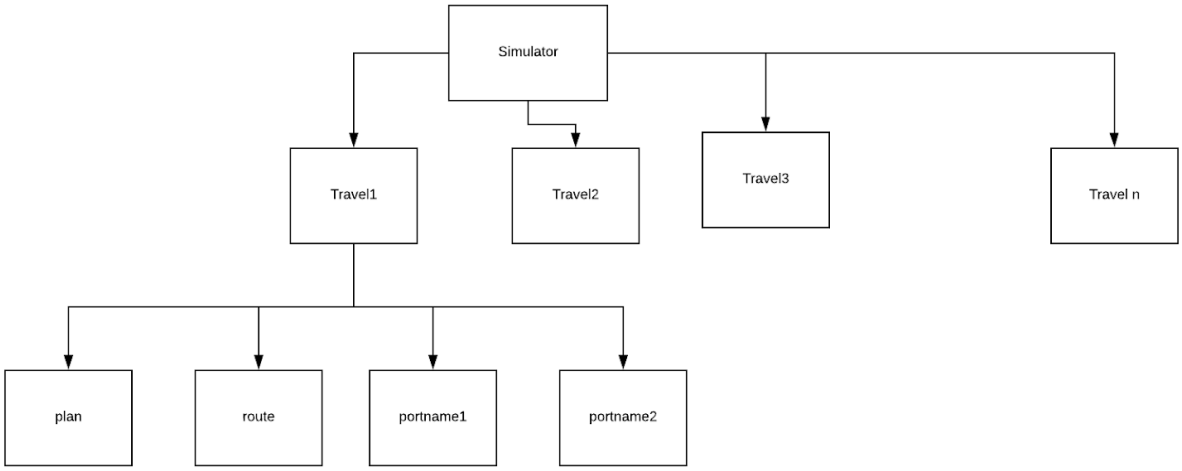
read me:

Algorithm implementation:

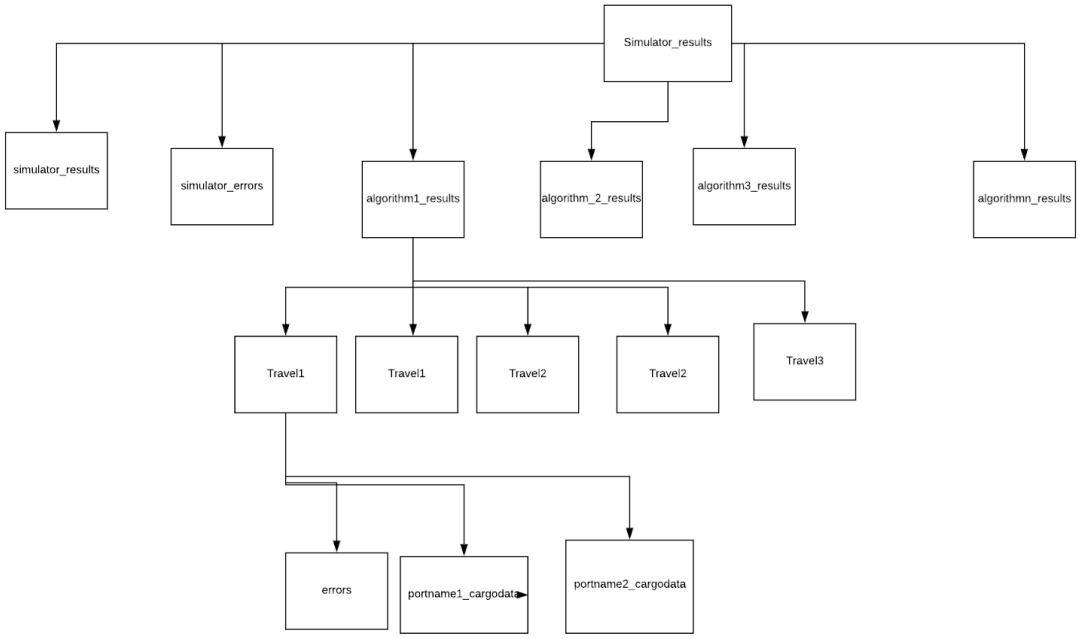
when the ship reach a port the algorithm get a list of containers to insert, the first thing that the algorithm do is unloading all the containers that their destination is the current port if there are containers above the wanted container we need to unload them also, after we unload them we add the unwanted containers to the list of containers that the algorithm get from the port, then the algorithm sort the list by their destination distance.
if ship is full or the ship will not reach container destination then the algorithm reject.

input/output format:
input:



input folder structure:
main folder(you are free to choose the name of the folder), in this folder you have to add subfolders of travels, in every subfolder you must include plan file, route file and port name file per every port in the route (note that you have to give port file for the last port also)
plan Format:
the first line include the dimensions of the ship separated by “,” h,x,y and you are free to insert spaces everywhere
route format:
every line include one port code (5 english letters) and you are free to insert spaces in every line
portname1 include data from the port to ship, container per line
<container_id>,<container_weight>,<container_destination>

Output format:



Output format structure:
the name of the main folder is simulator_results, in this folder you can find all the algorithms sumlation results plus sumulator_results simulator_errors files explained below
simulator_results file:
in this file you can find stpes number of running algorithm i in travel j, the format is:
Algorithm i Results on Travelj :
x steps (note that reject operations are not counted)
simulator_errors file:
this file include all the errors that prevents running the algorithm for example:
1.ship dimensions error
2.port/plan/route doesn't exist

algorithmi_results folder:
this folder include the simulation result for every travel as it described in the homework
errors:
this file include all the non critical errors for example:
invalid port/container id/weight format

this errors should not stop the algorithm run

How to Run our Code:
go to main.cpp change inputpath to simulator path (in our input path it
should be directory\simulator\) then change the outputpath to the path that you want to get the output folder in

the only printed message in the screen is when you didn't insert the right path for the simulation data

NOTE:If simulator_results folder found in the outputpath the simulation will not run because it create folder with same name.