Advanced Computer Architecture
MEng3 Test

Tuesday 12th December

Answer both *questions*
You have one-and-half hours

1 This question concerns the rationale behind Intel's IA-64 instruction set architecture (ISA). Where necessary, assume a processor architecture such as Intel's Itanium, which implements this instruction set using 6-wide parallel instruction issue, and a 10-stage static (in-order) pipeline.

a the IA-64 ISA includes 64 1-bit "predicate" registers, which can hold the result of a Boolean comparison. Almost all IA-64 instructions can be predicated - that is, whether the instruction actually takes effect depends on whether a named predicate register is set to 1. For example,

```
if (a==0)                        cmp.eq p1 = 0,a;; // if a=0, p1=1 else p1=0
  z = 8;     is implemented as:  (p1) mov z = 8    // if p1 is set to 1, move 8 to z
```

Suppose

- a correctly-predicted branch instruction takes effectively no time to execute,
- an incorrectly-predicted branch incurs a misprediction penalty of 10 cycles,
- the branch prediction accuracy is 50%
- the body of the `if-then` construct consists of N instructions

What is the maximum value of N for which this predicated approach could be faster than using a conditional branch?

b Consider the following instruction sequence:

```
if (a==1)           cmp.ne p1 = a,1;;
  a=0;              (p1) br.cond label1;;   // Branch A
                    mov a=0;;
if (b==1)   label1: cmp.ne p1 = b,1;;
  b=0;              (p1) br.cond label2;;   // Branch B
                    mov b=0;;
if (a!=b)   label2: cmp.ne p1 = a,b;;
  f();              (p1) br.call f;;        // Branch C
```

Suppose that `a` is a random number, either 0 or 1 with equal probability. Suppose that `b` is practically always 1.

(i) Suppose the processor uses a 2-bit bimodal branch predictor. What branch prediction accuracy would you expect for each of the three conditional branches (A, B and C)?

(ii) Suppose the processor uses a (2,2) correlating branch predictor. What branch prediction accuracy would you expect for each of the three conditional branches?

(iii) Use a diagram to illustrate how the (2,2) branch predictor arrives at a prediction for the third conditional branch.

(iv) Suppose the first two branches were avoided using predication. What branch prediction accuracy would you expect for the remaining (last) conditional branch?

(v) Explain how predication can make the prediction accuracy of *other* branches *worse*

(vi) Explain how predication can *improve* the prediction accuracy of *other* branches

(The seven parts carry, respectively, 15%, 15%, 15%, 25%, 10%, 10% and 10% of the marks).

2 In this question, consider the following loop:

```
declare float U[0:M, 0:N]

for t = 1 to M do
  for i = 1 to N-1 do
S:  U[t,i] = (U[t-1,i-1] + U[t-1,i+1]) * 0.5
```

a Suppose `M=4` and `N=6`. Draw the iteration space graph for the loop. Mark on the graph all the dependences present.

b Write down all the dependences present in the loop. For each dependence, indicate whether it is a data-dependence or an anti-dependence, whether it is loop carried, and write down its dependence distance vector.

c Consider the unimodular transformation matrix

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Draw the iteration space for the loop above after this transformation has been applied. Show the dependences.

d Write down the dependence distance vectors for the transformed loop.

e Can the transformed loop be tiled? Justify your answer with reference to your iteration space graph.

f How many registers would a processor need in order to calculate `U[4,0:N]` with the minimum number of memory accesses?

g Is this a good implementation for processor with a pipelined floating-point unit? Justify your answer.

(The seven parts carry, respectively, 15%, 20%, 15%, 10%, 10%, 20% and 10% of the marks).

End of Paper