CMSC 330 OCaml– solutions Fall 2012

1. a. `(string * int) list`

   b. `int -> int list`

   c. `'a -> 'a -> 'a list`

   d. `fun x -> x 2` or `let f x = x 2`

2. a. Any of the following errors would suffice:
   3 has type int but is used with type int list
   [3] has type int list but is used with type int list list
   can only add int list [1;2] to int list list
   trying to add int list [1;2] to int list [3]

   b. `2`

   c. `[3; 4; 5]`

3. Different answers are possible; here are some:

   a. `(2, [2])`

   b. `let f a b = match a with (h::t) -> h + b`

4. a. Different answers are possible; here are some:

```
let rec makeList x = match x with
    [] -> []
  | (h::t) -> [h]::(makeList t);;

let makeList x = map (fun y -> [y]) x;;

let makeList x = map (fun y -> y::[]) x;;

let makeList x = List.rev (fold (fun a y -> [y]::a) [] x);;
```

   b.

```
let over20 x = List.rev (fold
                          (fun a y -> if (y > 20) then y::a else a)
                          []
                          x);;
```

   c.

```
let growNum x =
   let rec g n = if n = 0
                 then []
                 else n::(g (n-1))
   in (List.rev (g x));;
```

   d.

```
let countNum n lst = fold
                       (fun a y -> if (y = n) then a + 1 else a)
                       0
                       lst;;
```