

CS 2041: Practice Exam 2

Fall 2018

University of Minnesota

Exam period: 20 minutes

Points available: 40

Problem 1 (10 pts): Write a function called `strlen_between` which accepts a min and max length and a list of strings. A list of strings with length in the given range is returned. Min/max lengths are inclusive. For full credit, make use of a higher-order function in your definition. Demo uses are given below.

Write your code for `strlen_between` here.

```

1 # #use "strlen_between.ml";;
2 val strlen_between :
3 int -> int -> string list -> string list = <fun>
4 # let lst = ["aaa";"bbbb"; "cccc";"dddd"];;
5 val lst : string list =
6 ["aaa"; "bbbb"; "cccc"; "dddd"]
7 # strlen_between 2 4 lst;;
8 - : string list = ["aaa"; "cccc"]
9 # strlen_between 4 7 lst;;
10 - : string list = ["bbbb"; "cccc"; "dddd"]
11 # strlen_between 9 12 lst;;
12 - : string list = []

```

Problem 2 (10 pts): Write a function called `largest_even` which accepts a list of integers. If no even numbers are in the list, return `None`. Otherwise, return `Some` of the largest integer in the list. For full credit, use pattern matching and a higher-order function in your solution. Demo uses are given below.

Write your code for `largest_even` here.

```

1 # #use "largest_even.ml";;
2 val largest_even :
3 int list -> int option = <fun>
4 # largest_even [];;
5 - : int option = None
6 # largest_even [1;3;5];;
7 - : int option = None
8 # largest_even [4];;
9 - : int option = Some 4
10 # largest_even [1;3;2;5];;
11 - : int option = Some 2
12 # largest_even [1;4;3;2;5;8;7;6];;
13 - : int option = Some 8

```

Background: Shae Lowcopy decided to extend the `multimanager` application to allow the current list to be copied using the `copyto` command. She modifies `multimanager.ml` to include the following additional command sequence.

```

1 let execute_command tokens =
2   let cmd = tokens.(0) in      (* 0th element is command *)
3   match cmd with
4   ...
5   | "copyto" ->
6     let new_name = tokens.(1) in
7     let added = Doccol.add global new_name global.curdoc in
8     if added then
9       printf "Copied list to '%s'\n" new_name
10    else
11      printf "ERROR: list '%s' already exists, cannot create copy\n" new_name
12    ...

```

Unfortunately when she begins testing she sees the below undesirable behavior: the copied list seems to affect the original list when they should be independent.

Problem 3 (10 pts): Describe in some detail why Shae's implementation of `copyto` does not work as expected and why the two lists seem to be linked somehow.

```

1 > multimanager
2 (default.txt)> add Korra
3 (default.txt)> add Mako
4 (default.txt)> add Bolin
5 (default.txt)> show
6 --BEG LIST--
7 Bolin
8 Korra
9 Mako
10 --END LIST--
11
12 (default.txt)> copyto others.txt
13 Copied list to 'others.txt'
14 (default.txt)> lists
15 2 docs
16 - others.txt
17 - default.txt
18
19 (default.txt)> edit others.txt
20
21 (others.txt)> show
22 --BEG LIST--
23 Bolin
24 Korra
25 Mako
26 --END LIST--
27
28 (others.txt)> remove Korra
29 (others.txt)> add Asami
30 (others.txt)> showall
31 --List others.txt--
32 Asami
33 Bolin
34 Mako
35
36 --List default.txt--
37 Asami
38 Bolin
39 Mako

```

Problem 4 (10 pts): Describe how to fix the problem so that a proper list copy is made for `copyto`. Provide at least some code to give a concrete idea of how your idea would work.