Problem 1–20: Let $f(n, m, k)$ be the number of strings of n 0's and 1's that contain exactly n 1's, no k of which are consecutive.

(a) Find a recurrence formula for f. It should have $f(n, m, k)$ on the left side , and *exactly three terms on the right.*

(b) Find, in simple closed form, the generating function

$$F_k(x, y) = \sum_{n,m\geq 0} f(n, m, k)x^n y^m, \quad k = 1, 2, 3, \ldots.$$

(c) Find an explicit formula for $f(n, m, k)$ from the generating function (this should involve only a single summation, of an expression that involves a few factorials).

Solution: (a) A legal string will be one satisfying the conditions of the problem. Consider the symbol in the nth position of a legal string of length n. If this is a 0 then the first $n-1$ positions can be filled with any legal string of length $n-1$ having m 1's. There are $f(n-1, m, k)$ of these.

If the nth symbol is a 1 then the first n positions can be filled with any legal string of length n with $m-1$ 1's except those that end in exactly $k-1$ 1's since then it is not legal to add a 1 to the end. The number of legal strings without this final restriction is $f(n-1, m-1, k)$ and the number satisfying this restriction is $f(n-k-1, m-k, k)$ since these end in the string $01\ldots1$, where there are $k-1$ 1's, and if this string of length k is deleted from the string, what is left is any legal string of length $n-k-1$ with $m-k$ 1's.

Putting this together, we have

$$f(n, m, k) = f(n-1, m, k) + f(n-1, m-1, k) - f(n-k-1, m-k, k). \qquad (1)$$

Notice that k is fixed in (1). From the definition, we need to have $k \geq 2$ before the restriction on consecutive 1's makes sense. We will assume $k \geq 2$ in the following.

Before continuing, we must examine the domain of validity of (1). Clearly we should define $f(n, m, k) = 0$ if $n < 0$ or if $m < 0$.

If we want (1) to hold for $n \geq 1$ then in particular we would need to have

$$f(1, 0, k) = f(0, 0, k) + f(0, -1, k) - f(-k, -k, k) = f(0, 0, k),$$

since the final two terms vanish. But clearly $f(1, 0, k) = 1$ since the set of strings described here is just the singleton $\{0\}$. Thus we must require that $f(0, 0, k) = 1$.

But then notice that (1) *does not* hold for $(m, n) = (0, 0)$ since the right side of (1) in this case is $0 + 0 - 0 = 0$ and not 1 as required.

We claim that (1) is also not valid for $(m, n) = (k, k)$. For if we examine the reasoning leading to (1) in the case $n = k$, we find that the final term in (1) should in fact be $f(k-k, m-k, k)$ and not $f(k-k-1, m-k, k)$ since the only string being considered is the string consisting of all 1's and so the statement that the strings of length $n-1$ that we must omit all end in $01\ldots1$ is incorrect. They in fact end in $1\ldots1$, where there are $k-1$ 1's. However, the difference between (1) and the correct formula is immaterial if $m < k$ since then both $f(k-k-1, m-k, k)$ and $f(k-k, m-k, k)$ vanish because of the negative second argument. The only case in which the distinction is important is if $m = k$ since in that case, the last term of (1) is $f(k - k - 1, 0, k) = 0$ rather than the correct last term $f(k - k, 0, k) = 1$. So (1) does fail if $(m, n) = (k, k)$. (Note that obviously $f(k, k, k) = 0$.)

(b) To obtain the recurrence, multiply (1) by $x^m y^n$ and sum over all $n, m \geq 0$ except for $(m, n) = (0, 0)$ and $(m, n) = (k, k)$. Taking into account the definition of $F_k(x, y)$ and noticing that the exceptional terms are easily evaluated, we find that this gives

$$F_k - f(0,0,k) = x(F_k - f(k-1,k,k)x^{k-1}y^k) + xy(F_k - f(k-1,k-1,k)x^{k-1}y^{k-1}) - x^{k+1}y^k F_k,$$

where we have omitted terms with either one of the first arguments of f being negative, since these all vanish. Since $f(0, 0, k) = 1$, $f(k - 1, k, k) = 0$ and $f(k - 1, k - 1, k) = 1$, we see that

$$(1 - x - xy + x^{k+1}y^k)F_k(x, y) = 1 - x^k y^k,$$

giving the simple explicit formula

$$F_k(x, y) = \frac{1 - x^k y^k}{1 - x - xy + x^{k+1}y^k}. \tag{2}$$

(c) Now we need to expand F_k from (b) into a series from which we can pick off the coefficient of $x^m y^n$. If we write the denominator as $(1 - xy) - x(1 - x^k y^k)$, we see that F_k has the form $b/(a - xb)$, where $b = 1 - x^k y^k$ and $a = 1 - xy$. We can expand $b/(a - xb)$ in a geometric series by writing it as $b/(a(1 - x(b/a)))$ and expanding in the usual way to obtain

$$\frac{b}{a - xb} = \sum_{j=0}^{\infty} x^j \frac{b^{j+1}}{a^{j+1}}.$$

Hence

$$F_k(x, y) = \sum_{j=0}^{\infty} x^j \left(\frac{1 - x^k y^k}{1 - xy}\right)^{j+1}. \tag{3}$$

