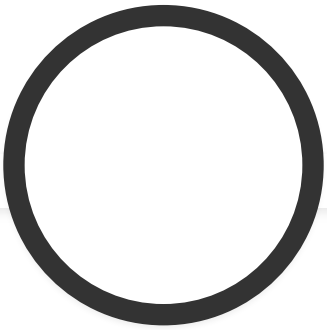
coding hangover

eat pray code :)



Spoj(ONEZERO) – Ones and zeros

Posted on May 1, 2015 by vishnujayvel

Problem link: http://www.spoj.com/problems/ONEZERO/

Suppose the number that you want is X. X mod N = 0.

So you need to store only N states i.e. 0 to n-1. Start with 1.

Implement bfs approach. If the current state modulo is Y, append 0 to it i.e calculate Y*10 + 0 and find its modulo which will lead you to a new modulo state.

Similary append 1 to Y. i.e calculate Y*10 + 1 and find its modulo.

Example: if Y=11 append 0 to it to get 110 and append 1 to Y to get 111.

Have a parent array which will store the previous modulo state from which the current modulo state is reached. This parent array also acts as checkpoint to check if a modulo state is already visited or not.

Have a value array to store the value (1 or 0) that is appended to the parent modulo state to get the current modulo state.

Once the modulo state 0 is reached stop bfs and backtrack using parent array and value array to get the number (i.e smallest multiple of the number *n* consisting only of digits 1 and 0 beginning with 1).

```
#include <vector>
#include <list>
#include <map>
#include <set>
#include <deque>
#include <queue>
#include <stack>
#include <string>
#include <bitset>
#include <algorithm>
#include <functional>
#include <numeric>
#include <utility>
#include <sstream>
#include <iostream>
#include <iomanip>
#include <cstdio>
#include <cmath>
#include <cstdlib>
#include <ctime>
#include <cstring>
#include <climits>
#include <stdlib.h>
#include <stdio.h>
using namespace std;
#define REP(i,n) for(int i=0; i<n; i++)
#define FOR(i,st,end) for(int i=st;i<end;i++)
#define db(x) cout << (#x) << " = " << x << endl;
#define mp make_pair
#define pb push_back
#define mod 1000003
int parent[20005];
typedef long long int ll;

queue<int>q;
```

```
int temp,currentState;
int value[20005];
void solve(int n){
        q.push(1);
        parent[1]=0;
while(!q.empty()){
                currentState=q.front();
                q.pop();
if(currentState==0){
                        stack<int> s;
while(parent[currentState]){
                                s.push(value[currentState]);
                                currentState=parent[currentState];
                        }
                        s.push(1);
while(!s.empty()){
printf("%d",s.top());
                                s.pop();
                        }
printf("\n");
break;
                }
                temp=(currentState*10)%n;

if(parent[temp]==-1){
                        q.push(temp);
                        parent[temp]=currentState;
                        value[temp]=0;
                }
                temp=currentState*10+1;
                temp%=n;
if(parent[temp]==-1){
                        q.push(temp);
                        parent[temp]=currentState;
                        value[temp]=1;
                }
        }
}

int main(){
int t,n;
```

```
77  scanf("%d",&t);
78  while(t--){
79  while(!q.empty()){
80                          q.pop();
81                  }
82  REP(i,20000)
83                          parent[i]=-1;
84
85  scanf("%d",&n);
86  solve(n);
87          }
88  }
```

ONEZERO.cpp hosted with ❤ by GitHub view raw

Spoj-Aggressive cows
In "spoj"

Spoj(CRANO2) - Roommate Agreement
In "spoj"

Topcoder SRM 646 Div 2 TheGridDivTwo
In "topcoder"

spoj bfs ONEZERO