

ES 314 Advanced Programming, Modeling and Simulation

Fall 2012

Home Work # 6

Work on one of the problems from the list below. Due: Dec 12, 2012

Problem 1:

Write a program that performs arithmetic operations on numbers with unlimited number of digits. Such numbers will be represented as strings. The operations to be supported are:

- (a) Addition (b) subtraction (c) multiplication

The inputs as well as outputs can be negative. See test cases below:

```
>> add('72487287743748374374095958263726',  
'9082710918735479376236109929')
```

```
ans =
```

```
'72496370454667109853472194373655'
```

```
>> mult('90717736460987373737373767376236267',  
'928288888748943849384938451625101')
```

```
ans =
```

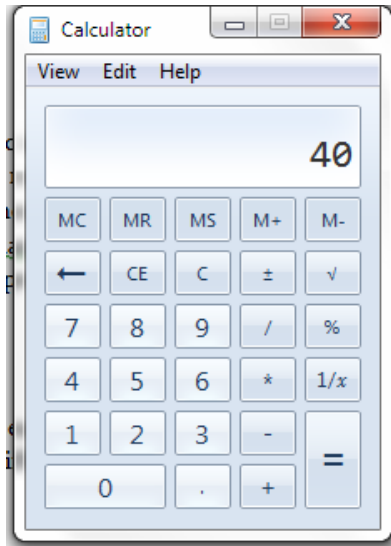
```
'84212266769189515301287365593553715765612524072820485189655083737967'
```

```
>> subtract('8190092188282818288', '871878387382738273871871122')
```

```
'-871878379192646085589052834'
```

Problem 2:

Extend the simple calculator you designed in HW 5 to a more realistic calculator that can compute the value of an expression with more than one operator. Also modify the program so that the operand can be entered using buttons for 0, 1, 2, ..., 9. For example, you use the calculator that comes with Windows OS as a model. But it is enough to implement buttons 0 to 9, +, -, *, /, =, and decimal point. (Decimal point will allow the user to enter a real number.)



Problem 3:

Many PDF files are stored in a file system containing many directories each of which has subdirectories and files etc. Some PDF files have multiple copies – in different directories. The goal of this problem is to write a program that takes as input the path to the home directory, and make a list of all PDF files, remove duplicates and create an index of all the words that appear in the titles of documents. Thus for each word that (other than article, prepositions etc.) that appears in the document, it should list all the books that contains the word. For example, a word like “Programming” should list all books containing this word. To make this a useful program, you should try to apply stemming – a technique that maps all the related words such as program, programs, programming, programmed etc. as a single root word. Your index list would contain the stemmed words. Your program should be flexible in applying the same process to other file extensions such as .mat, .cpp or .doc etc.

Problem 4:

Write a program that takes as input an audio clip, and plays the audio. While the audio plays, the program should show the graphical representation of the audio (with the part of the clip corresponding to five seconds of audio and a moving dot that shows the sample that currently plays. So the animation you create will show the dot moving from left end of the screen to the right end in about 5 seconds at which point the plot of the graph for the next 5 seconds will be displayed on the screen.

Problem 5:

Write a program that takes as input an image I, and the coordinates of a polygon, and produces a new image by retaining only the interior of the polygon in the image, and replacing the exterior by white pixels (or more generally by pixels of a color specified by the user). As a second step, allow the user to specify this polygon by clicking on the successive vertices on a GUI.