CO545 Spring Term 2020 Assessment 2

You should submit your solutions to Moodle in a single Erlang file by Friday 28 February, 23:55.

1. This question takes the example of fundraising through *crowdfunding*.

Erlang Enterprises (EE) are trying to raise a sum of money to support their further development. They ask for bids from people, which have the form `{atom, int}`, so that, joe bidding £1000 will be represented by the pair `{joe,1000}`.

All the bids (in the order in which they arrived, earliest first) are given in a list, e.g.
`[ {joe,1000}, {robert,3000}, {grace,5000}, {ada, 500} ]`
We call this list the *bid list*.

(a) Bids should not be zero or negative. Write a function `pos_bids` which takes a bid list `Bids` and returns a list leaving out any bids that do not contain a positive number as the bid amount. (You may assume that all the bids do contain a number in the second place, and so you don't have to check for that.) [6 marks]

(b) Suppose that EE have set a threshold for the crowdfunding exercise, and that if the total of the bids is less than this amount then all should be returned to the bidders. Write a function success that takes `Bids` and `Threshold` as arguments and checks whether the sum of all the bid amounts in the list `Bids` is at least the value given in the `Threshold` parameter. [6 marks]

(c) Suppose that the list `Bids` does contain enough to exceed the `Threshold`, as checked in part (b). Write a function `winners` that takes `Bids` and `Threshold` as arguments, and returns the list of bids that have been successful. These will be taken from the front of the list until the `Threshold` is exceeded.

For example, if `Bids` is the list given at the start of the question, then the result of `winners(5000, Bids)` will be the list
`[ {joe,1000}, {robert,3000}, {grace,1000} ]`

Note here that `grace`'s bid has been *lowered* to make the total precisely the threshold value, which is `5000` here. [8 marks]

2. This question concerns strings, like `"foo"` and `"football"`.

(a) Define a function `init` in Erlang that takes two strings and returns `true` if the first string is an initial segment of the other: in other words, if the first string can be extended to make the second. For example,
`init("foo","football")` should be `true`, and
`init("foo","ballfoot")` should be `false`. [4 marks]

(b) Define a function `drop` that takes an integer `N` and a string `St`, and which returns the string `St` with the first `N` elements dropped, if the string has that many elements (and as many as possible otherwise). For example,
`drop(2,"football")` should be `"otball"`, and
`drop(12,"football")` should be `""`. [4 marks]

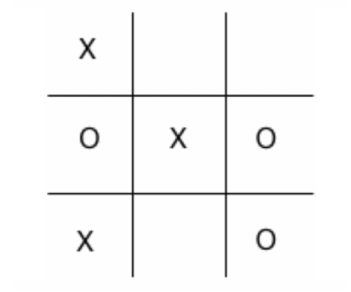
(c) Using the functions `init` and `drop`, or otherwise, define a function `subst` that takes three strings, `Old`, `New` and `St`. The function returns a string in which the *first* occurrence of `Old` is replaced by `New`; if it doesn't occur, then the string is returned unchanged. For example:
`subst("foo","bar","football")` should be `"bartball"`, and
`subst("foo","bar","ballfoot")` should be `"ballbart"`, and
`subst("foo","bar","footfoot")` should be `"bartfoot"`. [8 marks]

Simon Thompson 9 February 2020

(d) How would you modify your answer to (c) so that *all* occurrences of `Old` are replaced by `New`? How would you modify it so that only the *last* occurrence of `Old` was replaced? [4 marks]

3. This question concerns the game of *noughts and crosses.*

We can represent the state of a noughts and crosses board by a list of three lists, one for each line of the board. For instance, the board



is represented by `[[x,b,b],[o,x,o],[x,b,o]]` where `x`, `b`, `o` are atoms (and `b` stands for "blank").

(a) Define a function `isxwin` in Erlang that will take a line of a board and return a Boolean saying whether or not the line is a winning line of crosses, i.e. it consists of three crosses. [2 marks]

(b) Using your solution to (a) or otherwise, define a function `linexwin` in Erlang that takes a board and returns a Boolean if the board contains a winning line of crosses. [4 marks]

(c) Define a function `pick` in Erlang that takes an integer `N` and a list `Xs` and returns the `N`th element of the list, starting counting from `0`. For example

```
pick(0,[a,b,c]) = a
pick(2,[a,b,c]) = c
```

You can assume that the function is called with a value of `N` that makes sense for `Xs` (in the case of the list `[a,b,c]` it is 0, 1 or 2). [6 marks]

(d) Using your answer to (c) or otherwise, define a function `wincol` in Erlang that takes a board and returns true if the board contains a winning column. For example, the board shown at the start of the question does not contain a winning column, but the board

```
[[x,o,b],[o,o,x],[x,o,o]]
```

does (the middle column is a win for `o`). [8 marks]