average filter to the data. The filter operation should create a new array named `output` in which the first and last elements of the array are the same as the `input` data, but every other element of `output` is calculated according to the *C* expression

```
output[n] = (input[n-1] + input[n] + input[n+1])/3.0;
```

   where `n` is an integer array index.

   (iii) Write out the filtered data to a file called `filterdata.txt`, using the same format as for `rawdata.txt`.

[15 marks]

Discuss briefly the advantages and disadvantages of applying this type of moving-average filter to experimental data.

[9 marks]

Describe briefly the effects of using a 5-point moving-average filter instead of the 3-point filter described above. What modifications to your *C* code would be needed to make this change?

[6 marks]