COMPUTER SCIENCE TRIPOS Part IB

Wednesday 3 June 2009 1.30 to 4.30

COMPUTER SCIENCE Paper 5

*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
Script paper
Blue cover sheets
Tags

SPECIAL REQUIREMENTS
Approved calculator permitted

1 Compiler Construction

(*a*) Give *two* reasons why a compiler is typically broken down into *passes* (or *phases*) and indicate a typical set of passes and the data structures passed between them. [4 marks]

(*b*) Explain one way in which the process of compiling a Java applet, to be stored on a web page and run within many browsers, might differ from that of compiling an application and running it multiple times on a single machine. [2 marks]

(*c*) Explain the difference between a file containing textual assembly code and a file containing an object module. Indicate the essential structure of an object file, for example ELF, highlighting the purpose of each section. Give assembly-level or C-level code that produces entries in each of these sections. [7 marks]

(*d*) Explain the actions of a *linker*, the role of libraries and what makes certain object files directly executable. [4 marks]

(*e*) Explain the difference between *static* and *dynamic linking* giving one advantage of each. Which is most appropriate for an applet stored on a web site to be loaded into a browser? [3 marks]

2 Compiler Construction

Consider an ML-like language in which the set of values includes functions and these functions may have nested definitions.

(*a*) Explain how a function value may be represented at run-time, both in a syntax-tree interpreter and in compiled code. What is the importance of the word "nested" above? [3 marks]

(*b*) Give an example program that gives different results in static and dynamic scoping. In each case, explain how an interpreter or compiled code might perform function application. [3 marks]

(*c*) Considering only static scoping from now on, explain what restrictions are necessary to implement function values without using an auxiliary heap. Explain the notions of static and dynamic chains, giving an example of a situation in which they differ (i.e. reference to the wrong one would access the wrong variable). Illustrate how updating (by assignment to) a free variable can be implemented. [5 marks]

(*d*) Give an alternative heap-based implementation for function values when the restrictions in part (*c*) do not hold. Explain how free variables (and particularly their update) can now be implemented. [4 marks]

(*e*) Java normally holds local variables in stack frames and instance variables on the heap. Consider a Java-like language with nested classes, and a possible program of the form:

```
class C {
  int f(int x)
   { class D { int addx(int y) { return x+y; }
               void updtx(int y) { x = y; }
             }
     ...
   }
};
```

Explain a possible implementation of such nested classes based on your answers above; also give and justify a restriction on methods like `updtx` that eases the cost of implementation. [5 marks]

3 Computer Design

(*a*) Why are control-flow machines sensitive to memory access latency? [4 marks]

(*b*) What statistical properties of data access patterns do caches exploit to reduce memory access latency? [4 marks]

(*c*) What cache line replacement policies might be used for set-associative and direct-mapped caches? [4 marks]

(*d*) What are two write-back policies for a cache? [4 marks]

(*e*) What does a snoopy cache do in a dual-core machine? [4 marks]

4 Computer Design

(*a*) For a MIPS-32 processor executing a jump to subroutine, how are state and control passed between the function and the caller? Indicate what state is passed, but details of particular register numbers are not required. [5 marks]

(*b*) On the MIPS-32 processor the flow of control can be changed using branch or jump instructions, or by three other mechanisms. What are the *three* other mechanisms for changing the flow of control and what are they used for? [6 marks]

(*c*) What is a *control hazard* and what hardware and software techniques can be used to resolve control hazards? [5 marks]

(*d*) Some instruction sets make all instructions conditional (e.g. the ARM) or have conditional move instructions (e.g. IA32). How can these conditional instructions be used to avoid control hazards? [4 marks]

5 Concurrent Systems and Applications

(*a*) Reflection.

(*i*) Give Java code fragments demonstrating *two* different ways of obtaining a `Class` object that describes an array of `java.lang.String`s. [2 marks]

(*ii*) Given an object `x`, write a Java expression that uses reflection to create a new object of the same datatype as `x`. [2 marks]

(*iii*) The `clone()` method creates an exact copy of an object, including all of its fields. Briefly describe how you might implement this functionality using reflection, ignoring inherited fields. Assume that the object is not an array, has a zero-argument constructor, and contains only primitive fields. (You need not give code for an actual complete implementation.) [4 marks]

(*b*) Generics.

(*i*) Suppose a class `B` is a subclass of `A`. Is the class `Set<B>` a subclass of `Set<A>`? Explain why or why not, with regard to type safety. [2 marks]

(*ii*) The default clone method returns an `Object` that must be cast to the correct type. Using generics, give a declaration of a static `myclone` method that takes a single argument of any type and returns an object of the same type. [2 marks]

(*iii*) The `Contraster` interface is used to compare two objects. Its declaration is:

```
interface Contraster<T> {
  boolean greaterThan(T obj1, T obj2);
}
```

Suppose we want to declare a class `SortedList<E>` whose constructor takes a single `Contraster` argument that will be used to compare its elements. Give a declaration for the constructor that permits the choice of contraster implementation to be as general as possible, and explain your reasoning. [4 marks]

(*c*) Reference objects.

