The twelvefold way so far:

<table>
<thead>
<tr>
<th>$f$ is...</th>
<th>arbitrary</th>
<th>injective</th>
<th>surjective</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \rightarrow X$</td>
<td>$</td>
<td>X</td>
<td>^{</td>
</tr>
<tr>
<td>$L \rightarrow L$</td>
<td>$(</td>
<td>A</td>
<td>+ 1 \times 1 - 2)$</td>
</tr>
<tr>
<td>$U \rightarrow L$</td>
<td>$(</td>
<td>A</td>
<td>+ 1 \times 1 - 2)$</td>
</tr>
<tr>
<td>$L \rightarrow U$</td>
<td>${\lfloor</td>
<td>A</td>
<td>/2 \rfloor + \lfloor</td>
</tr>
<tr>
<td>$U \rightarrow U$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$= (1 \times 1, \{i \times 1\})$
More on Stirling numbers of 2nd kind:

(Recall: If \( n \in \mathbb{N} \) and \( k \in \mathbb{N} \), then \( \{n\}_k = \frac{1}{k!} \sum_{\pi \in \mathcal{P}(n,k)} \pi \) is the \# of all set partitions of \([n]\) into \( k \) parts.)

**Prop. 4.11.** Let \( n \in \mathbb{N} \) and \( k \in \mathbb{N} \),

\((2)\) we have \( \{0\}_k = \{n=0\} \),

\((6)\) \( \{k\}_0 = \{k=0\} \),

\((c)\) \( \{n\}_1 = 0 \) if \( k > 2 \),

\((d)\) \( \{n\}_k = \{n-1\}_k + k \{n-1\}_{k-1} \) if \( n > 0 \) and \( k > 0 \),

\((e)\) \( \{0\}_k = \sum_{j=0}^{\lfloor k/2 \rfloor} \binom{n}{j} \{k-1\}_j / k \).

**Proof.** Follows from Prop. 3.10, Prop. 3.11, Prop. 3.12. \( \Box \)
Pascal-like triangle for $\binom{n}{k}$:

- $n=0$: 1
- $n=1$: 0, 1
- $n=2$: 0, 1, 1
- $n=3$: 0, 1, 7, 15

$\binom{n}{k}$ for $k=0,1,2,3$:

- $k=0$: 1
- $k=1$: 0, 1
- $k=2$: 0, 1, 3
- $k=3$: 0, 1, 7, 1

Note: If $p$ is prime, then $p \mid \binom{p}{k}$ $\forall k \in \{2,3,...,p-2\}$.

Can you prove it? (Compare to $p \mid (k)$ $\forall k \in \{1,2,...,p-2\}$, but this is harder.)
4.6. U→U, AND INTEGER PARTITIONS

Idea: A U→U placement has the form

Since the boxes are indistinguishable, you can order them by decreasing number of balls, so the above placement is

Such a placement is thus encoded by how many balls each box has: here, the numbers are 3, 2, 2, 1, 0, 0, 0. The decreasing order makes this encoding unique.

Def. A partition of an integer n is a weakly decreasing list \((a_1, a_2, \ldots, a_k)\) of positive integers whose sum is n (that is, \(a_1 \geq a_2 \geq \cdots \geq a_k > 0\) and \(a_1 + a_2 + \cdots + a_k = n\)).
The integers $a_1, a_2, \ldots, a_k$ are called the parts of the partition.

Example: The partitions of 5 are

\[ (5), \ (4,1), \ (3,2), \ (3,1,1), \ (2,2,1), \]

\[ (2,1,1,1), \ (1,1,1,1,1). \]

The only partition of 0 is ( ).

Def: (So a partition is a weakly decreasing composition.) Let $n \in \mathbb{Z}$ and $k \in \mathbb{N}$. Then, $p_k(n)$ means the number of partitions of $n$ into $k$ parts (having $k$ parts).

Example:

\[ p_0(5) = 0, \quad p_1(5) = 1, \quad p_2(5) = 2, \quad p_3(5) = 2, \]

\[ p_4(5) = 1, \quad p_5(5) = 1, \quad \forall k > 5, \quad p_k(5) = 0. \]
Prop. 4.12. 

