CSCE 330 Fall 2007
FINAL EXAM
Friday 2007-12-17
Closed Book except for handouts on denotational semantics and FP

1 Short Answer Questions–8 points

1. (1 point) In Simplesem, the effect of the instruction `set 10, D[20]` is to copy the value stored at location 10 of data memory into location 20 of data memory. True or false?

2. (2 point) What are the three components of the state in the denotational semantics approach[1]?

3. (2 points) There are three major approaches to describing the semantics of programming languages. List them.

4. (1 point) What does the alphanumeric identifier `‘Z` stand for in ML?

5. (1 point) What does the alphanumeric identifier `‘‘Z` stand for in ML?

6. (1 point) What is the domain of = `‘‘Z * ‘‘Z`?

[1] Every question on denotational semantics refers to the simple language described in class.

2 Syntax and Semantics—20 points

1. (Robert Sebesta–5 points) Describe, using a single English sentence, the language defined by the following grammar:

```
<S> ::= <A><B><C>
<A> ::= a<A> | a
<B> ::= b<B> | b
<C> ::= c<C> | c
```

2. (15 points in total) Give a loop invariant for this program fragment :

```
x := 2;
i := 1;
(*What is the precondition here?*)
while (i <= n) do
  begin
    x := x*x;
    i := i+1
  end
```

with precondition $n \geq 1$ and postcondition $x = 2^{2^n}$ (5 points)

Also answer the following questions.

(a) (2 points) What is the precondition before the loop?

(b) (2 points) Your invariant should consist of the conjuction of two formulas. One of them should be very similar to the condition of the while loop. Explain why that formula is needed.

(c) (2 points) Show that the precondition at the line with asterisks implies the invariant.

(d) Show that the invariant together with the negation of the loop implies the postcondition.

(e) (2 points) Let x be the value of the variable `x` before executing the body of the loop and x' be the value of the variable `x` after executing the body of the loop. Write an equation that relates x and x'.

(f) (2 points) Let i be the value of the variable `i` before executing the body of the loop and i' be the value of the variable `i` after executing the body of the loop. Write an equation that relates i and i'.

3 FP–22 points

1. (4 points) Write a function that multiplies its argument by seven. Call it `timesseven`. So, for example, `timesseven:5` is `35.0`. (The "`.0`" appears if you use Carter Bays's FP interpreter.)

2. (4 points) Write a function that applies `timesseven` to all elements of a sequence and give an example of its application to a sequence of three numbers. Do not give a name to the function.

3. (2 points) What is `!+: <1 2 3>`? What do you call ! in FP?

4. (7 points) Write a function that computes the length of a sequence. Do not use recursion. Do not use `while`. Use composition. (Hint: What is `& %1 : <1 2 3>`?)

5. (5 points) Call the function you wrote in the previous exercise `length`. (So, for example, `length: <2 3 4>` is 3.) Write a function that computes the average of a sequence of numbers. Call the function `avg`. For example, `avg: <1 4 4>` is 3.0. (The "`.0`" appears if you use Carter Bays's FP interpreter.)

4 ML—48 points

1. (10 points) Consider the ML session printed below.

```
- fun cond(test, then_part, else_part) =
  if test then then_part else else_part;
= val cond = fn : bool * 'a * 'a -> 'a
- val x = 0;
val x = 0 : int
- if x = 0 then [0] else tl[];
val it = [0] : int list
- cond(x=0, [0], tl[]);
uncaught exception Empty
  raised at: boot/list.sml:37.38-37.43
 -
```

   Explain briefly why `if` and `cond` behave differently.

2. (10 points) Write a function `fact` of one argument that computes the factorial of a non-negative integer. Use of patterns is recommended, but not required. Do *not* use the code in the next two question.

3. (2 point) Consider the following function `bfact`, which uses `cond`.

```
fun bfact n: int = cond(n=0, 1, n*bfact(n-1));
```

   What is bfact(3)?

4. (6 points) The following function `facti` of two arguments computes the factorial of a non-negative integer. The second argument is the accumulator. The initial call to compute the factorial of `n` is `facti(n, 1)`.

```
- fun facti( n, p) = if n = 0 then p else facti( n-1, p*n);
val facti = fn : int * int -> int
- facti( 3,1);
val it = 6 : int
```

   Which one of $fact$ and $facti$ is tail recursive? Which one of $fact$ and $facti$ is more efficient? Why?

5. (10 points) Write a function `reverse` of one argument that reverses a list. Your function should not be tail recursive.

6. (10 points) Here is an example of definition and use of a higher-order function in ML.

```
- fun simpleMap(F, nil) = nil
|   simpleMap(F, x::xs) = F(x)::simpleMap(F,xs);
= val simpleMap = fn : ('a -> 'b) * 'a list -> 'b list
- fun plus2 x = x + 2;
val plus2 = fn : int -> int
- plus2 3;
val it = 5 : int
- simpleMap (plus2, [1,2,3,4,7]);
```

What is `it` now? What does `simpleMap` do? What is the name of `simpleMap` in the FP language?

5 Prolog—10 points

(10 points in total) Your textbook [Ghezzi and Jazayeri, p.394] gives, as an example of "Prolog database," a relation "composer" with these facts:

```
composer(monteverdi, 1567, 1643).
composer(vivaldi, 1678, 1750).
composer(bach, 1685, 1750).
composer(mozart, 1756, 1791).
composer(haydn, 1732, 1791).
composer(beethoven, 1770, 1827).
composer(schubert, 1797, 1828).
composer(schumann, 1810, 1856).
composer(brahms, 1833, 1897).
composer(verdi, 1813, 1901).
composer(debussy, 1862, 1918).
```

1. (3 points) What does Prolog return as the answer to the query `?-composer(X, Y, 1827)`?
2. (3 points) What does Prolog return as the answer to the query `?-composer(X, Y, Z)`?
3. (4 points) Write a query to find out whether there are two composers who were born in the same year.