


Algorithmics	Student information	Date	Number of session
	UO: UO294786	25/04/24	10
	Surname: Álvarez Iglesias	 Escuela de Ingeniería Informática Universidad de Oviedo	
	Name: Rafael		



Activity 1. [Numerical square]

YOU ARE REQUESTED TO: Design and implement an algorithm using the Branch and Bound technique to solve this problem optimally. The objective is to find a heuristic that, as far as possible, improves the times obtained with backtracking.

Implement said algorithm in Java (NumericSquareBaB.java) in such a way that it calculates a solution for a given board entry in the most efficient way possible.

Explain what heuristic has been used to quantify the quality of each node (that is, to quantify how close to a solution that node might be).

The heuristic chosen to make this algorithm is the one calculated by counting the number of empty squares.

Fill in the following table:

Test case	Time for first backtracking solution (ms)	Number of developed backtracking nodes	Time for first BaB solution	Number of developed BaB nodes
Test00	LoR	3	LoR	3
Test01	LoR	316	LoR	8
Test02	LoR	106	LoR	7
Test03	LoR	9608	438	12
Test04	LoR	80338	OoT	13
Test05	LoR	961	87	14
Test06	LoR	264	75	15
Test07	49	243079	OoT	-

What algorithm has worked in a better way? Why do you think this has happened?

Backtracking has proven to work better, since all the execution times are noticeably smaller. That could be because a lot of states must be created until the solution is found

Algorithmics	Student information	Date	Number of session
	UO: UO294786	25/04/24	10
	Surname: Álvarez Iglesias		
	Name: Rafael		

for the case of branch and bound, while on backtracking, the invalid solution cut much faster.