(a) \( p_k(n) = 0 \) \( \forall n < 0 \).
(b) \( p_k(n) = 0 \) if \( k > n \).
(c) \( p_0(n) = \lfloor n = 0 \rfloor \).
(d) \( p_1(n) = \lfloor n + 0 \rfloor \).
(e) \( p_k(n) = p_k(n-k) + p_{k-1}(n-1) \) \( \forall k \geq 1 \) and \( n \in \mathbb{Z} \).
(f) \( p_2(n) = \lfloor n/2 \rfloor \) (floor function).

Proof. 

(a) A sum of positive integers is not negative.

(b) A partition into \( k \) parts has sum \( \geq 1 + 1 + \ldots + 1 = k \) \( k \) times.

(c) The only partition into 0 parts is (\).

(d) The only partition of \( n \) into 1 parts is (\( n \)), which exists only if \( n \neq 0 \).

(e) Classify the partitions of \( n \) into \( k \) parts into 2 types:

Type 1: partitions whose last entry is 1.
Type 2: partitions whose last entry is \( \neq 1 \) (i.e., partitions whose all entries are \( \geq 2 \)).

Now, there is a bijection

\[
\{ \text{Type-1 partitions of } n \text{ into } k \text{ parts} \} \\
\rightarrow \{ \text{partitions of } n-1 \text{ into } k-1 \text{ parts} \},
\]

\[
(\lambda_1, \lambda_2, \ldots, \lambda_{k-1}, 1) \mapsto (\lambda_1, \lambda_2, \ldots, \lambda_{k-1}).
\]

Thus,

\[
\#(\text{Type-1 partitions}) = \binom{n-1}{k-1}.
\]

Also, there is a bijection

\[
\{ \text{Type-2 partitions of } n \text{ into } k \text{ parts} \} \\
\rightarrow \{ \text{partitions of } n-k \text{ into } k \text{ parts} \},
\]

\[
(\lambda_1, \lambda_2, \ldots, \lambda_k) \mapsto (\lambda_1-1, \lambda_2-1, \ldots, \lambda_{k-1}).
\]

Thus,

\[
\#(\text{Type-2 partitions}) = \binom{n-k}{k}.
\]

Adding these equalities together, we get the claim of (e).
(f) The partitions of $n$ into 2 parts are
\[(n-1, 1), (n-2, 2), (n-3, 3), \ldots, \left(\lceil \frac{n}{2} \rceil, 1, \frac{n}{2} \right)\]

By the way...

Prop. 4.13. Let $n \in \mathbb{N}$ and $k \in \mathbb{N}$. Then,

\[p_k(n) = \text{number of partitions of } n \text{ whose largest part is } k.\]

Proof outline. Picture proof: e.g., let $k=4$ and $n=14$.

Start with the partition $(5, 4, 4, 1)$ of $n$ into $k$ parts.

Draw a table of $k$ left-aligned rows, where the length of each row is the corresponding part of the partition.

(This table is called a Young diagram or Ferrers diagram.)
Now, flip this table around the \( \diagup \) diagonal.

The length of the rows of this new table form a partition of \( n \) whose largest part is \( k \).

This is a bijection. (It is called conjugation.)

Prop. 4.14, \( \# \text{ of } \mathcal{U} \rightarrow \mathcal{U} \text{ placements} = p_{1 \times 1}(1A1) \).

Proof outline: Encode a surjective \( \mathcal{U} \rightarrow \mathcal{U} \) placement as a partition of \( 1 \) into \( 1 \times 1 \) parts (where each part is the number of balls in some box). This is a bijection. \( \square \)
Prop. 4.15. (\# of \(U \to U\) placements)

\[
p_0(|A|) + p_1(|A|) + \ldots + p_{|X|}(|A|)
\]

Proof. Exercise.

Prop. 4.16. (\# of injective \(U \to U\) placements)

\[
|A| \leq |X|
\]

Proof. Exercise.
The twelvefold way:

<table>
<thead>
<tr>
<th>A → X</th>
<th>arbitrary</th>
<th>injective</th>
<th>surjective</th>
</tr>
</thead>
<tbody>
<tr>
<td>L → L</td>
<td>(</td>
<td>X</td>
<td>^{1A_1} )</td>
</tr>
<tr>
<td>U → L</td>
<td>( \frac{</td>
<td>A_1</td>
<td>+</td>
</tr>
<tr>
<td>L → U</td>
<td>( {</td>
<td>A_1</td>
<td>,</td>
</tr>
<tr>
<td>U → U</td>
<td>( p_0(</td>
<td>A_1</td>
<td>) + p_1(</td>
</tr>
</tbody>
</table>
Example: Given $n$ persons ($n > 0$) and $k$ tasks ($k > 0$).

