THE UNIVERSITY OF WARWICK
MSc Examinations: Summer 2016
High Performance Computing

Time allowed: 3 hours
Answer FOUR questions.
Calculators may be used.
Read carefully the instructions on the answer book and make sure that the particulars required are entered on each answer book.

1. (a) In a typical shared memory architecture, multiple CPU cores share the common memory, while a distributed memory architecture typically refers to the architecture where multiple computers are connected via network cables and each computer has its own main memory. Explain the differences in communication between shared memory architecture and distributed memory architecture. [3]

(b) Explain what the pipeline mechanism is. Discuss why it is difficult to keep the pipeline full. List two techniques that can be used to improve the performance of the pipeline mechanism. [8]

(c) Discuss the differences between scientific applications such as matrix multiplication and graph-based applications such as online-shopping recommendation. Focus your discussions on data structure, performance metric and key factors that affect the performance. [10]

(d) Analyse the two "for" loops in Listing 1. Describe whether the iterations of these two loops can be parallelised automatically by compilers and explain how you reached your conclusions. [4]

Loop 1:
```
for(i=1; i<=n; i++)
   a[i]= b[i] + c[i];
```

Loop 2:
```
for(i=2; i<=n; i++)
   a[i]= 2×a[i-1];
```

Listing 1: Two loops for Question 1(d)

Continued

2 (a) The Buffer mode is a communication mode in MPI. Explain what the Buffer mode is. Discuss in what circumstance the buffer mode is desired. [4]

(b) Assume there are two MPI processes, p0 and p1, and p0 and p1 are running in different machines. In p0, MPI_Send is first called to send message A to p1 and then MPI_Recv is called to receive message B from p1. In p1, MPI_Send is first called to send message B to p0 and then MPI_Recv is called to receive message A from p0. What will happen if the sizes of both message A and B exceed the system buffers managed by MPI? Explain why. [7]

(c) A collective communication operation is performed by all relevant processes at the same time with the same set of parameters. However different processes may interpret the parameters differently. Describe, using illustrative examples if necessary, the operations of the following two MPI collective communication calls. Further, discuss how different processes interpret different parameters in these functions.

i) MPI_Bcast(void **buf*, int *count*, MPI_Datatype *type*, int *root*, MPI_Comm *Comm*)

ii) MPI_Gather(void **sendbuf*, int *sendcnt*, MPI_Datatype *sendtype*, void **recvbuf*, int *recvcount*, MPI_Datatype *recvtype*, int *root*, MPI_Comm *comm*) [8]

(d) *MPI_Type_create_indexed_block* can be used to construct the users' own data types. The format of the function is as follows:

MPI_Type_create_indexed_block (int *count*,
int *blocklengths*,
int **array_of_displacements*,
MPI_Datatype *oldtype*,
MPI_Datatype **newtype*)

Let *oldtype* ={(MPI_INT, 0)} with the extent of 16 bits. Let D=(2, 10).

Give the memory layout of *newtype* after calling
MPI_Type_create_*indexed_block* (2, 3, D, *oldtype*, *newtype*) [6]

3. (a) Discuss the differences between multicore CPU and GPU in terms of architecture design and performance objective. [7]

   (b) Assume that a GPU device allows up to 8 concurrent thread blocks and 1024 threads per SM (Streaming Multiprocessor), whichever becomes a limitation first, and that it allows up to 512 threads in each block in the device. Given three candidate sizes of thread block, 8 x 8, 16 x 16 and 32 x 32, which block size should be used? Show your reasoning. [10]

   (c) Explain the workings of static scheduling and dynamic scheduling in the *schedule* clause of OpenMP. [4]

   (d) Discuss why in Java synchronized statements can improve concurrency over synchronized methods. [4]

---

4. (a) Describe the general steps of an iterative numerical method to solve a system of linear equations. [7]

   (b) Assume that when a program performs computations at each point in a 3-dimensional grid, it needs to get the data from four neighboring points. The communication pattern among the grid points in the horizontal dimensions is shown in Figure 1, while there is no communication among the grid points in the vertical dimension. The size of each dimension of the grid is N.

      i) Construct performance models for the program's execution time when the grid is partitioned using 1-D decomposition. [10]

      ii) Based on the constructed performance model, derive parallel efficiency and iso-efficiency function. [8]

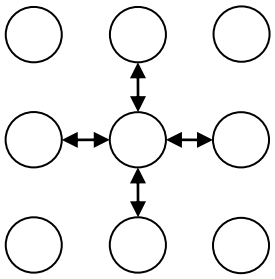

Figure 1. The Communication Pattern in Question 4(b)

Continued

5. (a) The topology of node interconnection plays an important role in the performance of a Cluster system. Draw the topology of a 3-D hypercube. What are the values of node degree and bisection width of the topology? [6]

   (b) Compared with the traditional security techniques for Client/Server systems, discuss why Grid systems need more complex security mechanisms. [6]

   (c) SEDF (Simple Earliest Deadline First) and Credit are two schedulers deployed in the Xen hypervisor. Discuss why the Credit scheduler provides better load balance than SEDF among multiple CPU cores in a multicore machine. [3]

   (d) Condor and PBS are two types of resource management system. Discuss the differences between them. [10]

---

6. (a) Discuss why collective I/O can potentially improve I/O performance. [6]

   (b) Explain why non-blocking I/O can help improve application performance. [3]

   (c) Discuss why there may be I/O inconsistency between MPI processes. Describe two methods that can be used to ensure I/O consistency in MPI. [6]

   (d) There are three potential methods to implement parallel I/O: 1) One process performs I/O operations for all other processes; 2) Each process reads or writes the data from or to a separate file; 3) Different processes access different parts of a common file. Discuss the advantages and disadvantages of each method. Which method of parallel I/O is most widely used nowadays? [10]

End