Matthew Silva
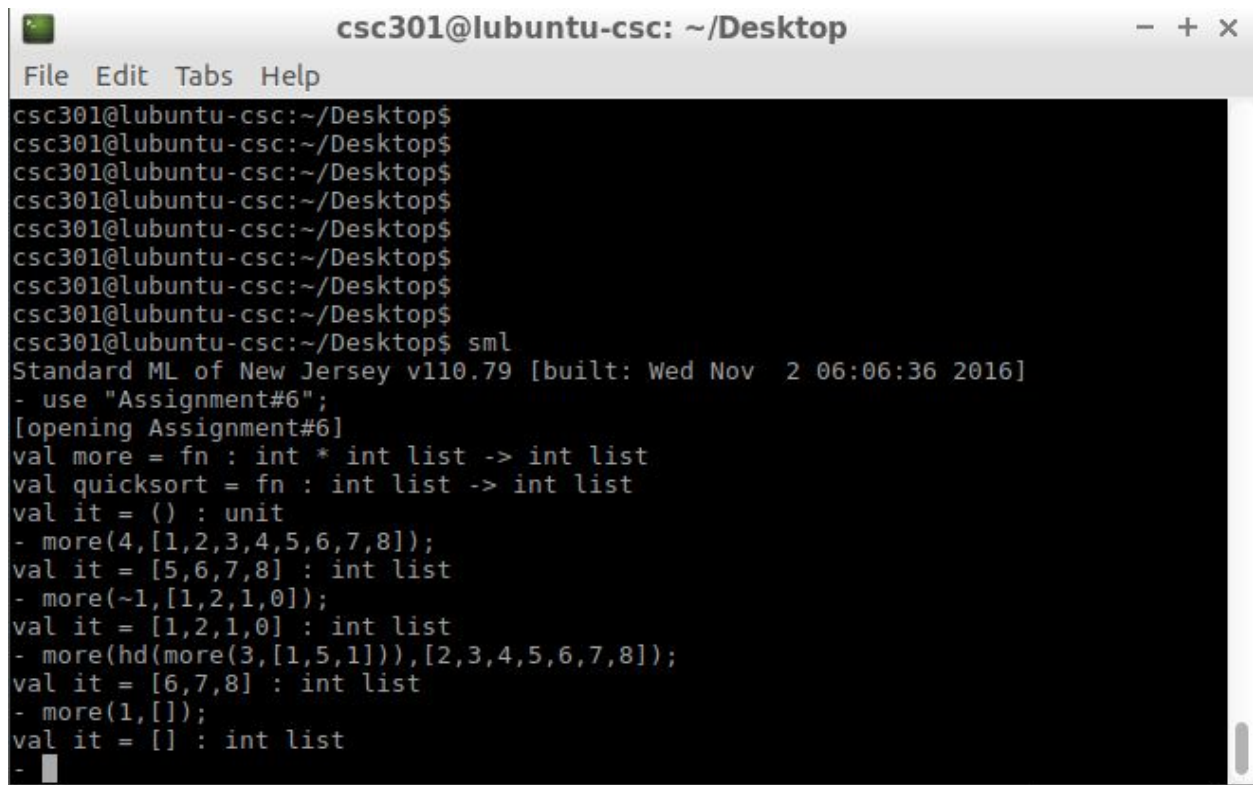
Professor Hamel

CSC 301

21 October 2017

<div align="center">Assignment #6</div>

```
fun more(e, []) = []
|       more(e, first::rest) =  if first > e then first::more(e, rest)
                               else more(e, rest)
```
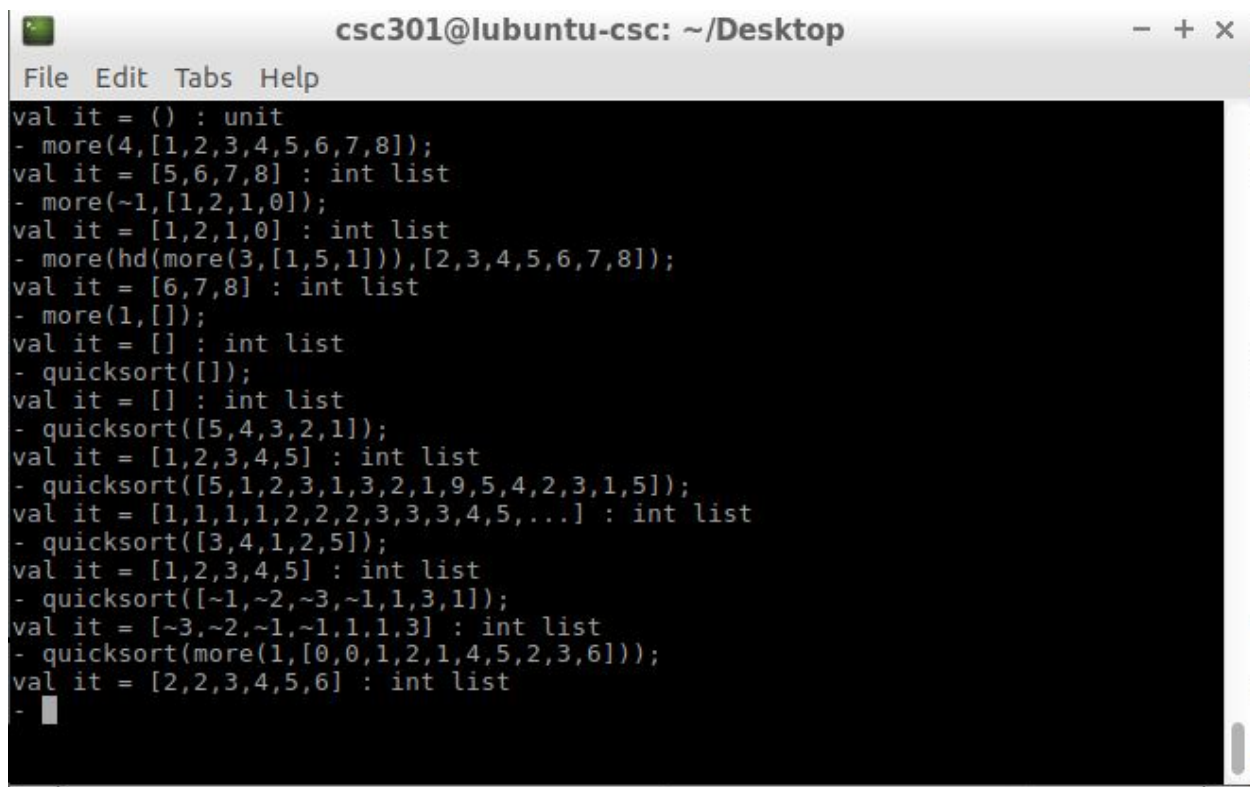
```
fun quicksort [] = []
| quicksort [a] = [a]
| quicksort (pivot::rest) =
    let
        fun compareToPivot(pivot, []) =([], [])
        |   compareToPivot(pivot, next::rest) =
                    let
                            val (x,y) = compareToPivot(pivot, rest)
                    in
                            if next < pivot then (next::x, y) else (x, next::y)
                    end;
        val (smaller, larger) = compareToPivot(pivot, rest)
    in
        (quicksort(smaller))@[pivot]@(quicksort(larger))
    end;
```



```
val it = () : unit
- more(4,[1,2,3,4,5,6,7,8]);
val it = [5,6,7,8] : int list
- more(~1,[1,2,1,0]);
val it = [1,2,1,0] : int list
- more(hd(more(3,[1,5,1])),[2,3,4,5,6,7,8]);
val it = [6,7,8] : int list
- more(1,[]);
val it = [] : int list
- quicksort([]);
val it = [] : int list
- quicksort([5,4,3,2,1]);
val it = [1,2,3,4,5] : int list
- quicksort([5,1,2,3,1,3,2,1,9,5,4,2,3,1,5]);
val it = [1,1,1,1,2,2,2,3,3,3,4,5,...] : int list
- quicksort([3,4,1,2,5]);
val it = [1,2,3,4,5] : int list
- quicksort([~1,~2,~3,~1,1,3,1]);
val it = [~3,~2,~1,~1,1,1,3] : int list
- quicksort(more(1,[0,0,1,2,1,4,5,2,3,6]));
val it = [2,2,3,4,5,6] : int list
-
```