ReadMe file from the “A 3-level ILSCSP Applied to a Factory” repository

This repository contains the instances used in the paper entitled “A 3-level Integrated Lot Sizing and Cutting Stock Problem Applied to a Truck Suspension Factory”, published by “Flexible Services and Manufacturing Journal”.

Link to access:

https://github.com/prochavetz/A-3-level-ILSCSP-Applied-to-a-Factory

There are 181 “.dat” files with all the instances that were solved in the paper.

These instances are divided into 18 branches named “G1”, “G2”, ..., “G18”, each containing the 10 instances with random data of the respective group.

The branch called “main” contains, in addition to this ReadMe file, the instance with real data from the spring company.

The interpretation of the information contained in each instance is made below:

“nK” represents the number of bar types (from 1 to nK);

“nI” represents the number of spring types (from 1 to nI);

“nT” represents the number of periods (from 0 to nT);

“nP” represents the number of types of spring bundles (from 1 to nP);

“l” represents the length of each type of spring;

“eminI” represents the minimum stock for each type of spring;

“emaxI” represents the maximum stock of each type of spring;

“r0” represents the initial stock of each type of spring;

“pi\_IT1” is a vector used to assist in filling the matrix “pi\_IT”, of dual variables referring to constraint (5);

“lAux” is a vector used to assist in filling a vector with the length of the springs for each subproblem;

“cI” represents the inventory cost for each type of spring;

“L” represents the length of each type of bar;

“s0” represents the initial stock of each type of bar;

“eminB” represents the minimum stock for each type of bar;

“emaxB” represents the maximum stock of each type of bar;

“aux” is a vector used to assist in filling a vector with information about which bar is used in each generated cutting pattern;

“cB” represents the inventory cost for each type of bar;

“eminP” represents the minimum stock for each type of spring bundle;

“emaxP” represents the maximum stock of each type of spring bundle;

“p0” represents the initial stock of each type of spring bundle;

“cP” represents the inventory cost for each type of spring bundle;

“dr” represents the demand for each type of spring in each period;

“pi\_IT” is a matrix that stores the value of the dual variables of the constraint (5), for each type of spring in each period;

“ds” represents the demand for each type of bar in each period;

“AuxKT” is a matrix used to assist in filling the vector “pi\_KT”, of dual variables referring to constraint (2);

“pi\_KT” is a matrix that stores the value of the dual variables of the constraint (2), for each type of bar in each period;

“dp” represents the demand for each type of spring bundle in each period;

“CAP” represents the production capacity of springs in each period;

“Emax” represents the bar purchase limit in each period;

“pi\_T” is a vector that stores the value of the dual variables of the constraint (6), for each period;

“p” is a matrix that contains information about what types of bars can produce each type of springs;

“z” is a matrix that stores information on how many units of each type of spring are used in the assembly of each type of spring bundle;

“Patterns” is a structure created to store information about each cutting pattern added to the master problem. The information is arranged in the structure according to the following model:

< cutting pattern identification ; cost ; used bar ; produced springs >;

This structure is initialized with the homogeneous maximal cutting patterns of each instance.

“PadAux” is a structure created to store information about each generated cutting pattern. The information is arranged in the structure according to the following model:

< cutting pattern identification ; relative cost ; cost ; used bar ; produced springs >;