ReadMe file from the AMMOD-D-20-01547 repository

This repository contains the instances used in the paper entitled “The Integrated Lot Sizing and Cutting Stock Problem in an Automotive Spring Factory”, published by the journal “Applied Mathematical Modeling”.

Link to access:

https://github.com/prochavetz/AMMOD-D-20-01547

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 “master” 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 object types (from 1 to nK);

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

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

“nF” represents the number of cutting machines (from 1 to nF);

“nP” represents the number of types of final products (from 1 to nP);

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

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

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

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

“pi\_IT1” is a vector used to assist in filling the matrix “pi\_IT”, of dual variables referring to constraint (2);

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

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

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

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

“eminP” represents the minimum stock for each type of final product;

“emaxP” represents the maximum stock of each type of final product;

“p0” represents the initial stock of each type of final product;

“cP” represents the inventory cost for each type of final product;

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

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

“AuxKT” is a matrix used to assist in filling the matrix “pi\_FT”, of dual variables referring to constraint (4);

“dp” represents the demand for each type of final product in each period;

“limite” represents the maximum number of item types per cutting pattern that each machine is capable of producing;

“aux2” is a vector used to assist in filling the vector with information about which machine is used in each generated cutting pattern;

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

“pi\_FT” is a matrix that stores the value of the dual variables of the constraint (4), for each machine in each period;

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

“pm” is a matrix that contains information about what types of items can be produced on each machine;

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

“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 machine ; used object ; produced items >;

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 machine ; used object ; produced items >;