G 535

Ollscoil na hÉireann, Gaillimh
National University of Ireland, Galway
Summer Examinations, 2004/2005

Exam Code(s)	1BS1, 1CS1, 1EH1, 1EL1, 1ER1
Exam(s)	First Science
Module Code(s)	CS102
Module(s)	Computer Science
Paper No	1
Repeat Paper	
Special Paper	
External Examiner(s)	Professor D. Bell
Internal Examiner(s)	Professor G. Lyons
	Professor T.N. Sherry
	Dr. R.F. Butler
	Dr. G. Pfeiffer
	Mr. A. Power
Instructions:	Answer *SIX* questions, at least *ONE* from each section. Use a *SEPARATE ANSWER BOOK* for each section. All questions carry the same marks.
Duration	3 hours
No. of Answer books	3
Requirements:	
Handout	
MCQ	
Statistical Tables	
Graph paper	
Log Graph Paper	
Other Material	
No. of Pages:	5
Department(s):	Mathematics, Experimental Physics
	Information Technology, Mathematical Physics

Section A

1. Answer each of the following *four* parts:

   (a) Convert the decimal values 57_{10} and 161_{10} into their binary number form.

   (b) Convert the decimal fractions 0.328125 and 0.7734375 into their binary number form. Show your calculations in full.

   (c) Using the usual ANSI/IEEE 754-1985 standard for single-precision real (floating point) numbers, what would be the representation of the decimal value 97.4375_{10}?

   (d) Construct a Huffman tree and its associated coding table, for a text document which uses a total of 10 different characters, at various frequencies of your own choosing. For each character, compare the number of bits used in its Huffman code to the number used in its ASCII code, and hence compute the average saving in transmitted bits. *Note*: you do not need to know the actual ASCII codes.

2. Answer each of the following *three* parts:

   (a) Using truth tables, prove the following logical identities:

$$\overline{P+Q} = \overline{P} \cdot \overline{Q}$$

$$P + (Q \cdot R) = (P+Q) \cdot (P+R)$$

   Note: you must show each individual logical step as a separate column in the truth tables.

   (b) Draw an electronic circuit which implements the following Boolean expression:

$$A \cdot B + A \cdot (\text{NOT } (B)) + A \cdot (\text{NOT } (A))$$

   (c) Discuss the logical design and operation of any two of the following electronic circuits: *(i)* half adder, *(ii)* 1:4 demultiplexor, *(iii)* astable.

3. Answer each of the following *three* parts:

   (a) Illustrate the procedure by which the Fetch/Execute cycle is implemented by the CS1 machine/simulator.

   (b) Explain the meaning and operation (in terms of <u>all</u> registers which are affected) of <u>any 8</u> of the following 10 machine code operations, used in the CS1 machine/simulator:

   1. JZA 2. JPA 3. JNA 4. LDX 5. STX
   6. ADX 7. SBX 8. SJX 9. JZX 10. DJX

   (c) Write a program in CS1 machine code which reads in two numbers from a text file, and writes their product and quotient out to a text file. You may wish to refer to the table of CS1 operation codes given below.

01	Read	02	Write	03	Load	04	Store
05	Add	06	Subtract	07	Multiply	08	Divide
09	Jump	10	JZA	11	JPA	12	JNA
13	LDX	14	STX	15	ADX	16	SBX
17	SJX	18	JZX	19	DJX	20	Halt

Section B

4. Answer each of the following *three* parts:

   (a) Explain each of the following terms:
   (i) *identifier*, (ii) *comment*, (iii) *variable*, (iv) *expression*, and (v) *statement*.

   (b) Explain the concepts of precedence and associativity of operators in a C program. Describe the precedence and associativity of +, -, * and =.

   (c) Indicate the order of evaluation of the operators in each of the following C statements, and determine the resulting value of x:

   i. x = 7 + 3 * 6 / 2 - 1;
   ii. x = 2 % 2 + 2 * 2 - 2 / 2;

5. Answer each of the following *two* parts:

   (a) Write a statement or comment, as appropriate, to accomplish each of the following:

   i. State that a program will calculate the product of three integers.
   ii. Define the four variables x, y, z and *product* to be of type **int**.
   iii. Prompt the user to enter three integers, please.
   iv. Read three integers from the keyboard and store them in variables x, y, and z.
   v. Compute the product of x, y, and z and assign the result to the variable *product*.
   vi. Print "The product is " followed by the value of the integer variable *product*.

   (b) Describe the process of function invocation in a C program.

6. Answer each of the following *two* parts:

   (a) Write a C program that, depending on the definition of N, prints a triangular shape of N lines of stars. If N is 3 the output should be:

```
  *
 **
***
```

   (b) Using suitable while loops, for loops, or functions, write a C program that, depending on the definition of N, prints 2*N+1 lines of stars and blanks in a diamond shape. If N is 2 the output should be:

```
  *
 ***
*****
 ***
  *
```

   Describe the important parts of your program and explain, briefly, why the program does the right thing.

Section C

7. Answer each of the following *three* parts:

   (a) Explain the difference between the following types of memory storage:
   Random Access Memory, Read Only Memory, Cache memory and Flash memory.

   (b) How does a computer with one Central Processing Unit appear to run multiple computer applications at the same time?

   (c) What do the following acronyms stand for and what are they used for in computer technology: ASCII, XML and TCP.

8. Answer each of the following *four* parts:

   (a) Explain *briefly* what a pointer is in C and how it is declared. Illustrate in the case of a pointer to a float variable.

   (b) The following code is part of a C programme:

```
char a, b, c, *m, *n, *q;
a = 'J';
m = &a;
b = '*';
n = &c;
c = 'a';
q = m;
*m = c + 10;
*n = *q - 32;
```

   After the code has executed, what values will a, b, c, *m, *n and *q have? Explain how you arrive at your answers - you may find a flow diagram helpful.

   (c) Explain how C deals with pointers to arrays.

   (d) Three dimensional vectors **b** and **c** are represented by simple arrays, b and c respectively, containing three integer elements. The vector **a** is defined by the equation $\mathbf{a} = 2\mathbf{b} + 3\mathbf{c}$ .The following programme looks for two vectors from the user, calculates the vector **a** and outputs the result to the screen:

```
# include <stdio.h>
main(void)
{
    int a[3], b[3], c[3];
    void inputvector( int * );
    void addvectors( int *, int *, int * );
    void printvector( int * );
    printf("First Vector: ");
    inputvector(b);
    printf("Second Vector: ");
    inputvector(c);
    addvectors( a, b, c );
    printvector( a);
}
```

   Write, and comment, the three function definitions for inputvector, addvectors and printvector.

9. Answer each of the following *four* parts:

   (a) Explain briefly how user-defined structured variables can be useful and give two examples.

   (b) Explain the significance of each part of the following C code:

```
typedef struct
{
    int re;
    int im;
}  complex;
```

   (c) A fraction is the quotient of two integers

$$\frac{\text{num}}{\text{den}} .$$

   Fractions can be treated in C using an appropriately defined structured variable type. Write a commented programme, where the entire programme is contained in the main function, to output the difference of two fractions, given the subtraction algorithm

$$\frac{\text{num}_1}{\text{den}_1} - \frac{\text{num}_2}{\text{den}_2} = \frac{\text{num}_1 * \text{den}_2 - \text{num}_2 * \text{den}_1}{\text{den}_1 * \text{den}_2} .$$

   (d) Re-write the commented programme of (c) above so that the subtraction of the fractions is carried out by the function

```
void subtract (fraction *x, fraction *y, fraction *z);
```