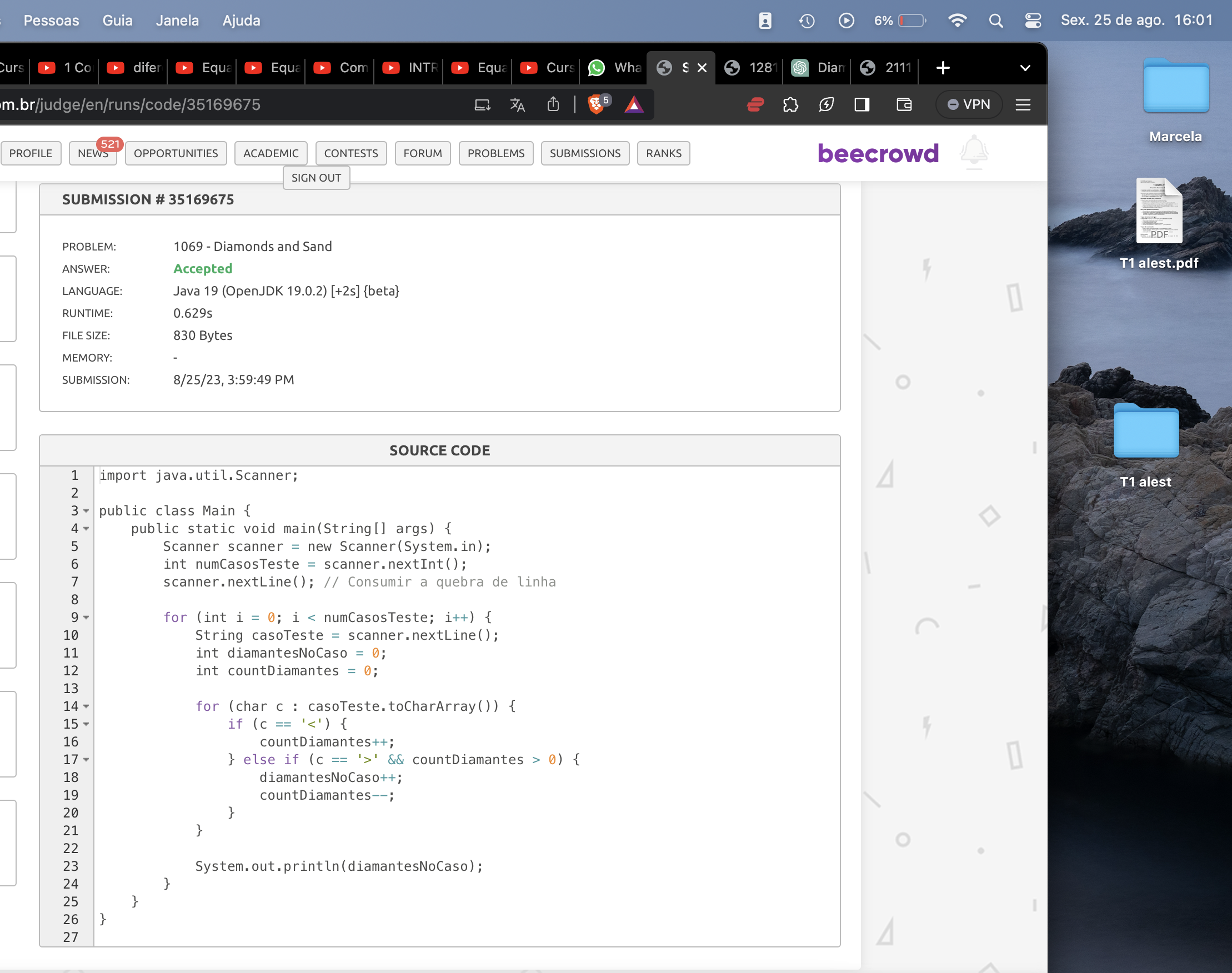
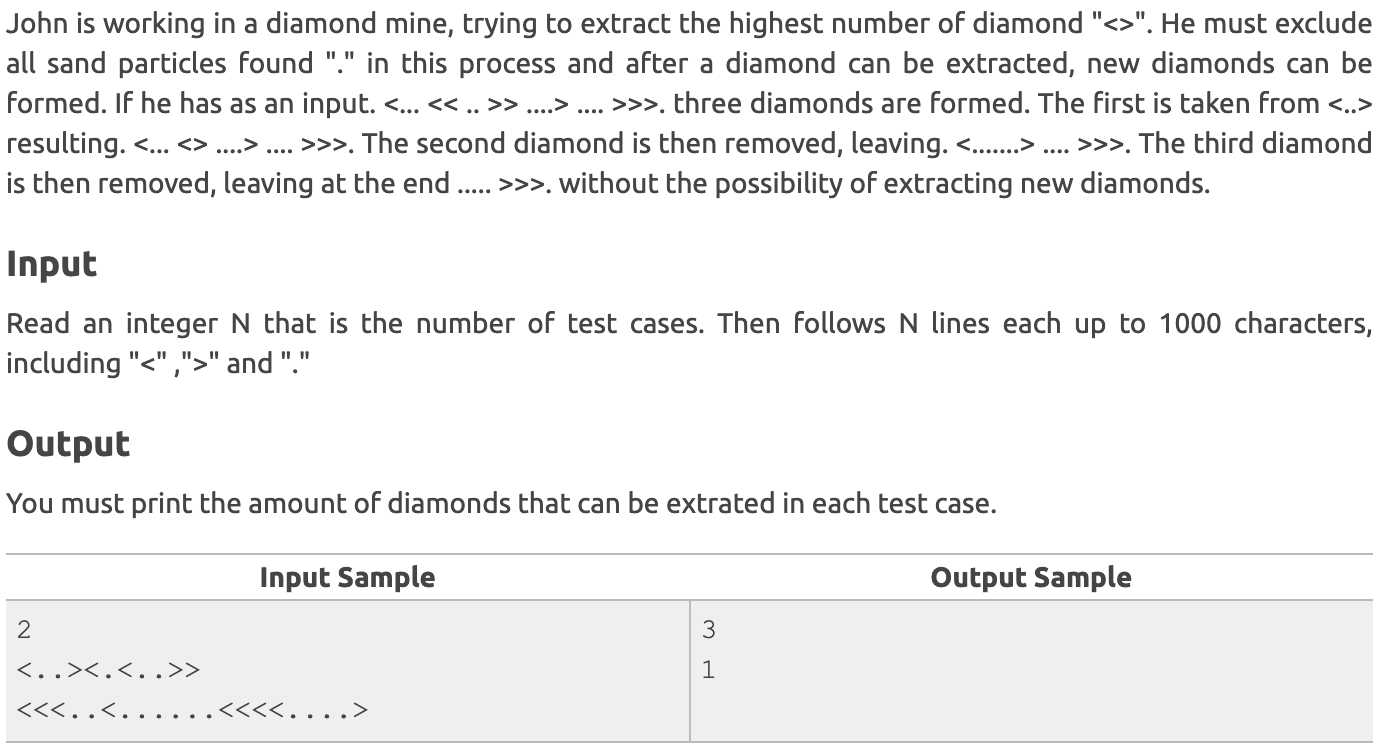
Relatório Trabalho 1 (Algoritmos e Estruturas de Dados)

