Partition Problem

Rishi Dua, 2010EE50557

9 February, 2014

1 PARTITION PROBLEM

1.1 PROBLEM STATEMENT

There are N packets, each with one or more candies. There are K students among which the packets have to be distributed. (Assume K less N for all cases). The parameters N and K have to be provided by the user at run-time. Each student gets only one packet. The number of candies in various packets are (x1, x2, x3,....xk) , where xi denotes the number of candies in the ith packet. Find the number of triplets (x1, x2, x3) possible such that sum of the candies (x1 + x2 + x3) is even.

Divide the packets into two parts (p1 and p2) such that the difference (|p1-p2|) is minimum, where p1 and p2 are the total number of candies in part 1 and part 2 respectively.

1.2 ABSTRACT

This problem was originally designed for Dynamic Programming. However, it has been solved as a deterministic problem as it is trivial.

1.3 SPECIFICATION AND ASSUMPTIONS

Tool Specifications:
Language used: Java
Platform: Ubuntu 12.04
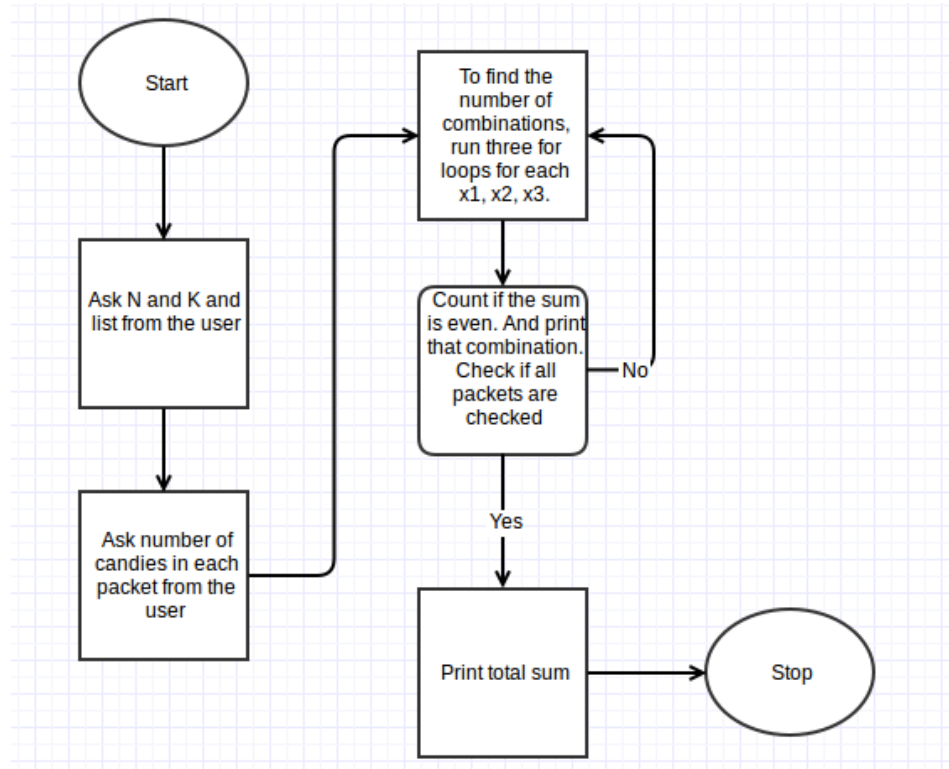Additional tools used: none
Eclipse Version: Version: 3.7.2

Assumptions
Input is taken one at a time
The list of x1 to xn is seperated by commas
Prints each combination of triplet whose sum is even on terminal

1.4 Flow chart

1.5 Logic Implementation

Following are the two main steps to solve this problem:

1. Calculate sum of the array. If sum is odd, there can not be two subsets with equal sum, so return false.

2. If sum of array elements is even, calculate sum/2 and find a subset of array with sum equal to sum/2.

The first step is simple. The second step is crucial, it can be solved either using recursion or Dynamic Programming.

1.6 Execution Directive

Compiling:
javac dp.java

Running:
java dp

1.7 Output Of The Program

```
Part A
Enter N
10
Enter K
4
Enter x0
10
Enter x1
20
Enter x2
30
Enter x3
3
Enter x4
3
Enter x5
3
Enter x6
3
Enter x7
3
Enter x8
3
Enter x9
2
The total number of possible triplets such that the sum is even is 64

Part B
The students get packets containing the following number of candies:
0
2 1
3 2
```

```
3 3
3 4
3 5
3 6
3 7
10 8
20 9
30

Part C
Partition problem
true
5 = 2+3
5 = 2+3
5 = 2+3
5 = 2+3
5 = 2+3
5 = 2+3
```

1.8 RESULT

Solved the trivial partition problem.

- Recognized when dynamic programming is a plausible approach. E.g., recursive formulation, repeated subproblems, Global opt depends on opt subsolution, but not details thereof.