Assignment 4

Question 1

- First we want to sort the input, we get: 2,6,7,11,16,17,22,23,33,44,54,56,66,107
- Then we want to solve using the formula given in lecture: $p(a_i, a_j) = 2/(j-i)$
- We can map each integers (17, 44) to numbers in the following way: i=5+1=6, j = 9
- Using the formula from class: $p(a_i, a_j) = 2/5$

Question 2

- From the question we can produce an approximation by using cutting patterns of uniform widths.
- Once the algorithm has been run, any cutting pattern that is less than half full we can combine with another less than half full to produce cutting patterns that are at least half full. Except for one in some occasions. This produces at most k/2 cutting patterns of mixed widths.
- We know that all rows are at least half full (except possibly one) because any roll that is less than half full can be continually combine with other rolls that are less than half full until that roll is at least half full.
- Next we want to look at an optimal solution the original problem, since we are looking for a lower bound on the approximation we will choose and optimal solution where each of the raw rolls are completely full with cutting patterns.
- Looking at this solution we can determine that a solution using twice the number of raw rolls can be satisfied if each roll is half full with cutting patterns.
- If we look at our solution using the approximation algorithm we see that since each final roll is at least full we have found a two approximation.

Question 3

- Given the solution to the fractional relaxation of the optimal solution we can calculate the number of rolls necessary by summing all of the x_is.
- We are trying to find a solution within ± k/2 of the fractional solution that is an integer solution.
- To find this we look at each of the individual x_is, and do one of the following options
- let $l_i = x_i - floor(x_i)$
  - if l_i > 1/2 then: we want to simply round up the x_is, this gives at most k/2 since — $sum_k(l_i) < k/2, if: l_i > (1/2)$
  - if l_i < 1/2 then: we do as we did in question three, we want to continuously join the l_is until we have a roll that is full greater than a half, and from the previous question we know that this can not sum to greater than k/2 cutting patterns of mixed width

Question 4

- The solution is i=j and x = w-w_j
- This is the solution because in the algorithm we want to set a cell to true if it can be chosen to fill the backpack, and we want to adapt the algorithm for the normal knapsack to a multi knapsack.
- To achieve this we must look to see in the current column, if the previous multiple is set to true for the current weight we are looking at. If it is set to true, then we can set the new multiple of the current weight to true and then we will set the value appropriately.