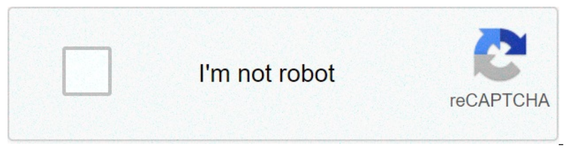

Continue

What kind of boat is pinwheel

What type of boat is the pinwheel. What boat is the pinwheel. What kind of boat is the pinwheel on wicked tuna.

If the application asks how many possible solutions there are, it can be resolved with generating functions. (This method was popular by George PÃƒÂ³la but I learned it from Knuth.) Leave that A (Z) is the generating function for the number of ways to spend n cents using only $ 1.22 Yoyo. So of course we can only spend those n who are multiple of 122 cents. So let's define a (z) = 1 + z ^ 122 + z ^ 244 + z ^ 366 + ... In this formulation the coefficient represents the number of ways: that's why the coefficient is 1 if the power is a multiple of 122 m but otherwise. Then we realize that this infinite sum can be transformed into A (Z) = 1 / (1 - Z ^ 122). Subsequently, we consider what happens if there is allowed to use both the $ 1.22 and $ 2.75 doll. We obtain B (Z) = 1 / (1 - Z ^ 275) * A (Z). We repeat this for all items, to get to the closed-shaped solution U (Z) = 1 / ((1 - Z ^ 122) * (1 - Z ^ 275) * (1 - Z ^ 185) * ... * (1 - Z ^ 87)) Now if we have a computer algebra system, we only ask you to perform a Taylor expansion and read the solution at the end z ^ 4394. Math may do it but unfortunately I can't get Wolfram Alpha to do it. The answer is 4794820. (This allows repetitions, and I think the author has not allowed repetitions. The author has also lost the third object, Duckie $ 1.85 which is evident from the posted code and the fact that there are references to 2 ^ 20 not 2 ^ 21.) If you do not have a computer algebra system you can obviously solve it manually with dynamic programming: the z ^ n coefficient for b (z) is the same as the coefficient of Z ^ N for A (Z) the coefficient of Z ^ (N-275) for B (Z). Think about thinking about the distribution law for multiplication. But then for hacker news, I imagine that dynamic programming is more favorite so: #include #include int main () {static costexpr std :: array kitempcces = {122, 275 , 185, 597, 647, 216, 713, 457, 146, 518, 316, 489, 711, 645, 477, 804, 671, 231, 621, 98, 231, 621, 98, 87}; Static Std :: Array dp = {}; // Base case for (int j = 0; j