COMPUTER SCIENCE TRIPOS Part IB

Monday 4 June 2007 1.30 to 4.30

PAPER 3

*Answer **five** questions.*

*Submit the answers in five **separate** bundles, each with its own cover sheet. On each cover sheet, write the numbers of **all** attempted questions, and circle the number of the question attached.*

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

STATIONERY REQUIREMENTS	SPECIAL REQUIREMENTS
Script paper	*None*
Blue cover sheets	
Tags	

1 Economics and Law

(*a*) What are the first and second theorems of welfare economics? [5 marks]

(*b*) Give *three* examples of ways in which markets can fail to reach competitive equilibrium when the assumptions of the first theorem do not hold, discussing the implications for the information goods and services industries in each case. [15 marks]

2 Probability

(*a*) The notation $\binom{n}{r}$ may be interpreted as the number of ways of choosing r items from n. Give an elementary proof of Pascal's theorem that $\binom{n}{r} = \binom{n-1}{r-1} + \binom{n-1}{r}$ given $n \geqslant 1$ and $0 < r < n$. The proof need not be very formal but do not exploit the representation that makes use of factorials. [4 marks]

(*b*) A College computer officer discovers that two of the workstations in a computer room have dirty keyboards. All the other keyboards are clean. No one else is in the room at the time of her inspection but she notices that four students enter the room just as she is leaving to fetch her cleaning materials.

If all four students keep clear of the dirty keyboards the computer officer won't have to disturb anyone on her return. Assuming that the students choose workstations at random she determines the probability a that all four happen to sit at clean keyboards. She also determines the probability b that both the workstations with dirty keyboards are amongst the four chosen. She is intrigued to note that $a = b$.

(*i*) Assuming that there are n workstations in the room, show that

$$a = \frac{n-4}{n} \cdot \frac{n-5}{n-1}$$ [4 marks]

(*ii*) Likewise express the probability b in terms of n. [10 marks]

(*iii*) By equating a and b determine the number of workstations in the computer room. [2 marks]

3 Floating-Point Computation

(*a*) A hypothetical (and practically rather useless) floating-point number representation inspired by the IEEE floating-point standards uses 6 bits—one bit for sign, three bits for exponent and two (stored) bits for mantissa (significand). Assuming that 1.0 is represented in this format as 0:011:00 give the values, in decimal notation (fractions are acceptable), of all the other non-negative floating-point values in this representation. You do not need to give details of denormalised numbers or NaNs. [10 marks]

(*b*) Explain the following terms:

(*i*) absolute error;

(*ii*) relative error;

(*iii*) rounding error;

(*iv*) truncation error;

(*v*) ill-conditionedness. [5 marks]

(*c*) Assuming the floating-point representation for type `float` has b bits in its mantissa (significand), what can be said about the output of the following program?

```
float f = 10.0/3.0;
for (i=0; i<100; i++)
{ int v = (int)f;       /* get integer part of f */
  printf("%d\n", v);    /* print it */
  f = (f - v) * 10;
}
```

Discuss how accurately `f` represents 10.0/3.0 at the start of each iteration and explain which operation(s) represent the main loss of accuracy in `f` on each iteration. (You may assume that 10 significant bits of accuracy is approximately 3 decimal digits of accuracy.) [5 marks]

4 Programming in C and C++

A C programmer is working with a little-endian machine with 8 bits in a byte and 4 bytes in a word. The compiler supports unaligned access and uses 1, 2 and 4 bytes to store `char`, `short` and `int` respectively. The programmer writes the following definitions (below right) to access values in main memory (below left):

Address	Byte offset			
	0	1	2	3
0x04	10	00	00	00
0x08	61	72	62	33
0x0c	33	00	00	00
0x10	78	0c	00	00
0x14	08	00	00	00
0x18	01	00	4c	03
0x1c	18	00	00	00

```
int **i=(int **)0x04;

short **pps=(short **)0x1c;

struct i2c {
  int i;
  char *c;
}*p=(struct i2c*)0x10;
```

(*a*) Write down the values for the following C expressions:

```
**i        p->c[2]        &(*pps)[1]        ++p->i
```

[8 marks]

(*b*) Explain why the code shown below, when executed, will print the value 420.

```
#include<stdio.h>

#define init_employee(X,Y) {(X),(Y),wage_emp}
typedef struct Employee Em;
struct Employee {int hours,salary;int (*wage)(Em*);};
int wage_emp(Em *ths) {return ths->hours*ths->salary;}

#define init_manager(X,Y,Z) {(X),(Y),wage_man,(Z)}
typedef struct Manager Mn;
struct Manager {int hours,salary;int (*wage)(Mn*);int bonus;};
int wage_man(Mn *ths){return ths->hours*ths->salary+ths->bonus;}

int main(void) {
  Mn m = init_manager(40,10,20);
  Em *e= (Em *) &m;
  printf("%d\n",e->wage(e));
  return 0;
}
```

[4 marks]

(*c*) Rewrite the C code shown in part (*b*) using C++ primitives and give *four* reasons why your C++ solution is better than the C one. [8 marks]

5 Computer Graphics and Image Processing

(*a*) In image compression we use three different mechanisms to compress pixel data:

(*i*) mapping the pixel values to some other set of values;

(*ii*) quantising those values;

(*iii*) symbol encoding the resulting values.

