Computer Science 15–212: Midterm Examination

February 23, 1995

Name

Name: ______________________________________

Andrew ID: ______________________________________

Section: ______________________________________

Instructions

- This is a closed-book, closed-notes examination.
- There are 10 pages in this examination, including two worksheets.
- This examination consists of 4 questions worth a total of 100 points. The point value of each question is given with the question.
- Read each question completely before attempting to solve any part.
- Write your answers legibly *in the space provided* on the examination sheet. If you use the back of a sheet, indicate clearly that you have done so on the front.
- Write your name, Andrew id, and section letter in the space provided at the top of this page, and write your name at the top of each page in the space provided.
- Two worksheets are attached to the end of this examination for your personal use. *Under no circumstances will anything written on the worksheets count for or against your grade on this exam.*

Grading

1	2	3	4	*Total*

Binding and Scope [20 points]

Consider the following compound declaration:

```
val x : int = 2
and y : int = 4
and z : int = 6
val a : int = x + x
and x : int = y + 3
and b : int = x * x
```

Given these declarations, what are the values of the following expressions?

1. [2pts] `a*x + b*y` $\to^*$ ____________________
2. [2pts] `let val x:int=1 in a+a end` $\to^*$ ____________________
3. [2pts] `let val y:int=2 in b * y end` $\to^*$ ____________________

What are the values of the following expressions?

1. [4 pts] The value of

   ```
   let
     val f:int->int = fn x:int => 3*x
     val f:int->int = fn 0 => 1 | 1 => 1 | n => f (n-1) + f (n-2)
   in
     f 5
   end
   ```

   is ____________________.

2. [4pts] The value of

   ```
   let
     val f:int->int = fn x:int => 3*x
     val rec f:int->int = fn 0 => 1 | 1 => 1 | n => f (n-1) + f (n-2)
   in
     f 5
   end
   ```

   is ____________________.

3. [6pts] The value of

   ```
   let
     val y:int = 7
     val f:int->int = fn x:int => x*x + y*y
     val y:int = 9
   in
     f 5
   end
   ```

   is ____________________.

Induction and Recursion [30 points]

Consider the elements of a list of length n to be numbered $0, \ldots, n-1$. The problem is to extract the sublist $l[i:j]$ of l consisting of the elements numbered i through j, inclusive, where $0 \leq i \leq j < n$. (Note that according to these assumptions $l[i:j]$ is defined only for non-empty lists l!)

[15 points] Define a function `segment` of type `'a list * int * int -> 'a list` such that `segment (l, i, j)` is the segment of the list l determined by i and j as described above. Give your definition in clausal form using a `fun` declaration. *It is not required that your solution be in tail recursive (iterative) form.*

[15 points] Prove that if `0`$\leq$`i`$\leq$`j`$<n$, where n is the length of l, then `segment(l,i,j)`$\rightarrow^*$ `l[i:j]`. Here is a checklist of things to consider:

1. The proof is by induction on ______________ (specify the variable).
2. State and prove the basis of the induction:

3. State the inductive hypothesis:

4. State and prove the inductive step:

Abstract Data Types: Polynomials [20 points]

A degree n polynomial $a_0 + a_1\,x + \cdots + a_{n-1}\;x_{n-1}$ may be represented by the list of its coefficients `[`a_0`,...,`a_{n-1}`]` written in ascending degree order. We regard the empty list as standing for 0.0. Note that if as `: real list` is a non-empty list of real numbers representing the polynomial p, and a is a real number, then a`::`as represents the polynomial $a + p\,x$.

Let us make the following type definition:

```
type poly = real list
```

1. [4pts] Write down the equations for addition and multiplication of polynomials in recursive form by completing the following equations:

   (a) $(a + p\,x) + (b + q\,x) =$ ______________________________.

   (b) $(a + p\,x) \cdot q =$ ______________________________.

2. [5pts] Define the function `add_poly : poly * poly -> poly` to add two polynomials.

3. [5pts] Define the function `mul_poly : poly * poly -> poly` to multiply two polynomials. *Hint:* make use of an auxiliary function suggested by your recursion equation for multiplication given above.

4. [6pts] Given a list `[`a_0`,...,`a_{n-1}`]` of $n > 0$ real numbers, there is a unique polynomial p with these n numbers as roots. Define a function `poly_with_roots : real list -> poly` that creates a polynomial from the given set of roots. *Hint:* Write down in ordinary mathematical notation a formula for the required polynomial. Then use the functions `add_poly` and `mul_poly` to define `poly_with_roots`.

Recursion [30 points]

Ms. M.L. has a summer house which she wants to rent out for period of n days. She has solicited and received m bids. Each bidder has specified a start date and an end date (integers from 1 to n) and the rent (in number of dollars) he or she is willing to pay for that period. Ms. M.L. must select a subset of the bids so as to maximize her total rental income, subject to the restriction that only one bidder can stay in the house at a time.

Let $S(i)$ be the starting date of bid number i, $E(i)$ be the ending date of bid number i, and $B(i)$ be the proposed rent for bid number i. Let $R(d)$ be the maximum rent for days 1 through d, where $d \geq 0$. Then

$$\begin{aligned} R(0) &= 0 \\ R(d) &= \max_{i=1}^{m}\{\, R(S(i)-1) + B(i) \mid E(i) \leq d\,\} \end{aligned}$$

That is, she earns no income for not renting for zero days, and maximizes her income for day $d > 0$ by calculating for each bid whose end date is no later than d the maximum rent she can earn up to, but not including, the start date of that bid, plus the rent accrued from that particular bid, and choosing the largest such value.

Let us represent a bid by the following type:

```
type bid = { start_day : int, end_day: int, rent: int }
```

You are to define a function `max_rent : int * bid list -> int` such that `max_rent (n,l)` yields the maximum rent that Ms. M.L. may earn for days 0 through n from the list of bids l. You may assume that for each bid b in l, `0` $<$ `#start_day` $b \leq$ `#end_day` $b \leq n$.

You may assume given a function `max_list` of type `int list -> int` which yields the largest element of a given list of integers.

Extra Credit: Modify your function to not only return the maximum rent but also a list of bids which yields the maximum rent.

Worksheet

Worksheet