Branch and bound algorithm for Knapsack

Introduction

We will focus on the optimisation problem of the knapsack, recalled below. We will implement in OCaml several algorithms to solve this problem :

— an exhaustive enumeration algorithm (section 2),

— a dynamic programming algorithm (section 3),

— a Branch-and-Bound algorithm (section 7).

Knapsack

The Knapsack problem is defined by :

Input : An integer $n \in \mathbb{N}$, two finite sequences $(p_i)_{i\in[1,n]} \in \mathbb{N}^n$ and $(v_i)_{i\in[1,n]} \in \mathbb{N}^n$, an integer $P \in \mathbb{N}$ and a threshold $K \in \mathbb{N}$.

Output : Exists an $I \subseteq [1, n]$ such that $\sum_{i\in I} p_i \leq P$ et $\sum_{i\in I} v_i \geq K$?