ENGINEERING TRIPOS   PART IIA

Wednesday 24 April 2013   9.30 to 11

Module 3F5

COMPUTER AND NETWORK SYSTEMS

Answer not more than **three** *questions.*

All questions carry the same number of marks.

The **approximate** *percentage of marks allocated to each part of a question is indicated in the right margin.*

There are no attachments.

STATIONERY REQUIREMENTS   SPECIAL REQUIREMENTS

Single-sided script paper   Engineering Data Book

                                                             CUED approved calculator allowed

> **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**

FINAL version

1   (a)   In the context of parallel processing, define and very briefly explain the following acronyms: SIMD, MIMD, SMP, UMA, NUMA, SMT.   [20%]

(b)   Figure 1 shows schematics of two MIMD machines. Explain why smaller MIMD machines tend to be laid out like Machine A, while larger MIMD machines resemble Machine B.   [20%]

(c)   Machine A employs bus snooping to enforce cache coherency. Explain why a *write invalidate* protocol is usually preferred to a *write update* protocol.   [20%]

(d)   An application running on a quad-core example of Machine A requires the scalar product of two large vectors x and y. Since this calculation is the rate-determining step, the programmer decides to parallelize it. Specifically, each of the four cores runs, concurrently, the code segment in Fig. 2. For core 1, the integer variable me is set to 0: this core therefore deals with elements 0 to 2,499,999 of the vectors, storing the partial sum in partial_sum[0]. For core 2, me is set to 1: this core therefore deals with elements 2.500,000 to 4,999,999 of the vectors, storing the partial sum in partial_sum[1]. And so on for cores 3 and 4. In a later section of unthreaded (uniprocessor) code, the four elements of partial_sum are added together to give the scalar product.

Explain carefully why the code, as written, might not achieve anything close to a 4× speedup. Suggest how the programmer (or a smart compiler) might optimize the code to improve its performance.   [40%]
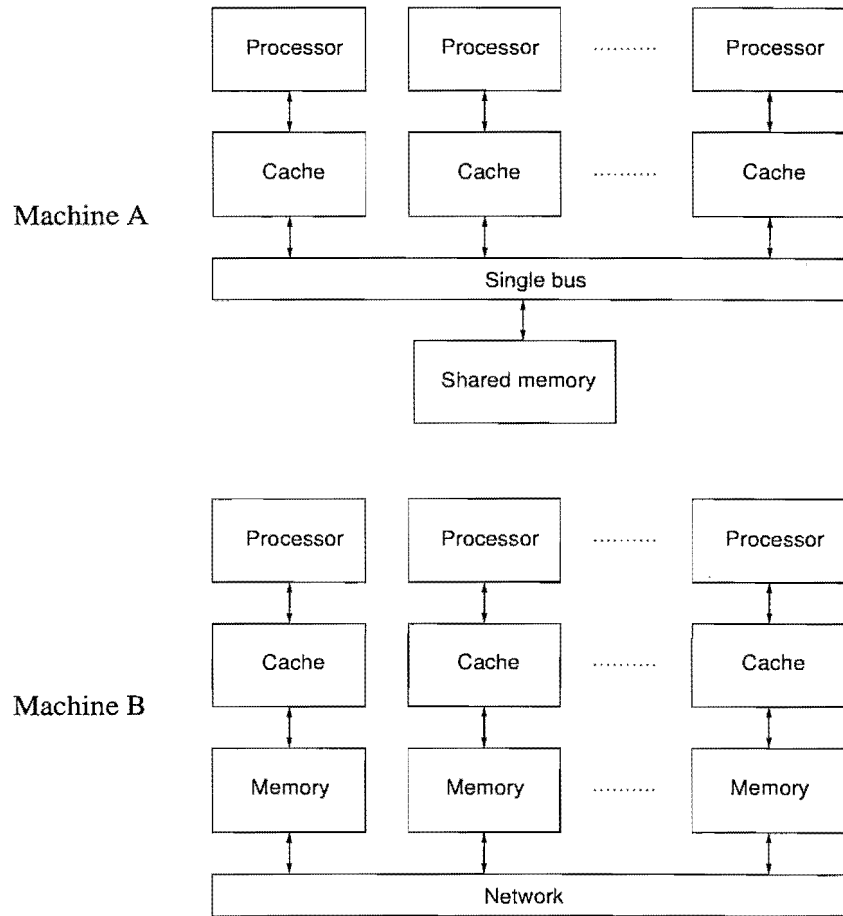
Fig. 1

```
#define NUM_THREADS 4
#define SIZE_VECTOR 10000000

// partial_sum, x and y are global variables stored in shared memory
partial_sum[me] = 0.0;
for (int i = me*SIZE_VECTOR/NUM_THREADS; i < (me+1)*SIZE_VECTOR/NUM_THREADS; i++)
    partial_sum[me] = partial_sum[me] + x[i]*y[i];
```

Fig. 2

2   (a)   What requirements of a modern computer system motivate the adoption of a virtual memory system? [15%]

(b)   In a virtual memory system, what is the purpose of the translation lookaside buffer (TLB)? Give typical values for a TLB's total size, block size, hit time and miss rate. [15%]

(c)   A computer system has a 64-bit virtual address, 4 KB pages and 512 MB physical memory. The page tables are indexed by the virtual page number, with one entry per virtual page. If the system is running 100 processes, what is the total amount of memory tied up by page tables? Assume that the page table entries are a whole number of bytes long, and contain valid and dirty bits but not disk addresses. [20%]

(d)   An *inverted page table* is indexed instead by the physical page number. Each inverted page table entry contains the virtual page number, valid and dirty bits, and also a 16-bit process identifier.

    (i)   How many inverted page tables are required: one per process, or just one? Justify your answer. [15%]

    (ii)   For the computer system in (c) running 100 processes, and again assuming that each entry is a whole number of bytes long, calculate the total amount of memory required by the inverted page table(s). [15%]

    (iii)   Discuss how an inverted page table might be searched for the required translation. Suggest both naive (slow) and sophisticated (fast) search strategies. How does the search time compare with that of the traditional page table arrangement in (c)? [20%]

3   (a)   Explain, with the aid of a simple diagram, how the structure of a coaxial cable makes it a suitable candidate for a physical layer technology in a local area network. Describe what is meant by the term *base* in the context of coaxial cable, and explain why coaxial cables have been mostly replaced by twisted pair cabling.   [30%]

(b)   Coaxial cable was used successfully for many years in both wide and local area networks. Give two examples of how the choice of coaxial cable has had a major impact on protocols operating above the physical layer. Why is this not ideal when developing protocols under the open systems interconnect (OSI) model?   [50%]

(c)   Coaxial cable has recently been revived as a possible physical layer technology as part of the 10G and 100G ethernet standards. Discuss the physical layer options available at these data rates and explain how some of the limitations of coaxial cable have been overcome.   [20%]

4   (a)   Describe the main function of a layer-three router and the main differences between static and adaptive routing methodologies when switching packets between nodes in a packet switched network. Explain how the routing methodology might dictate the quality of the service offered by the network.   [30%]

(b)   One of the most important concepts when considering routing techniques within a packet switched network is the address structure. Explain, with the aid of an example protocol, the key features of both layer-two and layer-three address structures. How does the choice of address structure affect the scalability of a packet switched network such as the internet?   [40%]

(c)   Since the late 1960s, the internet has evolved into the global network that we have today. What are the two key features of the internet protocol which have allowed this evolution to occur? Explain why these same features are also two of the main reasons why high quality services, such as video, remain a challenge over the internet.   [30%]

END OF PAPER