We pause to notice that this has a combinatorial interpretation. Notice that the jth term of this sum is

$$x^j (1 + xy + x^2 y^2 + \ldots + x^{k-1} y^{k-1})^{j+1}. \tag{4}$$

In the second term of (4), the powers of x and y are the same. Since the power of x keeps track of the length of the string and the power of y keeps track of the number of 1's in the string, the term $x^i y^i$ here refers to a string of length i consisting of i 1's. So the sum inside the $(j + 1)$st power is the generating function of strings of length at most $k - 1$ consisting of all 1's. A typical term in the expansion of (4) is of the form $x^n y^m$ with $n - m = j$. Thus the parameter j simply describes the number of 0 's in the string. So (3) can be obtained directly by thinking of the possible ways that one can parse a legal string into $j + 1$ blocks of 1's of any legal length i.e. 0 to k, separated by j 0's. Since we allow the blocks of 1's to be of length 0, this still allows the string to begin or end in a 0 and for there to be a number of consecutive 0's.

For example, the string $s = 01110011011110101$ with $n = 17$, $m = 11$ (and hence $j = 6$) is legal for any $k \geq 5$. If we write the symbol 1^i for a string of i 1's (including 1^0 as the empty string, then we would parse s into blocks as $s = 1^0 01^3 01^0 01^2 01^4 01^1 01^1$.

Returning to the expansion (3), we can use the fact that x and y appear to the same power in (3) to conclude that

$$f(n, n-j, k) = [x^n] x^j \Big(\frac{1-x^k}{1-x}\Big)^{j+1}. \tag{5}$$

Think this over carefully. The power of y in any term of (4) is automatically exactly j less than the corresponding power of x so the term we are picking out in (5) is exactly the same as

$$[x^n y^{n-j}] x^j \Big(\frac{1-x^k y^k}{1-xy}\Big)^{j+1}.$$

We can expand the generating function in (5) by writing it as the product of

$$(1-x^k)^{j+1} = \sum_i (-1)^i \binom{j+1}{i} x^{ik}, \tag{6}$$

and

$$\frac{x^j}{(1-x)^{j+1}} = \sum_\ell \binom{\ell}{j} x^\ell, \tag{7}$$

using the two familiar generating functions for the binomial coefficients.

Multiplying gives

$$x^j \Big(\frac{1-x^k}{1-x}\Big)^{j+1} = \sum_{i,\ell} (-1)^i \binom{\ell}{j} \binom{j+1}{i} x^{\ell+ik}. \tag{8}$$

The coefficient of x^n in (8) is thus

$$f(n, n-j, k) = \sum_{\substack{i,\ell \\ \ell+ik=n}} (-1)^i \binom{\ell}{j} \binom{j+1}{i},$$

and now writing $n-j = m$ and using i as the summation index, with $\ell = n - ik$, we obtain the explicit formula

$$f(n, m, k) = \sum_i (-1)^i \binom{n-ik}{n-m} \binom{n-m+1}{i}, \tag{9}$$

where the sum is over all i for which the expression is non-zero. It can be restricted to $0 \leq i \leq [m/k]$ since for the first term to be non-zero requires $n - m \leq n - ik$ and for the second term to be non-zero requires $i \geq 0$.

Testing the formula.

The following Maple functions give two evaluations of $f(n, m, k)$, one based on the recursive definition (1) and the other on the explicit formula (8). (Don't forget the "option remember" in the first function or you will be waiting a long time for it to compute any given value! It keeps a table of values of $f(n, m, k)$ that have been computed in previous branches of the recursion so it does not have to repeat the computation down to the leaves on each branch of the ternary tree):

```
f:=proc(n,m,k) option remember;
    if m < 0 or n < 0 or n < m then 0
    elif m = 0 and n = 0 then 1
    elif m = k and n = k then 0
    else f(n-1,m,k)+f(n-1,m-1,k)-f(n-k-1,m-k,k)
    fi
end;

g:=proc(n,m,k) local i;
    sum((-1)^i*binomial(n-i*k,n-m)*binomial(n-m+1,i), i=0..floor(m/k))
end;
```

We can check that both procedures give the same value for various choices of (m, n, k). For example, $f(113, 29, 4) = 544687665653944995812970825$, where the computation using the recursive procedure takes about 19 times as long as the computation from the explicit formula.

The analysis of the complexity of the procedure g is quite easy, although the speed depends on exactly how the [m/k]+1 binomial coefficients are evaluated.

Without the "remember" table option, a lower bound on the complexity of f is easy but discouraging. Notice that the only values defined on the leaves of the tree are either 0 or 1, thus one reaches at least $f(n, m, k)$ leaves in order to have the sum add up to $f(n, m, k)$. So for the choice $(m, n, k) = (113, 29, 4)$, one would reach at least 544687665653944995812970825 leaves (and this is certainly an underestimate since it ignores all those leaves which have the value 0). This would take a long time! So it is certainly important to store the values of $f(n, m, k)$ that one reaches in the recursion, which is what Maple's "option remember" does. If we write n_0 and m_0 for the initial values of the variables n and m then it is clear that in computing $f(n_0, m_0, k)$ using the recursion, the only values of (n, m) that one encounters satisfy $-k \leq n < n_0$ and $-k \leq m \leq m_0$ so the computation using the remember table option contains at most $(n_0 + k)(m_0 + k + 1)$ entries for which the value of $f(n, m, k)$ needs only to be computed once.