"Gift Distribution" Problems[1]

The general setting: N gifts for n children. The gifts can be different or identical; all children are different. If $N \le n$ (more children than gifts), then there could be a restriction from above: *at most* k gifts per child, $0 < k \le N$ and $k = N$ means no restrictions; if $N \ge n$, then there could be a restriction from below: *at least* m gifts per child, $N \ge mn$ and $m = 0$ means no restrictions.[2]

The case $N \le n$.
(I). Different gifts

(a) $k = 1$: At most one gift per child. The total number of ways is $\binom{n}{N} \cdot N!$, where $\binom{n}{N}$ counts the ways to select the children who receive a gift, and $N!$ counts the ways to order the gifts.

(b) $k = 2$: At most two gifts. Here, a more careful case-by-case analysis is necessary, looking at integer partitions of N in parts at most 2. The counting procedure then includes (i) Selecting children who receive gifts; (ii) From those who receive gifts, selecting those who receive two gifts; (iii) Counting the ways to order the gifts, keeping in mind that, for those who get two gifts, the order of those two gifts does not matter.
For example, if $N = 7$ and $n = 10$, then

$$7 = 1+1+1+1+1+1+1 = 2+1+1+1+1+1 = 2+2+1+1+1 = 2+2+2+1,$$

and the corresponding total number of ways

$$\binom{10}{7}7! + \binom{10}{6}\binom{6}{1}\frac{7!}{2!} + \binom{10}{5}\binom{5}{2}\frac{7!}{2!2!} + \binom{10}{4}\binom{4}{3}\frac{7!}{2!2!2!} = \frac{3 \cdot 11!}{16}.$$

(c) $k = N$: no restrictions. Here, the answer is n^N, as any of the N gifts can go to any of the n children.

(II). Identical gifts. If g_i is the number of gifts going to child number i, $i = 1, \ldots, n$, then the problem is about counting the number of non-negative integer solutions of the equation

$$g_1 + \ldots + g_n = N, \tag{1}$$

possibly with a restriction $g_i \le k$.

(a) $g_i \le 1$: At most one gift per child. Then the total number of ways is $\binom{n}{N}$.

(b) $g_i \le 2$: At most two gifts per child. Here, the answer is similar to the case $k = 2$ above, except that there is no need to count the orderings of the gifts. For example, if $N = 7$ and $n = 10$, then total number of ways to distribute the gifts is

$$\binom{10}{7} + \binom{10}{6}\binom{6}{1} + \binom{10}{5}\binom{5}{2} + \binom{10}{4}\binom{4}{3} = 4740.$$

(c) $g_i \le N$: no restrictions. In this case, we count the total number of non-negative solutions of (1), which is $\binom{N+n-1}{n-1}$.

The case $N \ge n$.
(I). Different gifts. The total number of ways, without any restrictions, is still n^N. When the lower bound m is small, then the corresponding number is easier to compute by counting the number of ways that violate the restriction [using inclusion-exclusion], and then subtracting the number from n^N. For example, if $m = 1$, then the answer is

$$n^N - \sum_{\ell=1}^{n-1}(-1)^{\ell+1}\binom{n}{\ell}(n-\ell)^N,$$

where ℓ represents the number of children who got no gifts.
For $m > 1$, the easiest way is to ask a compute to do the counting.

---

[1] Sergey Lototsky, USC. Updated September 3, 2022

[2] Further generalizations, not discussed here, can allow different k or m for different children, and some, but not all, of the N gifts to be identical.

The following is a MatLab script, written by a Math 407 student,[3] to count the number of ways to distribute $N =$ `gifts` to $n =$ `kids` with at least $m =$ `min` gifts per child, when the gifts are different.

```
% Distributing gifts to children
clear all

% Define Parameters
gifts=20;       % # of distinct gifts
kids=7;         % # of distinct children
min=1;          % Minimimum # of gifts per child

%Part 1: Stars and Bars
% Generate array from 1 to the number of positions in "Stars and Bars"
vector=1:(kids+gifts-1-min*kids);

% Generate matrix of all possible positions of kids-1 "stars"
combos=nchoosek(vector,kids-1);

%Convert Star Positions to Bar Lengths, add back minimum # of gifts
%bars(i,j) represents number of gifts for jth child in ith distribution

size=size(combos); % size(1)=# of solutions, size(2)=# of stars=kids-1

bars=zeros(size(1),size(2)+1); %Initialize matrix of bar lengths
for ii=1:size(1)
    bars(ii,1)=combos(ii,1)-1+min;
    % First bar = First star - 1
    for jj=2:(size(2))
        bars(ii,jj)=combos(ii,jj)-combos(ii,jj-1)-1+min;
        % Bar length = distance between successive stars
    end
    bars(ii,size(2)+1)=(kids+gifts-1-min*kids)-combos(ii,size(2))+min;
    % Last bar = distance between end of problem and final star
end

%Part 2: Add terms for each Grouping: gifts!/(x1!*x2!*...*xn!)
sum=0;
for ii=1:size(1)
    term=factorial(gifts); % Gifts!
    for jj=1:(size(2)+1)
        term=term/factorial(bars(ii,jj)); %Divide by each factorial
    end
    sum=sum+term; %Add terms
end
disp(sum)
```

(II). Identical gifts. Once again, we are counting the number of non-negative solutions of (1), but now the restriction is of the form $g_i \geq m$. As a result, with $p_i = g_i - m$, the problem is reduced to counting the number of non-negative solutions of

$$p_1 + \ldots + p_n = N - nm,$$

and the corresponding answer is $\binom{N - mn + n - 1}{n - 1}$.

[3] J. G., Spring 2017