Jewel Query Problem

April 25, 2014

1. **Problem** For a given set of slots (each slot can be of 1 hole, 2 holes or 3 holes), and a given set of jewels (each jewel occupies 1 hole, 2 holes or 3 holes, and are called size-1 jewel, size-2 jewel and size-3 jewel respectively), find the set of combination of jewels that fit into the set of slots. Each Jewels can be used multiple times.

2. **Model** Assign each jewel an unique id, and ensure that all size-3 jewel ids ¿ all size-2 jewel ids ¿ all size-1 jewel ids ¿= the dummy size-1 jewel (represent an empty embedding) ids.

   - Define C_1 as the set of all the 1-hole jewels. Define C_2, C_3 analogical to C_1.
   - Represent each jewel combination as an ordered collection of their ids. For example, $(1, 3, 3, 4)$ is a jewel combination of 4 jewels, and $\{(1, 1), (1, 2), (5)\}$ is a set of jewel combinations.
   - Define the operator $+$ over the **set of** set of jewel combinations: $A+B = A \cup B$, and $+$ is valid if and only if $A \cap B = \emptyset$.
   - Define the operator $\times$ over the **set of** set of jewel combinations:

     $$A \times B = \{(a_1, \cdots, a_m, b_1, \cdots, b_n) \mid (a_1, \cdots, a_m) \in A \text{ and } (b_1, \cdots, b_n) \in B \text{ and } a_m \leq b_1\}$$

     where $a_1, \cdots$ are all jewel ids.
   - Define $f(i, j, k)$ to be the set of all the jewel combinations that fits into i 1-hole, j 2-hole and k 3-hole slots. Define $f(i, \overline{j}, k)$ to be similar to $f(i, j, k)$, but the j 2-hole slots can only hold jewels that occupy exactly 2 holes (size-2 jewels). The same goes for i, k when there is a bar over each one of them.

     Example:

     (1) $f(i, j, \overline{k})$ = the set of all jewel combinations that fits into i 1-hole, j 2-hole and k 3-hole slots, where the k 3-hole slots can only be occupied by size-3 jewels.

     (2) $f(i, \overline{j}, \overline{k})$ = the set of all jewel combinations that fits into i 1-hole, j 2-hole and k 3-hole slots, where the k 3-hole slots can only be occupied by size-3 jewels and the j 2-hole slots can only be occupied by size-2 jewels.

3. **Analysis** Let us start from the simplest case.

- $f(i,0,0)$.
  It is easy to find that
  $$f(1,0,0) = C_1 \tag{1}$$
  and also, force 1-hole slot to take size-1 jewel does not make a difference:
  $$f(\bar{i},0,0) = f(i,0,0) \tag{2}$$
  Besides, to iteratively generate $f(i,0,0)$, we have
  $$f(i,0,0) = f(i-1,0,0) \times C_1 \quad (i > 1) \tag{3}$$
- $f(i,j,0)$.
  Now we introduce 2-hole slots. For $f(i,j,0)$, it's easy to find out that there are only two cases:
  (1) All the j 2-hole slots take size-2 jewels. This is exactly $f(i,\bar{j},0)$.
  (2) At least one of the j 2-hole slots is taking size-1 jewels. This is exactly $f(i+2,j-1,0)$.

  Note that the above two cases are disjoint. Which makes it easy to find the iterative formula for $f(i,j,0)$. Basically, when $j > 0$,
  $$\begin{aligned} f(i,j,0) &= f(i,\bar{j},0) + f(i+2,j-1,0) &(4)\\ &= f(i,0,0) \times f(0,\bar{j},0) + f(i+2,j-1,0) &(5)\end{aligned}$$
  This shows that we also need to maintain the state of $f(0,\bar{j},0)$ in the implementation, namely
  $$f(0,\bar{j},0) = f(0,\overline{j-1},0) \times C_2 \tag{6}$$
- $f(i,j,k)$.
  This is the most complicated cases, at least it seems to be. Though it can still be divided into two cases:
  (1) All the k 3-hole slots take size-3 jewels. This is exactly $f(i,j,\bar{k})$.
  (2) At least one of the k 3-hole slots is taking size-1 or size-2 jewels. This is actually $f(i+1,j+1,k-1)$. **Note that** we do not need to count $f(i+3,j,k-1)$ since $f(i+3,j,k-1) \in f(i+1,j+1,k-1)$!

  The bottom line is that when $k > 0$,
  $$\begin{aligned} f(i,j,k) &= f(i,j,\bar{k}) + f(i+1,j+1,k-1) &(7)\\ &= f(i,j,0) \times f(0,0,\bar{k}) + f(i+1,j+1,k-1) &(8)\end{aligned}$$
  This requires maintaining states for $f(0,0,\bar{k})$ via
  $$f(0,0,\bar{k}) = f(0,0,\overline{k-1}) \times C_3 \tag{9}$$

With all the equations, we are now finally able to calculate each $f(i,j,k)$ in a mnemonic lazy-way.