Aluna: Camila Rizzotto

**Problema 1:**

**Título**: Diamonds and Sand

**Nível**: 3

 **URL**: <https://www.beecrowd.com.br/judge/en/problems/view/1069>

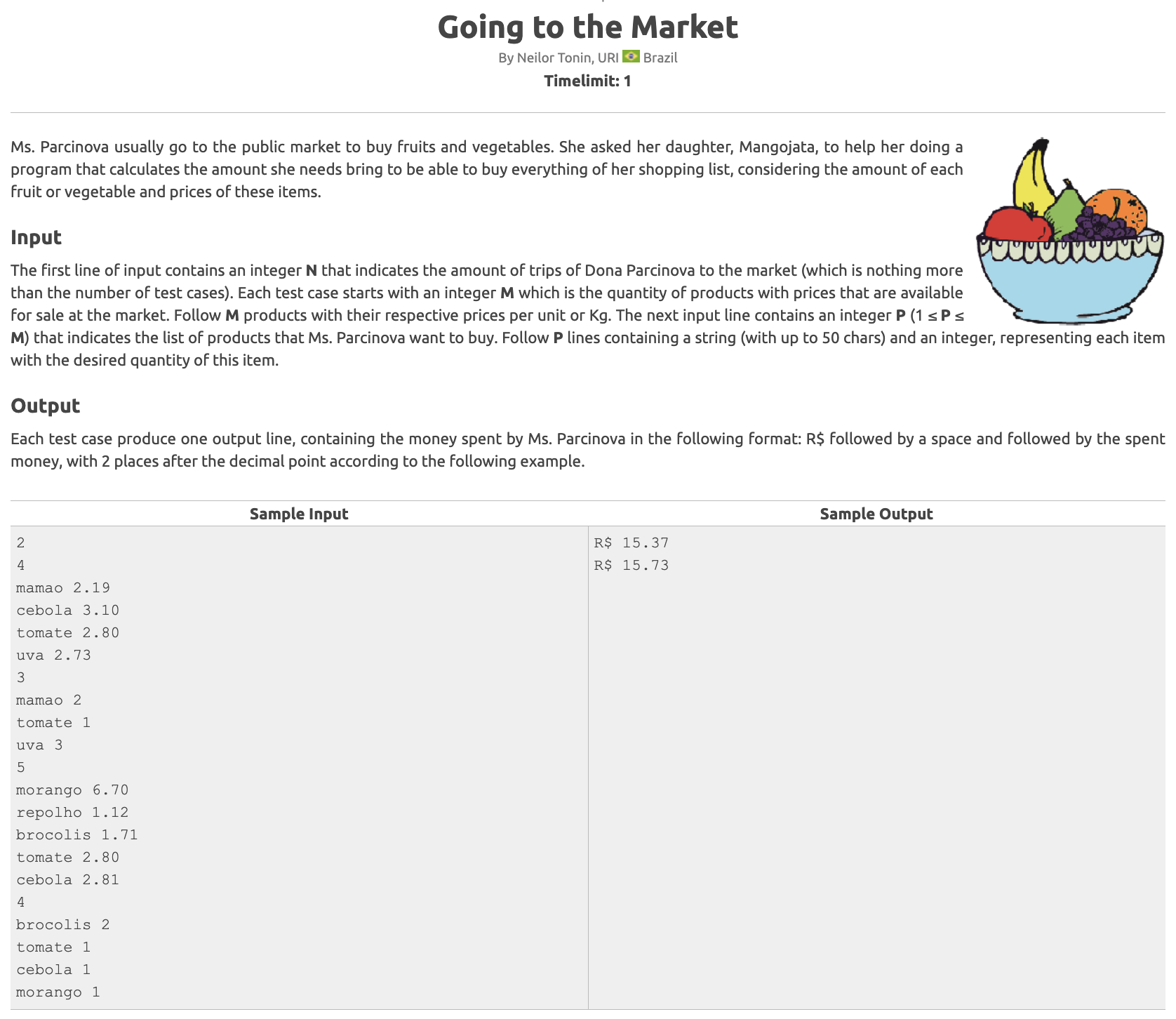
**Código fonte**: em anexo em Ex1069.java

**Complexidade** **do algoritmo**: O(n2)

O algoritmo desenvolvido tem a complexidade apontada ja que para reconhecimento e “separação" dos caracteres pesquisados foi utilizada uma abordagem com dois laços de for percorrendo todo array de caracteres em busca da identificação da sequencia desejada que correspondia ao formado de diamante conforme solicitado no enunciado. Vale lembrar o “pior dos casos” ocorre praticamente sempre ja que o array inteiro sempre será percorrido dado que nao sabemos quantos diamantes podem estar separados por areia e se o numero de carácteres < equivale ao numero de caracteres > podendo ou não formar um numero exato de diamantes. Conforme os dois laços de for aninhados a complexidade se expande ja que o for externo itera sobre todos os casos de teste e o loop interno sobre todos os caracteres de cada caso de teste. Dado que o laço interno de for pode ter um tamanho que depende diretamente da entrada n poderíamos considerar um caso médio(m) e representar assintoticamente como complexidade O(n\*m), onde n é o numero de entradas e m o numero medio de casos a serem analisados a partir do tamanho da entrada.

