OLLSCOIL NA hÉIREANN
THE NATIONAL UNIVERSITY OF IRELAND

COLÁISTE NA hOLLSCOILE, CORCAIGH
UNIVERSITY COLLEGE, CORK

2015/2016

Semester 2 – Summer 2016

CS2507 (Sample)

Computer Architecture

Dr Helen Purchase

Professor Cormac Sreenan

Mr Humphrey Sorensen

1.5 Hours

Answer All Questions

Paper Total: 80 Marks

PLEASE DO NOT TURN THIS PAGE UNTIL INSTRUCTED TO DO SO

PLEASE ENSURE THAT YOU HAVE THE CORRECT EXAM PAPER

1. Architectural & Performance Issues

(a) One commonnly-applied "great idea" in computer architecture is *Performance via Prediction*. Specify any practical example of this principle in action. *[4 Marks]*

(b) Define the term *CPI* and state its role in measuring processor performance *[4 Marks]*

(c) A common approach to improving processor speed is *over-clocking* – increasing the clock speed of the processor. What limitations does this approach have? *[4 Marks]*

2. Number Representation & Processing

(a) What is *2s Complement integer representation* and what benefits does this representation have? *[4 Marks]*

(b) When using the IEEE 754 standard for floating-point representation a single precision number uses a sign bit, 8 bits for the exponent and 23 bits for the fraction [32 bits in total]; for double precision numbers, it uses a sign bit, 11 bits for the exponent and 52 bits for the fraction [64 bits in total]. Why are the extra bits mainly added to the fraction – rather than being evenly distributed among fraction and exponent? *[4 Marks]*

(c) Addition of integer numbers can be achieved in a single clock cycle. Is the same true for floating point numbers? Why? *[4 Marks]*

3. Instruction Set Architecture (ISA) & Assembly Language Programming

(a) The MIPS processor supports *Type-R* and *Type-I* instructions. How do they differ? Give an example of each. *[4 Marks]*

(b) Specify an instruction that increments the (numeric) content of a register by 1 as efficiently as possible? Can you adapt this to decrement the register by 1? *[4 Marks]*

(c) MIPS employs *indirect addressing* for memory references, whereby it adds a constant to the contents of a register to generate the memory address. Give an instruction that illustrates this. What benefit is there to this addressing technique? *[4 Marks]*

(d) For the following C instructions, write corresponding MIPS assembly language fragments (stating any assumptions you make regarding memory/register allocation):

i. `f = 2 * f;` *[4 Marks]*

ii. `B[8] = A[8];` *[4 Marks]*

iii. *[8 Marks]*

```
while i > 0
  { count = count + i;
    i = i – 1;   }
```

(e) With respect to procedure invocation in MIPS, answer the following: *[4 Marks each]*

i. How and why are registers saved on procedure entry?

ii. How and why are registers restored on procedure exit?

iii. Do all registers have to be saved and restored in this way?

4. Memory Hierarchy

(a) Caching:

(i) Why is cache memory small and fast, while main memory [RAM] is, relatively speaking, large and slow? How does the underlying technology dictate that this is the case? *[4 Marks]*

(ii) Caching strategies are designed to exploit the *principle of locality*. What is this and how is it exploited? *[4 Marks]*

(iii) When a program first begins execution, the use of cache might retard its execution speed. When a program has been executing for some time, he use of cache should speed up its execution. Why? *[4 Marks]*

(b) How does *set associative* cache differ from *fully associative*? *[4 Marks]*