Dane:

- **Wektor x**, który składa się z ułamków sumujących się do 1. Jego długość sobie losujemy, powiedzmy od 3 do 10 lub 15 (jak chcemy, ale to determinuje pozostałe rozmiary). Długość tego wektora oznaczmy przez **N**.
  przykład: x = [0.5, 0.5], czyli N=2.
- **R** czyli tablica wypłat dla wektora **x**. Jest to macierz NxN wypełniona losowymi liczbami naturalnymi.
  Przykład:

4	2
7	10

- **C** czyli tablica wypłat dla wektora **y** (o nim za chwilkę). Jest to macierz NxN wypełniona losowymi liczbami naturalnymi.
  Przykład:

1	2
0	1

Przebieg algorytmu:

Dla nas będzie on 2 etapowy.

Najpierw na podstawie wektora **x** i tablicy **C** będziemy szukać wektora **y**. Wektor **y** jest również rozmiaru N, ale składa się z jednej '1' i reszty '0'.

Etap I:

Wykonujemy sumowanie (wzór 23) ze skryptu, czyli liczymy $\max_{q=[q1,\ q2]}$ = 0.5 * q1 * 1 + 0.5*q1*2 + 0.5*q2*0 + 0.5*q2*1 = 3/2 * q1 + 1/2 * q2. Ok, mamy równanie. Teraz musimy za q1 i q2 tak podstawić 0 i 1, żeby wartość była maksymalna (pamiętając, że tylko za jedną zmienną możemy podstawić '1'). Czyli wiadomo, że gdy za q1 podstawimy '1', wartość będzie największa (3/2). Skoro tak to wektor **y=[1, 0]**. Etap I skończony.

Teraz na podstawie tablicy **R** i wektora **y** będziemy szukać nowego wektora **x** (nowy zastępuje stary).

Etap II (ciut gorszy):

Tutaj będziemy wykorzystywać (wzór 25) ze skryptu. Wygląda to tak:

$\text{Max}_{x=[x1,\ x2]}$ = 1 * 4 * x1 + 1 * 2 * x2 + 0 * x1 * 7 + 0 * x2 * 10 = 4*x1 + 2*x2. Teraz musimy znaleźć nowe wartości x1 i x2, tak, aby wielomian ten miał jak największą wartość. UWAGA, one muszą być **ułamkami** sumującymi się do 1. To właśnie ten problem mamy rozwiązać. Możemy zastosować metodę Simplex lub inną jeżeli chcemy.