STUDENT NAME: _____
STUDENT NUMBER: _____

FACULTY OF SCIENCE
FINAL EXAMINATION

COMPUTER SCIENCE COMP 302

Programming Languages and Paradigms

Examiner: Prof. Prakash Panangaden
Associate Examiner: Prof. Laurie Hendren

11th December 2006
9am to 12 noon

Instructions:

This exam has 10 questions. Please answer all questions. This is an **open book exam**: You may use any books or notes that you have including dictionaries. You have three hours in all. You may **not** use calculators, computers, cell phones, or electronic aids of any kind. Please answer all questions **on the question paper itself** and return it at the end.

This exam has 11 pages, including the cover page.

Q 1	Q 2	Q 3	Q 4	Q 5	Q 6	Q 7	Q 8	Q 9	Q 10	Total
10	10	10	10	10	10	10	10	10	10	100

Question 1

A square matrix can be represented by a list of lists. For example, the 4×4 matrix

$$\begin{pmatrix} 3 & 1 & 2 & 6 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 3 & 9 \\ 1 & 0 & 0 & 4 \end{pmatrix}$$

would be represented in SML by the list of lists

```
val matrix = [[3,1,2,6],[0,2,1,0],[0,0,3,9],[1,0,0,4]] : int list list
```

Write an SML program to take a list of lists of numbers and a positive integer n as input, and return a list representing the nth column. You may use the built-in function `map`. An empty list is a valid square matrix. You may assume that you are getting a proper list of lists as input; you do not have to test whether you have a square matrix. You do have to test whether your column number is in range; raise an exception if it is out of range. For example, if we give the above matrix and the number 3 as input, we should get back `[2,1,3,0]`.

Question 2

Suppose that I have a real-valued function $F(x, y)$ of two arguments, x can be of any type, but y is real. I can define a function of one argument as follows:

$$f(x) = \int_0^{y_0} F(x, y)dy.$$

Write a *higher-order* Sml function `indefint`, that takes as arguments a 2-argument function, like F above, a real value, y_0, and a real value `delta` to be used in the integration. The result to be returned is a *function* like the f shown above. You may assume that you are given an Sml function to compute definite integrals, given a one-argument function, upper and lower limits and a `delta`. The type for `integral` and the type that I expect for `indefint` are shown below.

```
val integral = fn : (real -> real) * real * real * real -> real
val indefint = fn : ('a * real -> real) * real * real * real -> 'a -> real
```

Question 3

You have seen examples of sorting functions in Sml, for example, insertion sort. Insertion sort was defined on integer lists and used the built-in comparison function $<$. Write a higher-order function `make_sorter` that takes, as argument, a two argument boolean valued function, `comparator` and returns a function that sorts a list using the comparator.

```
val make_sorter = fn : ('a * 'a -> bool) -> 'a list -> 'a list
```

Question 4

Define a one-parameter function `previous`, in Sml, that returns as its value the argument that was passed to it the previous time that it was called. The first time it is called it should return 0.

```
val previous = fn : int -> int ref
val it = () : unit
- previous 4;
val it = ref 0 : int ref
- previous 5;
val it = ref 4 : int ref
- previous 3;
val it = ref 5 : int ref
```

Question 5

What is the result of evaluating the following expression? Explain your answer drawing the relevant environment diagrams. Without the explanation I will give zero, even for a correct answer, which, by the way, is 7.

```
let
    val x = 1
    val y = 2
in
    let
        val f = fn u => let val x = 3 in (u + x) end end
    in
        let
            val y = 4
        in
            f(y)
        end
    end
end
```

Question 6

We use the following datatype for simple arithmetic expressions.

```
datatype exptree =
Plus of exptree * exptree | Times of exptree * exptree | Leaf of int
```

Write an Sml function `eval` that evaluates an expression tree and returns an integer.

Question 7

Do the following two examples of SML code type check? If yes, return the final type, SML would assign and explain in one sentence informally how SML would derive such a type, if no, explain why SML is not able to type-check the given example. If the value restriction is invoked, this is considered to be not typable.

```
let
  val f = (fn y => (fn x => x))
in
  ((f true) 5, (f 5) true)
end;
```

The type is:

```
val f = ref (fn x => x)
```

The type is:

Question 8

The Church numerals can be implemented in Sml as well as Scheme. Here are the expressions for one and two.

```
val one = fn f => (fn x => f x)
val two = fn f => (fn x => f(f x))
```

What are the most general types for these expressions? Please derive the types.

Question 10

Please give a true or false answer to each question. No explanations are needed. All questions refer to Java.

1. If a class A extends a class B then any time an A object is needed, a B object can be supplied.
2. If a class C implements an interface I, then C is a subtype of I.
3. If a class D and a class C both implement an interface I, then one of C or D has to be a subtype of the other.
4. It is not possible for a type to be a subtype of two other types.
5. Subtyping is exactly the same thing as inheritance.