```
/* Author : Prateep  */
#include <iostream>
#include <cstdio>
#include <cmath>
#include <cstring>
#include <cstdlib>
#include <cctype>
#include <algorithm>
#include <iterator>
#include <map>
#include <vector>
#include <list>
#include <set>
#include <queue>
#include <cassert>
#include <deque>
#include <stack>
#include <bitset>
#include <functional>
#include <numeric>
#include <utility>
#include <sstream>
#include <iomanip>
#include <cstring>
#include <cmath>
#include <ctime>
#include <time.h>

using namespace std;

#define LET(x,a)        __typeof(a) x(a)
#define IFOR(i,a,b)     for(LET(i,a);i!=(b);++i)
#define EACH(it,v)      IFOR(it,v.begin(),v.end())
#define FOR(i,a,b)      for(int i=(int)(a) ; i < (int)(b);i++)
#define REP(i,n)        FOR(i,0,n)
#define SZ              size()
#define PB              push_back
#define PF              push_front
#define V(x)            vector< x >
#define two(X)          (1<<(X))
#define twoL(X)         (((int64)(1))<<(X))
#define contain(S,X)    (((S)&two(X))!=0)
#define containL(S,X)   (((S)&twoL(X))!=0
#define swap(x,y)       x ^=y ^= x^=y
#define ALL(v)          (v).begin(),(v).end()

const double pi=acos(-1.0);
const double eps=1e-11;
const int oo=0x3f3f3f3f;
template<class T> inline void checkmin(T &a,T b){if(b<a) a=b;}
template<class T> inline void checkmax(T &a,T b){if(b>a) a=b;}
template<class T> inline T sqr(T x){return x*x;}
typedef pair<int,int> ipair;
template<class T> inline T lowbit(T n){return (n^(n-1))&n;}
template<class T> inline int countbit(T n){return (n==0)?0:
(1+countbit(n&(n-1)));}

typedef long long       ll;
typedef V(int)          VI;
typedef V(VI)           VII;
typedef V(string)       VS;
typedef long long       int64;
typedef unsigned long long uint64;
typedef long double     LD;
typedef pair<int,int>   PI;

//#define LocalHost
const int denom[8] = {1,4,7,13,28,52,91,365};

int solve_greedy(int S)
{
  int res = 0, p = 7;
  while(S > 0 && p >= 0) {
    int v = denom[p];
    while(S >= v) {
      ++res;
      S-=v;
    }
    --p;
  }
  return res;
}

int solve_dp(int S)
{
  int* minm = new int[S+1];
  memset(minm, oo, (S+1)*sizeof(int));
  minm[0] = 0;

  REP(i,S+1) REP(j,8) {
    if(denom[j] <= i && minm[i-denom[j]] + 1 < minm[i]) {
      minm[i] = minm[i-denom[j]] + 1;
    }
  }
  return minm[S];
}

int numCoinsOfEachType[8];
int numCoinsSoFar = 0;
int best = -1;
bool ok = false;
int bestCoinsOfEachType[8];

bool check(int change, int numCoinsRet)
{
  if(change < 0) return false;
  else if(change == 0) return true;
  else {
    return !(ok && numCoinsRet + 1 >= best);
  }
}

void solve_naive(int S)
{
  REP(k,8) {
    int change = S - denom[k];
    if( check(change, numCoinsSoFar + 1) ) {
      //cout << change << endl;
      if(change == 0) {
        if(!ok || numCoinsSoFar + 1 < best) {
          best = numCoinsSoFar + 1;
          ok = true;
        }
      } else {
        ++numCoinsSoFar;
        numCoinsOfEachType[k] += 1;

        solve_naive(change);

        --numCoinsSoFar;
        numCoinsOfEachType[k] -= 1;
      }
    }
  }
}

void print_mismatch(int a, int b, int k)
{
    if(a != b)
        cout << k << " : (" << a << " , " << b << ")\n";
    return;
}

int main(int argc, char* argv[])
{
#ifdef LocalHost
    freopen("input.txt","r",stdin);
    freopen("output.txt","w",stdout);
#endif

    if(argc < 2)
    {
      cerr << "Usage : " << argv[0] << " type[ Greedy = 1, DP = 2, Naive = 3,
Mismatches(Greedy vs. DP)] = 4\n";
      return -1;
    }

    int type = atoi(argv[1]);

    ios::sync_with_stdio(false);
    cout << "Denominations available : ";
    REP(i,8) cout << denom[i] << (i == 7 ? "\n" : " ");
    clock_t beg = clock();
    if(type == 1) {
        cout << "Greedy algorithm\n";
        cout << "------------------------\n";
    } else if(type == 2) {
        cout << "DP algorithm\n";
        cout << "------------------------\n";
    } else if(type == 3) {
        cout << "Naive algorithm\n";
        cout << "------------------------\n";
    } else if(type == 4) {
        cout << "Printing mismatches (Greedy vs DP). Range : [1-600]\n";
        cout << "------------------------\n";
    } else {
      cerr << "Unknown option.\n";
      return -1;
    }
    FOR(k,1, 600) {
//    for(int k = 10; k <= 55; k+= 5) {
      int a = -1, b = -1, c = -1;
      if(type == 1) {
        a = solve_greedy(k);
        cout << "Denomination: " << k << " Result : " << a << endl;
      } else if(type == 2) {
        b = solve_dp(k);
        cout << "Denomination: " << k << " Result : " << b << endl;
      } else if(type == 3) {
         best = -1;
         numCoinsSoFar = 0;
         ok = false;
         solve_naive(k);
         c = best;
         cout << "Denomination: " << k << " Result : " << c << endl;
      } else if(type == 4) {
         a = solve_greedy(k);
         b = solve_dp(k);
         print_mismatch(a,b,k);
      }
    }
    clock_t en = clock();
    printf("Time required : %.9lf\n",double((en - beg)/CLOCKS_PER_SEC) );
    return 0;
}
```