

# Trabalho Prático 2 - O Saco do Papai Noel

A região com menor densidade populacional da Europa é a Lapônia, na Finlândia<sup>1</sup>. Dentro da Lapônia, o município com menor densidade populacional é Savukoski<sup>2</sup>. Nesse local, que é o menos povoado da Europa, nas redondezas do monte Korvatunturi, reside um ilustre habitante: **Papai Noel**.

Alguns raríssimos relatos dos habitantes locais explicam o motivo pelo qual o Papai Noel só é visto no resto do globo durante a época de Natal. A grande verdade é que ele não trabalha só no Natal! Papai Noel começa trabalhando muito antes, pois ele tem que decidir como levar os melhores presentes para as crianças. Papai Noel, na verdade, nem é muito velho. Mas ganhou cabelos e barbas brancas devido ao estresse diário que enfrenta para decidir quais presentes ele deve colocar no seu saco. Seu trabalho é usar o conhecimento adquirido em Algoritmos e Estrutura de Dados II para facilitar a vida do bom velhinho.

## 1 O Problema

Vamos explicar o problema do Papai Noel. Todo ano, Papai Noel recebe doações de presentes para que ele possa distribuir às crianças mais necessitadas. Baseado em sua experiência, Papai Noel sabe o **valor sentimental** que cada presente tem. Obviamente, Papai Noel quer levar os melhores presentes em seu saco, ou seja, os presentes com maior valor sentimental. Mas nem todos os presentes cabem no saco do Papai Noel. Os presentes têm **pesos** diferentes, mas o saco do Papai Noel tem um peso máximo permitido. Como Papai Noel deve escolher os presentes de forma que **não ultrapassem o peso máximo** e, ao mesmo tempo, **maximizem a soma total do valor sentimental**?

Sejam

- $n$ : número de presentes disponíveis para escolher (aqueles recebidos por doação);
- $p_{\max}$ : capacidade máxima do saco do Papai Noel;
- $v_i$ : valor sentimental do presente  $i$  ( $i = 1, \dots, n$ );
- $p_i$ : peso do presente  $i$  ( $i = 1, \dots, n$ ).

## 2 Formato da Entrada

Todos os números da entrada são naturais. O valores  $n$  e  $p_{\max}$  são os primeiros valores a serem lidos. Depois, há outros  $2n$  valores a serem lidos, na seguinte ordem:

```
n
p_max
v_1 p_1
v_2 p_2
.
.
.
v_n p_n
```

Por exemplo,

---

<sup>1</sup><https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20170406-1>

<sup>2</sup><https://en.wikipedia.org/wiki/Savukoski>

4  
5  
4 2  
5 3  
3 1  
7 4

### 3 Objetivo

Usando a técnica de **backtracking**, implemente um programa usando a linguagem de programação C, que percorra *cada um dos subconjuntos do conjunto dos  $n$  presentes*, verificando para cada subconjunto se **(1)** o peso dos presentes do subconjunto não excede o peso máximo do saco do Papai Noel e **(2)** se a soma do valor sentimental dos presentes no subconjunto é o maior até então. Ao final, deve ser impresso:

- os índices dos presentes que Papai Noel irá colocar em seu saco para maximizar a soma do valor sentimental
- a soma do valor sentimental dos presentes escolhidos

No exemplo de entrada apresentado anteriormente, a saída deve ser:

3 4  
10

A explicação dessa saída é que a melhor combinação é pegar o 3º item (valor 3, peso 1) e o 4º item (valor 7, peso 4), pois não excedem o peso máximo (que é 5), dando um valor sentimental total de 10 (3+7).

### 4 Arquivos a Serem Entregues

Dois arquivos:

- Arquivo `main.c` com o código-fonte de seu programa;
- Arquivo `Makefile` para que o professor possa compilar o trabalho usando o comando `make`;
- Arquivo `.txt` contendo o nome e o GRR dos alunos responsáveis pelo trabalho.

### 5 Entrega

O trabalho pode ser feito em duplas. Os arquivos devem ser empacotados em um arquivo `grr1_grr2.tar.gz`, onde `grr1` e `grr2` são os grrs dos alunos. Ao descompactar este arquivo deverá ser criado um diretório de nome `grr1_grr2` que conterá todos os demais arquivos.

O trabalho deve ser entregue via Moodle. A data limite para o envio está estipulada no link de entrega do Moodle. Não serão aceitas entregas em atraso, exceto para os casos explicitamente amparados pelas resoluções da UFPR.

### 6 Distribuição da Nota

O trabalho tem peso de 15% no semestre. Alguns descontos padrão, considerando uma nota entre 0 e 100 pontos para o trabalho:

- Plágio: perda total da pontuação para todos os envolvidos. Isso é válido mesmo para casos onde o plágio se refere a apenas um trecho do código;
- Algoritmos que não usam backtracking, ou implementados em linguagem diferente de C, serão desconsiderados;

- Programa não segue o formato da entrada exatamente como o requisitado: 10 a 50 pontos;
- Falta de algum arquivo requisitado: desconto de 10 a 100 pontos;
- Inclusão de arquivos desnecessários (lixo): desconto de 5 a 20 pontos;
- Erros e avisos de compilação: desconto de 5 a 100 pontos;
- Nomes de arquivo incorretos: 5 pontos por arquivo;
- Arquivo com formato incorreto: 5 pontos por arquivo.

## **7 Demais Regras**

- Dúvidas ou casos não especificados neste documento podem ser discutidos com o professor até a data de entrega do trabalho.
- Os trabalhos não serão aceitos após a data/hora limite.