■ Changing Money

How many ways can you make change for 1 dollar, using 1−,10−,25−,50−,and 100−centcoins?

In this section, $a = \{1, 5, 10, 25, 50, 100\}$ is the list of coin denominations.

If we try to solve this with a recurrence relation, we could reason that the number of ways $f(n)$ of making change for n cents is the sum, over all coins we could subtract from n, of the number of ways of making the smaller amount, i.e.

$$f(k) = \begin{cases} 0, & k < 0 \\ 1, & k = 0 \\ \sum_{i=1}^{|a|} f(k - a_i), & k > 0 \end{cases}$$

Unfortunately, this gives 8,577,874,824,929 ways to make change for a dollar, which cannot be correct as there are only 444,793 integer partitions of 100.

The error in reasoning should be found with the first incorrect result.

k	$f(k)$
1	1
2	1
3	1
4	1
5	2
6	3

There are only two ways to make change for 6 cents, but $f(6) = 3$. f gives the number of ways of making 5 cents and adding a penny, plus the number of ways of making 1 cent and adding a nickel. When changing money, we would like “1 penny, 1 nickel” and “1 nickel, 1 penny” to be considered as just one way of making change for 6 cents.

We can fix this by adding each type of coin in turn; we will not be able to count different orderings twice if there are no different orderings.

By increasing the dimension of the recurrence relation, we keep track of enough information to actually solve the problem. Let $f(n, k)$ be the number of ways of making change for n cents, using only the first k types of coin.

$$f(n, k) = \begin{cases} 0, & k < 1 \text{ or } n < 0 \\ 1, & n = 0 \\ f(n, k-1) + f(n - a_k, k), & \text{else} \end{cases}$$

The boundary cases assert that there is exactly one way to make change for 0 cents using a positive number of coins, but there are no ways to make change for negative amounts, or using no coins.

In the non−boundarycase, the number of ways of making change using the first k types of coin is the number of ways that don’t use that coin, $f(n, k-1)$, together with the number of ways that do, $f(n - a_k, k)$.

The purpose of describing this, is to give a method to solve this problem by hand. The previous recurrence relation is still too awkward for direct computation, so we present two tricks.

First, the number of ways of making change using only the first type of coin, (which is always assumed to be 1 in this sort of problem so that change can be made for any integral amount), is always exactly 1. Therefore we consider only multiples of 5, which reduces the number of relevant values for n to 21, which is managable by hand.

The following table shows that there are 293 ways to make change for a dollar.

n	$f(n, 1)$	$f(n, 2)$	$f(n, 3)$	$f(n, 4)$	$f(n, 5)$	$f(n, 6)$
0	1	1	1	1	1	1
5	1	2	2	2	2	2
10	1	3	4	4	4	4
15	1	4	6	6	6	6
20	1	5	9	9	9	9
25	1	6	12	13	13	13
30	1	7	16	18	18	18
35	1	8	20	24	24	24
40	1	9	25	31	31	31
45	1	10	30	39	39	39
50	1	11	36	49	50	50
55	1	12	42	60	62	62
60	1	13	49	73	77	77
65	1	14	56	87	93	93
70	1	15	64	103	112	112
75	1	16	72	121	134	134
80	1	17	81	141	159	159
85	1	18	90	163	187	187
90	1	19	100	187	218	218
95	1	20	110	213	252	252
100	1	21	121	242	292	293

The rule for updating is that given by the recurrence; for example, $187 = f(90, 4) = f(90, 3) + f(65, 4)$. This table is still too awkward for hand computation, so we present a second trick.

By using the equivalent recurrence relation

$$f(n, k) = \begin{cases} 0, & k > |a| \text{ or } n < 0 \\ 1, & n = 0 \\ f(n, k+1) + f(n - a_k, k), & \text{else} \end{cases}$$

the resulting table is sparse, and the numbers are smaller.

n	$f(n, 1)$	$f(n, 2)$	$f(n, 3)$	$f(n, 4)$	$f(n, 5)$	$f(n, 6)$
0	1	1	1	1	1	1
5	2	1				
10	4	2	1			
15	6	2				
20	9	3	1			
25	13	4	1	1		
30	18	5	1			
35	24	6	1			
40	31	7	1			
45	39	8	1			
50	50	11	3	2	1	
55	62	12	1			
60	77	15	3			
65	93	16	1			
70	112	19	3			
75	134	22	3	2		
80	159	25	3			
85	187	28	3			
90	218	31	3			
95	252	34	3			
100	293	41	7	4	2	1

■ Dynamic Programming

This is a specific instance of a more general technique known as dynamic programming, which is often effective in attacking combinatorial problems. It works best with a computer, but with a few tricks, it can be employed by hand.

The general idea of solving a problem this way is to find a recursive solution in which many subcases overlap. The answers to these subproblems can be stored in a table, so that they are never solved more than once.

For example, the coin−changing problem has a great deal of overlap, since we add different coins to the same total many times. In fact, a straightforward computer implementation of the recurrence relation requires 26,905 function calls to $f(n, k)$ in order to find the number of ways to make change for a dollar. But the answer above was found with only 68 computations.