(a) What is the number of ways to assign to each person a task such that each task has at least one person working on it?

(b) What if we additionally want to choose a leader for each task (among the people assigned to it)?

(c) What if, instead, we want to choose a vertical hierarchy (between the people working on the task) for each task?

Example: 8 people (1, 2, 3, 4, 5, 6, 7, 8) and 3 tasks.

(a)  
\[
\begin{align*}
&\text{task 1: } 1, 2, 5 \\
&\text{task 2: } 3 \\
&\text{task 3: } 4, 6, 7, 8
\end{align*}
\]

(b)  
\[
\begin{align*}
&\text{task 1: } 1, 2, 5 \\
&\text{task 2: } 3 \\
&\text{task 3: } 4, 6, 7, 8
\end{align*}
\]
Answers:

(a) \( \text{spr}(n, k) \),

(b) \( \binom{n(n-1) \ldots (n-k+1)}{k} \cdot k^{n-k} \),

(c) \( n! \cdot \binom{n-1}{k-1} \).

(See 4909 Fall 2017 Oct 10.)
4.8. A GLIMPSE OF RANKING & UNRANKING

Ranking & unranking is an important subject; we only will scratch the surface. See [Loehr, Ch. 5] and [Knuth, "The Art of Computer Programming", vol. 4A] for more.

Statement of the problem: You have a finite set $X$. Find ways to

1. construct a list of all elements of $X$ ("listing").
2. for each $k \in [1 \times 1]$, compute the $k$-th element of this list without having to write down the whole list ("unranking").
3. for each $x \in X$, compute the rank of $x$ in this list (i.e., the # of elements before $x$ in the list) ("ranking").

Equivalently: construct a bijection $[1 \times 1] \rightarrow X$ and its inverse, ("construct" = find an algorithm.)

Some questions apply to a countable set $X$, using $\mathbb{N} = \{1, 2, 3, \ldots\}$ instead of $[1 \times 1]$. 
Example 1: Let $N \in \mathbb{N}$, let $\mathcal{P}(X)$ (where $X$ is any set) denote the set of all subsets of $X$.

How to list $\mathcal{P}([N])$ (a $2^N$-element set)?

Example: A list for $\mathcal{P}([3])$:

$$\{
\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{2,3\}, \{1,3\}, \{1,2,3\}\}.$$

This is not very systematic.

Better: recursion.

A list of $\mathcal{P}([0])$: $\emptyset$.

To build a list of $\mathcal{P}([N+1])$ from a list of $\mathcal{P}([N])$, we first write down this list of $\mathcal{P}([N])$, and then write down a second copy of this list, but replacing each subset $S$ by $S \cup \{N+1\}$.

For example:

<table>
<thead>
<tr>
<th>$\mathcal{P}([0])$</th>
<th>$\emptyset$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{P}([1])$</td>
<td>$\emptyset, {1}$</td>
</tr>
<tr>
<td>$\mathcal{P}([2])$</td>
<td>$\emptyset, {1}, {2}, {1,2}$</td>
</tr>
<tr>
<td>$\mathcal{P}([3])$</td>
<td>$\emptyset, {1}, {2}, {3}, {1,2}, {1,3}, {2,3}, {1,2,3}$</td>
</tr>
</tbody>
</table>
Now, how do we rank/unrank in this list?

Observations:
- The list of $P([N+1])$ starts with the list of $P([N])$.
  Thus, the rank of a $S \in P([N])$ does not depend on $N$.
- If $S \in P([N])$ has rank $r$,
  then $S \cup \{N+1\} \in P([N+1])$ has rank $r + 2^N$.

$\Rightarrow$ (by induction): The rank of any $S \in P([N])$ is

$$\sum_{i=1}^{\infty} 2^{i-1} \quad \text{if} \quad S \in S$$

(E.g., the rank of $\{1,3\}$ is $2^{1-1} + 2^{3-1} = 1 + 4 = 5$.)

