CSCI 421 Project 3: Map and fold

Objectives

- Implement a function that mimics built-in `map` function in SML.
- Apply `foldr`, `foldl`, and `map` to solve some problem with a single line of code.

Problem description

Implement the following five functions:

1. Define a function named `mymap1` with the same type and behavior as built-in `map` function without actually using `map`. This should be one line of code and should use foldr or foldl. (Refer to problem Exercise 24 on page 147.)
2. Define a function named `mymap2` with the same type and behavior as `map`, but unlike before you may not use foldr nor foldl. You still cannot use `map` itself either. (Refer to problem Exercise 26 on page 147.)
3. Write a function named `ordlist` of type `char list -> int list` that take a list of characters and returns the list of integer codes of those characters. For example, if you evaluate `ordlist [#"A", #"b", #"C"]` you should get `[65, 98, 67]`. (Refer to Exercise 2 on page 144.)
4. Write a function named `mylength` of `type 'a list -> int` that returns the length of a list. You cannot use the built-in `length` function. (Refer to Exercise 11 on page 145.)
5. Write a function named `max` of type `int list -> int` that returns the largest element of a list of integers. Your function must use either `foldr` or `foldl` and need not behave well if the list is empty. (Yes, this is the same function from Project 1. Here you must use a `fold` variant rather than recursion.)

Write these functions in a file called ProjectThree.sml and turn it in on Blackboard.

Sample run

```
$ sml
Standard ML of New Jersey (64-bit) v110.99 [built: Thu Dec 24 11:47:23 2020]
- use "ProjectThree.sml";
[opening ProjectThree.sml]
val mymap = fn : ('a -> 'b) -> 'a list -> 'b list
val mymap2 = fn : ('a -> 'b) -> 'a list -> 'b list
val mylength = fn : 'a list -> int
val ordlist = fn : char list -> int list
val max = fn : int list -> int
val it = () : unit
- ordlist [#"A", #"B", #"C"];
val it = [65,66,67] : int list
- mylength [];
val it = 0 : int
- mylength [9, 1, 4, 2, 3, 8, 7];
val it = 7 : int
- max [8, 2, 5, 9, 4, 7, 1, 3];
val it = 9 : int
- max [~1,5,0];
val it = 5 : int
```

Grade breakdown

Criteria	Weight
Code is cleanly formatted & appropriately documented	20%
Correctness of `mymap1` (including not using forbidden functions)	20%
Correctness of `mymap2` (including not using forbidden functions)	20%
Correctness of `ordlist`	10%

Criteria	Weight
Correctness of `mylength` (including not using forbidden functions)	10%
Correctness of `max` (including using either `foldl` or `foldr`	20%

If your code doesn't compile without modifications, 20% will be deducted from your score.