Backtracking

Serge Kruk

November 19, 2024

1 Problem statement

A classic puzzle of recreational mathematics is the cryptarithm, where the solver is given a math problem in words and must systematically substitute digits for the letters of the puzzle to form a valid calculation. For instance, the famous cryptarithm SEND + MORE = MONEY is solved as M=1, Y=2, E=5, N=6, D=7, R=8, S=9 and O=0, giving 9567 + 1085 = 10652. Cryptarithms were invented by H. E. Dudeney in the July 1924 edition of Strand Magazine.

Your task is to write a backtracking solution to this sort of puzzle. Start small, with a version that does not do anything fancy other than exploring the whole search tree, then try to improve it by finding ways to prune the search tree.

Once you have worked out the solution to "SEND + MORE = MONEY", for which I give you the solution, apply your code to the following "THREE + THREE + TWO + TWO + ONE = ELEVEN". Report the solution and contrast the time to solve both puzzles.

1.1 Brute force solution

Here is a solution to give a baseline to which you can compare your own solution. Yours should be faster; in your report, state explicitely the improvement in speed.

```
def solve_money():
    for s in range(1, 10):
        for e in range(0, 10):
            for n in range(0, 10):
                for d in range(0, 10):
                    for m in range(1, 10):
                        for o in range(0, 10):
                            for r in range(0, 10):
                                for y in range(0, 10):
                                    if distinct(s, e, n, d, m, o, r, y):
                                        send = 1000 * s + 100 * e + 10 * n + d
                                        more = 1000 * m + 100 * o + 10 * r + e
                                        money = 10000 * m + 1000 * o + 100 * n + 10 * e + y
                                        if send + more == money:
                                            return send, more, money


def distinct(*args):
    return len(set(args)) == len(args)

from time import time
start=time()
print(solve_money())
end=time()
print("It took {0:4.2f} seconds".format(end-start))
```

```
(9567, 1085, 10652)
It took 35.57 seconds
```