To unrank, we start with an $n \in \{0,1, \ldots, 2^N-1\}$, and we look for the subset $S$ of $[N]$ with $\sum_{i \in S} 2^{i-1} = n$.

This is just the set of positions of the 1-bits in the binary expansion of $n$.

Exercise: Ranking/unranking for labunar subsets leads to the Zeckendorf theorem.
Example 2: Given \( j \in \mathbb{N} \) and \( \mathbb{N} \) a set \( X \), we let 
\( \mathcal{P}_j(X) \) be the set of all \( j \)-element subsets of \( X \).

Given \( j \in \mathbb{N} \), how to list \( \mathcal{P}_j([N]) \) or \( \mathcal{P}_j(\mathbb{N}) \)?

**Example:** \( j=3 \); try to list \( \mathcal{P}_3(\mathbb{N}) \).

**Bad attempt:** \( \{0,1,2\}, \{0,1,3\}, \{0,1,4\}, \ldots \)

This never gets to \( \{0,2,3\} \).

**Better:** recursion. Let's first list \( \mathcal{P}_j(\{0,1,\ldots,N\}) \) for each \( N \geq -1 \).

Again, \( \mathcal{P}_0(\text{anything}) \) has \( \mathcal{P}_0(\emptyset) \).

Also, \( \mathcal{P}_j(\{0,1,\ldots,N\}) \) has list ( ) if \( j > 0 \).

To build a list of \( \mathcal{P}_j(\{0,1,\ldots,N+1\}) \) from a list of \( \mathcal{P}_j(\{0,1,\ldots,N\}) \) and a list of \( \mathcal{P}_{j-1}(\{0,1,\ldots,N\}) \), we first write down the list of \( \mathcal{P}_j(\{0,1,\ldots,N\}) \), and then write down the list of \( \mathcal{P}_{j-1}(\{0,1,\ldots,N\}) \), but replacing each subset \( S \) by \( S \cup \{N+1\} \).
For example:

<table>
<thead>
<tr>
<th>$P_0$ (void)</th>
<th>(void)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_0$ ({}0)</td>
<td>(void)</td>
</tr>
<tr>
<td>$P_1$ ({}0)</td>
<td>({}1)</td>
</tr>
<tr>
<td>$P_0$ ({}1)</td>
<td>(void)</td>
</tr>
<tr>
<td>$P_1$ ({}1)</td>
<td>({}0, {}1)</td>
</tr>
<tr>
<td>$P_2$ ({}1)</td>
<td>({}0, {}1)</td>
</tr>
<tr>
<td>$P_0$ ({}0,{}1)</td>
<td>(void)</td>
</tr>
<tr>
<td>$P_1$ ({}0,{}1)</td>
<td>({}0, {}, {}, {}1)</td>
</tr>
<tr>
<td>$P_2$ ({}0,{}1)</td>
<td>({}0, {}, {}, {}1)</td>
</tr>
<tr>
<td>$P_3$ ({}0,{}1)</td>
<td>({}0,{}1)</td>
</tr>
<tr>
<td>$P_4$ ({}0,{}1)</td>
<td>({}0,{}1)</td>
</tr>
</tbody>
</table>
Again, the list of \( P_j(\{0, 1, \ldots, N+1\}) \) starts with the list of \( P_j(\{0, 1, \ldots, N\}) \). Thus, the rank of an \( S \in P_j(\{0, 1, \ldots, N\}) \) does not depend on \( N \).

How to rank/unrank?

Observe:

- If \( S \in P_{j-1}(\{0, 1, \ldots, N\}) \) has rank \( r \), then \( S \cup \{N+1\} \) has rank \( \binom{N+1}{j} + r \).

\[ \implies \text{(by induction): The rank of any \( \{s_1 < s_2 < \ldots < s_j\} \in P_j(\{0, 1, \ldots, N\}) \text{ is } \binom{N}{1} + \binom{s_2}{2} + \cdots + \binom{s_j}{j}.} \]

This solves Exercise 7 on Midterm 1 again, as the list of \( P_j(\{N\}) \) is infinite in length when \( j > 0 \).

To unrank, you need to expand an \( n \in \mathbb{N} \) in the form

\[ n = \binom{s_1}{1} + \binom{s_2}{2} + \cdots + \binom{s_j}{j} \text{ with } 0 \leq s_1 < s_2 < \cdots < s_j. \]