Explain each mechanism, describe the way in which it helps us to compress the image, and describe how the mechanism is implemented in the baseline JPEG compression method. [10 marks]

(*b*) Describe the limitations of human vision in terms of:

(*i*) spatial resolution,

(*ii*) luminance,

(*iii*) colour,

and explain the implications that each of these has on the design of display devices, including numerical estimates of the limits beyond which a human cannot discriminate. [10 marks]

6 Mathematical Methods for Computer Science

Let

$$b_a(x) \;=\; \begin{cases} 1 & \text{for } |x| \leq a \\ 0 & \text{for } |x| > a \end{cases}$$

where a is a constant such that $0 < a \leq \pi$.

(*a*) Find the Fourier Transform, $F_a(\omega)$, of $b_a(x)$. [6 marks]

(*b*) Suppose that $f(x)$ is some 2π-periodic function with complex Fourier coefficients, c_n, for $n = 0, \pm 1, \pm 2, \ldots$

(*i*) State an expression for c_n, for $n = 0, \pm 1, \pm 2, \ldots$ [2 marks]

(*ii*) Show that $c_n = G(n)$ for $n = 0, \pm 1, \pm 2, \ldots$ where the function $G(\omega)$ is the Fourier transform of $f(x)b_\pi(x)$. [6 marks]

(*c*) Now suppose that $f(x)$ is the 2π-periodic function defined such that $f(x) = b_a(x)$ for $|x| \leq \pi$. Find the complex Fourier coefficients, c_n, for this choice of the function $f(x)$ using your result derived in part (*b*)(*ii*). [6 marks]

7 Computation Theory

(*a*) (*i*) Define the notion of a *register machine* and the computations that it carries out. [5 marks]

(*ii*) Explain, in general terms, what is meant by a *universal* register machine. (You should make clear what scheme for coding programs as numbers you are using, but you are not required to describe a universal register machine program in detail.) [5 marks]

(*b*) (*i*) Explain what it means for a partial function f from $\mathbb{N}$ to $\mathbb{N}$ to be *computable* by a register machine. [2 marks]

(*ii*) Let $n > 1$ be a fixed natural number. Show that the partial function from $\mathbb{N}$ to $\mathbb{N}$

$$f_n(x) = \begin{cases} nx & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$$

is computable. [3 marks]

(*iii*) Explain why there are only countably many computable functions from $\mathbb{N}$ to $\mathbb{N}$. Deduce that there exists a partial function from $\mathbb{N}$ to $\mathbb{N}$ that is not computable. (Any standard results you use about countable and uncountable sets should be clearly stated, but need not be proved.) [3 marks]

(*iv*) If a partial function f from $\mathbb{N}$ to $\mathbb{N}$ is computable, how many different register machine programs are there that compute f? [2 marks]

8 Artificial Intelligence I

We have a basic search problem, consisting of a set S of states, a start state s_0, a goal test $G(s)$ that returns True if $s \in S$ is a goal and False otherwise, and a function $\exp(s)$ that returns the set of states obtained by expanding state s.

(*a*) Describe in detail the *Graph Search* algorithm for solving a problem of this type. How does it differ from the related *Tree Search* algorithm? [8 marks]

(*b*) Give a detailed description of the *Recursive Best First* search algorithm, and explain why it might be used in preference to the $A^\star$ algorithm. [12 marks]

9 Introduction to Security

(*a*) You have received a shipment of hardware random-number generators, each of which can output one 128-bit random number every 10 milliseconds. You suspect that one of these modules has been tampered with and that it actually produces only 30-bit random numbers internally, which are then converted via a pseudo-random function into the 128-bit values that it outputs.

(*i*) How does this form of tampering reduce the security of a system that uses a generated 128-bit random number as the secret key of a block cipher used to generate message authentication codes? [2 marks]

(*ii*) Suggest a test that has a more than 0.5 success probability of identifying within half an hour that a module has been tampered with in this way. [6 marks]

(*b*) Explain briefly

(*i*) the encryption and decryption steps of Cipher Feedback Mode; [3 marks]

(*ii*) why some operating systems ask the user to press a special key combination (e.g., Alt-Ctrl-Del) before each password login; [3 marks]

(*iii*) how a secure hash function can be used to implement a one-time signature scheme; [3 marks]

(*iv*) what happens if the same private key of the scheme from (*iii*) is used *multiple times*, to sign different messages. [3 marks]

END OF PAPER