CSCI 342

Group Program

You may work with other students
(list them in your documentation)
Choose **one** project to work on.

Projects should be substantially original code, but you are welcome to use properly cited code for minor parts of the code.

Project 1: Customer Handling

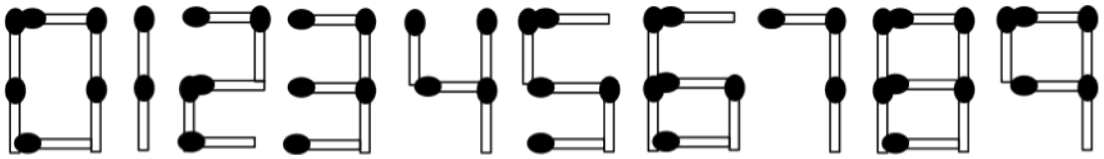
Write a program which reads a file of frequent customers for a business (including at least their names, postal addresses, email addresses, and date of last visit) into a binary search tree. The program should allow for insertion of new customers and deletion of former customers. When the program is ready to end, write the revised list to the customer file. Please ensure that your program will correctly work multiple times (so you need to make sure your program can read in the file it writes out), and that whatever process you use for reading and writing the file will produce a genuine binary tree (instead of the equivalent of a linked list).

Project 2: Matchsticks

This problem comes from the Digikey DKC3 competition, 2012.

We can make digits with matches as shown below. We shall only make numbers greater than or equal to 0, so no negative signs should be used. For instance, if you have 3 matches then you can only make the numbers 1 or 7 [2 numbers]. If you have 4 matches, then you can make 1, 4, 7, or 11 [4 numbers]. Superfluous leading zeros are not allowed (so 0001, 042, etc are illegal - 0 is the only legal number to begin with 0).

Write a program that inputs the number of matches available (up to 80), and outputs the number of different numbers representable using up to that many matches.



Project 3: Princess

This problem is an adaptation of one of the Digikey's DKC3 problems from 2011.

You are on an epic quest to save the Princess (and grab the treasure) who is locked in a tower (T). To get there, you will have to cross some combination of roads (R), which take one hour to cross; fields (F), which take 2 hours to cross; and moutanins (M), which take 3 hours to cross. Your program will have to determine the fastest route to the princess from your castle (C).

INPUT:
The test file will hold a square map as follows: The first line will contain an integer (N). The next N lines will contain a string of N characters: C, R, F, M, or T.

OUTPUT:
Your program should print the fastest route, and the time taken to to make your way from the castle to the Princess's tower along the fastest route.

SAMPLE INPUT:

```
7
RTFMMMF
FMFMMMM
RRFFMRR
FRFFFMR
FRMMFFR
FFMMMMR
RRRCRRR
```

SAMPLE OUTPUT:
West, west, north, north, north, north, north.
10 hours to reach the Princess.