The standard example is the computation of the Fibonacci numbers,

$$f(0) = f(1) = 1$$
$$f(n) = f(n-1) + f(n-2)$$

A straightforward C program to compute $f(50)$,

```
f(n) {return n<3?1:f(n−1)+f(n−2);}
main(void) {printf("%d\n",f(50));}
```

makes 25,172,538,049 calls to $f(n)$; only 50 are necessary.

■ Coin change revisited

The dynamic programming approach to this problem does not work for very large values. By throwing some analytic stuff at it, we overcome this problem. This section follows the relevant part of Graham, Knuth, and Patashnik's *Concrete Mathematics*.

The generating function for the problem is $F(x) = \frac{1}{1-x}\,\frac{1}{1-x^5}\,\frac{1}{1-x^{10}}\,\frac{1}{1-x^{25}}\,\frac{1}{1-x^{50}}\,\frac{1}{1-x^{100}}$. First we would like to simplify by considering only multiples of 5.

$$F(x) = \frac{1+x+x^2+x^3+x^4}{1-x^5}\,\frac{1}{1-x^5}\,\frac{1}{1-x^{10}}\,\frac{1}{1-x^{25}}\,\frac{1}{1-x^{50}}\,\frac{1}{1-x^{100}}$$
$$= (1+x+x^2+x^3+x^4)\left(\frac{1}{(1-x^5)^2}\,\frac{1}{1-x^{10}}\,\frac{1}{1-x^{25}}\,\frac{1}{1-x^{50}}\,\frac{1}{1-x^{100}}\right)$$
$$= (1+x+x^2+x^3+x^4)\,G(x^5)$$
$$G(x) = \frac{1}{(1-x)^2}\,\frac{1}{1-x^2}\,\frac{1}{1-x^5}\,\frac{1}{1-x^{10}}\,\frac{1}{1-x^{20}}$$

The nth coefficient of the power series of $F(x)$ is the $\lfloor n/5 \rfloor$th coefficient of $G(x)$. $G(x)$ is $(1-x^{20})^{-6}$ (which has a nice power series) times a polynomial, $A(x)$.

$$G(x) = \frac{A(x)}{(1-x^{20})^6}$$
$$A(x) = \frac{(1-x^{20})^6}{(1-x)^2\,(1-x^2)\,(1-x^5)\,(1-x^{10})\,(1-x^{20})}$$
$$= (1+x+x^2+\cdots+x^{19})^2\,(1+x^2+\cdots+x^{18})\,(1+x^5+x^{10}+x^{15})\,(1+x^{10})$$
$$= 1+2x+4x^2+6x^3+9x^4+\cdots+24x^{74}+18x^{75}+13x^{76}+9x^{77}+6x^{78}+4x^{79}+2x^{80}+x^{81}$$

The coefficients of $A(x)$ are

1, 2, 4, 6, 9, 13, 18, 24, 31, 39, 50, 62, 77, 93, 112, 134, 159, 187, 218, 252, 287, 325, 364,
406, 449, 493, 538, 584, 631, 679, 722, 766, 805, 845, 880, 910, 935, 955, 970, 980, 985,
985, 980, 970, 955, 935, 910, 880, 845, 805, 766, 722, 679, 631, 584, 538, 493, 449, 406,
364, 325, 287, 252, 218, 187, 159, 134, 112, 93, 77, 62, 50, 39, 31, 24, 18, 13, 9, 6, 4, 2, 1

Now

$$G(x) = A(x)\,\frac{1}{(1-x^{20})^6} = A(x)\sum_{k=0}^{\infty}\binom{k+5}{5}x^{20k}$$

If $G(x) = \sum_{k=0}^{\infty} g_k\,x^k$, and $A(x) = \sum_{k=0}^{81} A_k\,x^k$, then

$$g_{20q+r} = \sum_{\substack{j,k=0 \\ 20q+r=20k+j}}^{\infty} A_j \binom{k+5}{5}$$

which is essentially a closed form, as for any given $n = 20\,q + r$, there are at most 5 non−zeroterms. This is a *Mathematica* program to compute the number of ways, $f(n)$, of making change for n cents using pennies, nickels, dimes, quarters, half−dollars, and dollar coins.

```
a[j_] := Coefficient[Apart[(1-x^20)^5 / ((1-x)^2 (1-x^2) (1-x^5) (1-x^10))], x, j];
f[n_] := Block[{v = ⌊n / 5⌋, q = ⌊v / 20⌋, r = v~Mod~20},
  Sum[a[r + 20 j] Binomial[q + 5 - j, 5], {j, 0, 4}]]
```