The `get()` method of the `PhantomReference` class always returns `null`. Why is this so, and why must a `PhantomReference` always be used together with a `ReferenceQueue`? [4 marks]

6 Concurrent Systems and Applications

(*a*) The following method is intended to return unique integer values to callers:

```
volatile int x = 0;
int getNext() {
  x = x + 1;
  return x;
}
```

(*i*) Two threads call `getNext` concurrently on the same object. Explain how both threads can receive the result `1`. [1 mark]

(*ii*) Explain the semantics of the `synchronized` keyword in Java, and illustrate this by correcting `getNext` (you may ignore the possibility of integer overflow). [6 marks]

(*iii*) Explain the meaning of the `volatile` modifier. Explain whether or not you need to use it with your new implementation of `getNext`. [2 marks]

(*b*) The following method is intended to implement a *barrier* for synchronization between four threads. The first three threads to call the `barrier` method are meant to block. These threads are all unblocked when the fourth call is made.

```
int barrierCount = 0;
void synchronized barrier() throws InterruptedException {
  barrierCount ++;
  if (barrierCount < 4) {
    wait();
  } else {
    notifyAll();
} }
```

(*i*) A programmer finds that some threads return early, although there have been fewer than four calls to `barrier`. How can this happen? [2 marks]

(*ii*) Rewrite `barrier` so that threads wait correctly. [2 marks]

(*iii*) Explain whether or not it would be correct to use `notify` in place of `notifyAll` in your solution. [2 marks]

(*iv*) If a thread is *interrupted* while waiting within `barrier` then the call to `wait` will fail with `InterruptedException`. Rewrite `barrier` so that, if one thread is interrupted when using a barrier, then any future (or concurrent) calls to that barrier will also fail with `InterruptedException`. [5 marks]

7 Digital Communication I

(*a*) Compare *flow control* with *congestion control*. [2 marks]

(*b*) Describe what is meant by *sliding-window protocol*. [2 marks]

(*c*) Describe how a sliding-window protocol may be used to implement flow control. [2 marks]

(*d*) Explain why implementing flow control in this manner is not a good idea. [2 marks]

(*e*) Provide an alternative to a sliding-window protocol for the implementation of flow control. [2 marks]

(*f*) Consider a sliding-window protocol for a point-to-point link from the surface of the earth to a geostationary satellite. The link speed is 1Gbps and the one-way latency is 125ms.

(*i*) Assuming each packet of data is fixed to 1KByte in length, what is the minimum number of bits you need for the sequence number? [2 marks]

You have been asked to construct an emulation of the satellite link, replicating the behaviour (delay and speed), allowing others to test their applications without using the satellite system. A simple way to do this is to provide an artificial delay of packets, emulating their flight to and from the satellite.

(*ii*) For a simple packet length of 1KByte, how much memory is required to emulate the satellite link alone? Comment on other sources of memory utilisation. [2 marks]

(*iii*) What capabilities must a standard computer have to emulate a link with 1Gbps capacity? Consider the speed and delays in the CPU, memory, and PCI interconnect. Comment on the suitability of a standard PC platform for such a task. [6 marks]

8 Digital Communication I

(*a*) Define the following terms and illustrate with an example.

(*i*) Baud (sometimes referred to as the Baud rate), comparing it with bit rate. [1 mark]

(*ii*) Manchester line-encoding. [2 marks]

(*iii*) CRC (Cyclic-Redundancy Check) function. [3 marks]

(*iv*) Hamming distance. [3 marks]

(*b*) Digital data may be represented as 1's and 0's. On a communications link it may be difficult to differentiate an idle link (consecutive 0's) from a broken link. A number of schemes are used to indicate that the link is idle but functioning; two examples include *data scramblers* and *block codecs*.

(*i*) Compare and contrast block codecs with scramblers, taking care to describe each fully. [4 marks]

(*ii*) We wish to achieve an encoded data rate of 1Gbps. Compute the required (symbol) line bit rate for a block codec (e.g. the 8b/10b block codec) and a fixed-length scrambler (e.g. as used in the 64b/66b codec) to achieve a 1Gbps data rate. Which method is more efficient? [2 marks]

(*iii*) Give an example where the 8b/10b block codec would be more desirable than the 64b/66b codec. [2 marks]

(*iv*) Scramblers such as that used in the 64b/66b codec are sometimes referred to as self-synchronising. Describe, using an example if required, what is meant by self-synchronising. [3 marks]

9 Introduction to Security

(*a*) Make the following statements correct by changing one word or number. (Negating the sentence is not sufficient.)

(*i*) The Advanced Encryption Standard defines a 16-round Feistel cipher. [1 mark]

(*ii*) Files encrypted with Cipher Block Chaining start with a zero initial vector. [1 mark]

(*iii*) Each user on a Unix system is identified by a unique prime number. [1 mark]

(*iv*) The "read" bits associated with a Unix directory affect whether the files in its subdirectory "`foo`" can be accessed. [1 mark]

(*v*) The "real user ID" associated with a Unix process determines its access rights. [1 mark]

(*b*) Name *five* examples of actions for which a Unix application will need to be invoked with *root* privileges. [5 marks]

(*c*) Explain the attack on Double DES that motivates the use of Triple DES. [6 marks]

(*d*) Under which conditions is the Vignère cipher unconditionally secure? [4 marks]

END OF PAPER