# Лабораторна робота №1

## Вирішення задачі 0-1 Knapsack problem за допомогою алгоритма Hill Climbing

Умова задачі. Для заданої множини предметів *N*, кожен з яких має вагу *wi* і цінність *c*i, визначити яку кількість кожного з предметів слід взяти, так, щоб сумарна вага не перевищувала задану *Cap*, а сумарна цінність *Profit* була максимальною:

Вирішенням задачі є бінарний вектор заповнення X = { x1, x2, …, xn }, та значення Profit, якого вдалось досягти.

Визначаємо цільову функцію для алгоритму HillClimbing.

Якщо P(X) – це вартість предметів для певного набору X

То цільова функція F(X) має вигляд:

N-Околом для певного бінарного вектора X є множина векторів {Y0, Y1, …, Yn}, у яких 0 - N бітів буде відрізнятись. (N-flip окіл)

Реалізація алгоритма мовою С++ знаходиться за посиланням <https://github.com/ospodinets/knapsackhc>

Був обраний варіант Variable Depth Search. Початкова відстань половина від розміру вхідних даних, кількість спроб: 100000, відстань зростає/спадає збільшенням/зменьшенням вдвічі, відповідно.

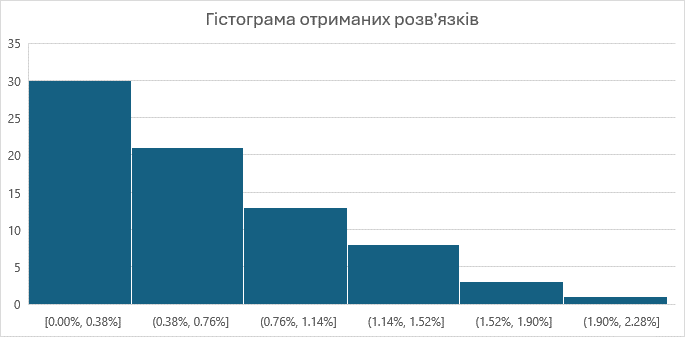
Вхідні дані та еталонний розв’язок згенеровано за допомогою відповідних утіліт, розроблених David Pisinger: http://hjemmesider.diku.dk/~pisinger/codes.html

Серед різних варіантів тестових даних було обрано набір зі 100 елементів максимальною вартістю і вагою 100 одиниць.

Було проведено 100 випробувань, дані випробувань викладено у наступних графіках:

Еталонний результат, 2671, він же найкращій, отриманий алгоритмом.

Найгірший результат – 2617, на 2% відхиляється від еталонного



Отримані розв’язки розподілені таким чином.

Графік значень цільової функції на прикладі чотирьох ітерацій