Here is the code at work.

```
Timing[{f[100], f[10^102], Short[f[10^10002]], Short[f[10^100002]]}]
```

```
{2.641 Second, {293,
  13333333333333333333333333333333333333333333333333333333333333333333333333333333333
   3333333333333333398333333333333333333333333333333333333333333333333333333333333333
   3333333333333333333333333333333333445333333333333333333333333333333333333333333333
   3333333333333333333333333333333333333333333333333333413833333333333333333333333333
   3333333333333333333333333333333333333333333333333333333333333333333333354500000000
   0000000000000000000000000000000000000000000000000000000000000000000000000000000000
   000000001, 133333333333333333 <<49965>> 0000000000000000001,
  133333333333333333 <<499966>> 000000000000000001}}
```

In 2.641 seconds, we computed the number of ways to make change for a dollar (293, just as before), the number of ways to make change for 10 duotrigintillion dollars, as well as for 10^{10000} and 10^{100000} dollars. The last two are not shown in full, as they are very long; the number of ways to make change for 10^{100000} dollars is an integer with 500,002 digits.

The number of ways to make change for $10^{1,000,000}$ dollars is an integer 5,000,002 digits long; this may be a pattern. It is doubtful that anyone has bothered computing such things before, or that anyone cares.

■ Solving Recurrence Relations

There are strong connections between differential equations and difference equations. As with differential equations, recurrence relations can not usually be solved explicity when they are defined by non−linearequations.

Recurrence relations have intriguing properties. The following, up to and including the statement of Sarkovskii's theorem, is taken mostly verbatim from *Bendixson Criteria for Difference Equations* by C. Connell McCluskey. The proof is referenced to page 63 in *An Introduction to Chaotic Dynamical Systems* by R. L. Devaney.

Sarkovskii's ordering: the order of the odd integers greater than 1 is reversed, i.e. $3 \rhd 5 \rhd 7 \rhd \cdots$. Even multiples are added to the order by $2^n\, 3 \rhd 2^n\, 5 \rhd \cdots$, and $2^n\,(2\,i+1) \rhd 2^m\,(2\,j+1)$, if $m > n$ and $i,\ j > 0$. Powers of 2 are added in decreasing order so that

$$3 \rhd 5 \rhd \cdots \rhd 2\cdot 3 \rhd 2\cdot 5 \rhd \cdots \rhd 2^2\cdot 3 \rhd 2^2\cdot 5 \rhd \cdots \rhd 2^n\cdot 3 \rhd \cdots \rhd 2^3 \rhd 2^2 \rhd 2 \rhd 1$$

Theorem (Sarkovskii). If $f : \mathbb{R} \to \mathbb{R}$ is continuous, and has a periodic point of least period k, then if $k \rhd l$, f also has a point of least period l.

In particular, if the recurrence relation $a_0 = a_0$, $a_{n+1} = f(a_n)$ has any periodic orbit, then it must have an orbit of period 2.

If we allow random recurrence relations, i.e. $a_{n+1} = f(a_n)$, where $f(x) = g_i(x)$, where each g_i is chosen with probability p_i, then the result is a fractal if the g_i and p_i are chosen appropriately.

There are a number of techniques for solving recurrence relations. Generating functions usually work in simple cases. They may remind you of power−seriestechniques for solving differential equations.

Define the generating function $G(x) = \sum_{n=1}^{\infty} f(n)\,x^n$, where $f(n)$ is the nth Fibonacci number. We completely ignore questions of convergence; they may get in the way of solving the problem.

$$\begin{aligned}
\sum_{n=1}^{\infty} f(n)\,x^n &= \sum_{n=1}^{\infty} (f(n-1) + f(n-2))\,x^n \\
&= \sum_{n=1}^{\infty} f(n-1)\,x^n + \sum_{n=2}^{\infty} f(n-2)\,x^n \\
&= x \sum_{n=0}^{\infty} f(n)\,x^n + x^2 \sum_{n=0}^{\infty} f(n)\,x^n \\
-1 + G(x) &= (x + x^2)\,G(x) \\
-1 &= (x^2 + x - 1)\,G(x)
\end{aligned}$$

So that $G(x) = -1/(x^2 + x - 1)$. Applying the method of partial fractions,

$$\begin{aligned}
G(x) &= \frac{-1}{x^2 + x - 1} = \frac{1}{\left(x - \frac{1+\sqrt{5}}{2}\right)\left(x - \frac{1-\sqrt{5}}{2}\right)} \\
&= \frac{1}{\sqrt{5}} \left(\frac{1}{x - \frac{1+\sqrt{5}}{2}} - \frac{1}{x - \frac{1-\sqrt{5}}{2}} \right) \\
\sum_{n=1}^{\infty} f(n)\,x^n &= \frac{1}{\sqrt{5}} \left(\sum_{n=1}^{\infty} \left(\frac{1+\sqrt{5}}{2} \right)^n - \sum_{n=1}^{\infty} \left(\frac{1-\sqrt{5}}{2} \right)^n \right)
\end{aligned}$$

Since the two power series are equal, their coefficients must be the same;

$$f(n) = \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right)$$

Since $\frac{1-\sqrt{5}}{2}$ is small, $\frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n$ rounded to the nearest integer is equal to $f(n)$.