**Problema 2:**

**Título**: Going to the market

 **Nível**: 3

**URL**: https://www.beecrowd.com.br/judge/en/problems/view/1281



**Código fonte**: em anexo em Ex1281.java

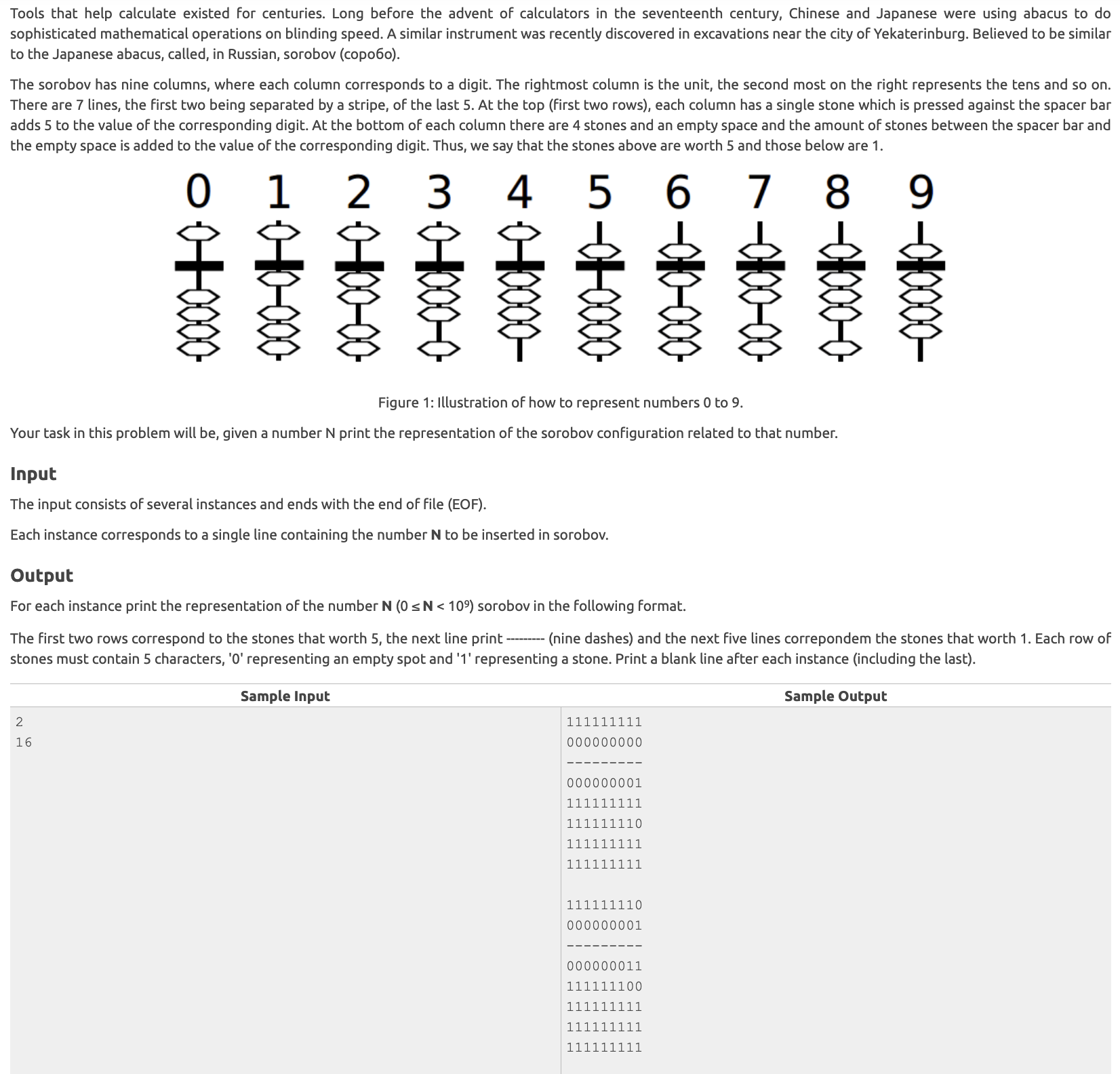
**Complexidade** **do algoritmo**: O(n\*p\*i)

O algoritmo desenvolvido tem complexidade assindética apontando para os laços principais do código. O primeiro laço de iteração tem um pior caso de n incidências e renova o numero de idas ao supermercado podendo conter diversos produtos. O Segundo laço de for itera sobre o numero de produtos comprados que pode ser variado tendo um total de P produtos a serem listados. Finalmente o terceiro laço de for tem uma complexidade I pois depende do numero total de itens que foram comprados. O crescimento total do algoritmo vai depender do numero de incidências de entrada que o usuário inserir.

**Problema 3:**

**Título**: Understanding Sorobov

**Nível**: 4

 **URL**: https://www.beecrowd.com.br/judge/en/problems/view/2111

**Código fonte**: em anexo em Ex2111.java

**Complexidade** **do algoritmo**: