**1)** (10 pts) DSN (Recursive Coding)

A regular odometer of 6 digits counts from 000000 to 999999. A lucky odometer setting is one that contains the digit 7 at least twice. Complete the **recursive function** below so that the code below prints out each lucky odometer setting of n = 6 digits. In the recursive function, k represents the number of digits of the odometer already filled in.

```
#include <stdio.h>
#include <stdlib.h>

int numd(int* odometer, int n, int d);
void printlucky(int n);
void printluckyrec(int* odometer, int k, int n);
void print(int* odometer, int n);

int main() {
    printlucky(6);
    return 0;
}

void printlucky(int n) {
    int* odom = malloc(n*sizeof(int));
    printluckyrec(odom, 0, n);
    free(odom);
}

void printluckyrec(int* odometer, int k, int n) {

    if (k == n){                                // 1 pt
        if (numd(odometer, n, 7) >= 2)          // 2 pts
            print(odometer, n);                 // 1 pt
        return;                                 // 1 pt (or an else...)
    }

    for (int i=0; i<10; i++) {                  // 1 pt
        odometer[k] = i;                        // 1 pt
        printluckyrec(odometer, k+1, n);        // 3 pts
    }
}

int numd(int* odometer, int n, int d) {
    int res = 0;
    for (int i=0; i<n; i++)
        res += (odometer[i] == d);
    return res;
}

void print(int* odometer, int n) {
    for (int i=0; i<n; i++)
        printf("%d ", odometer[i]);
    printf("\n");
}
```