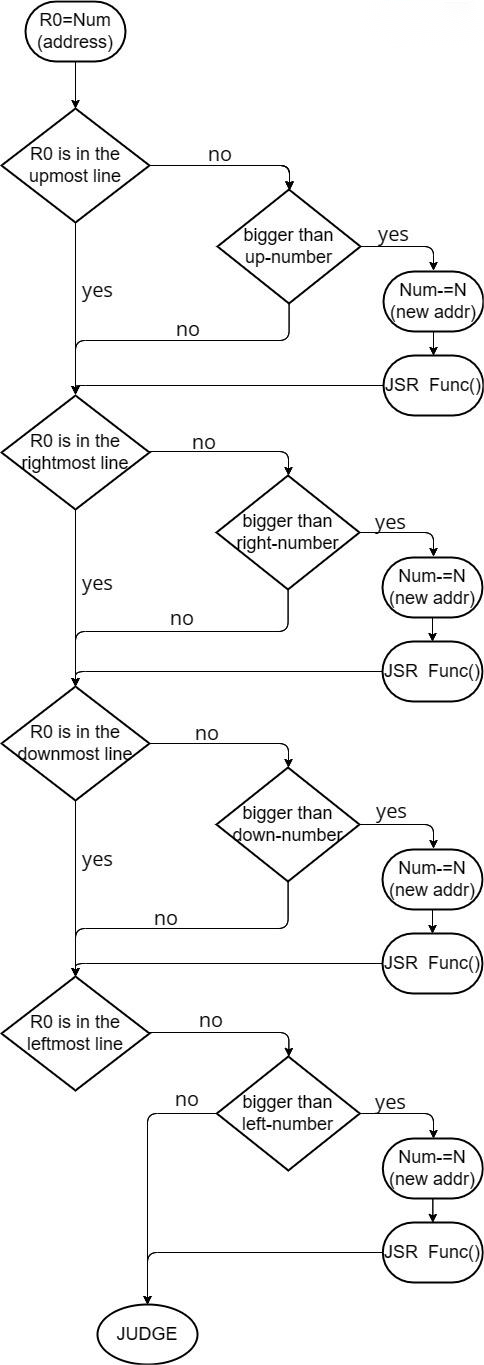
Lab5

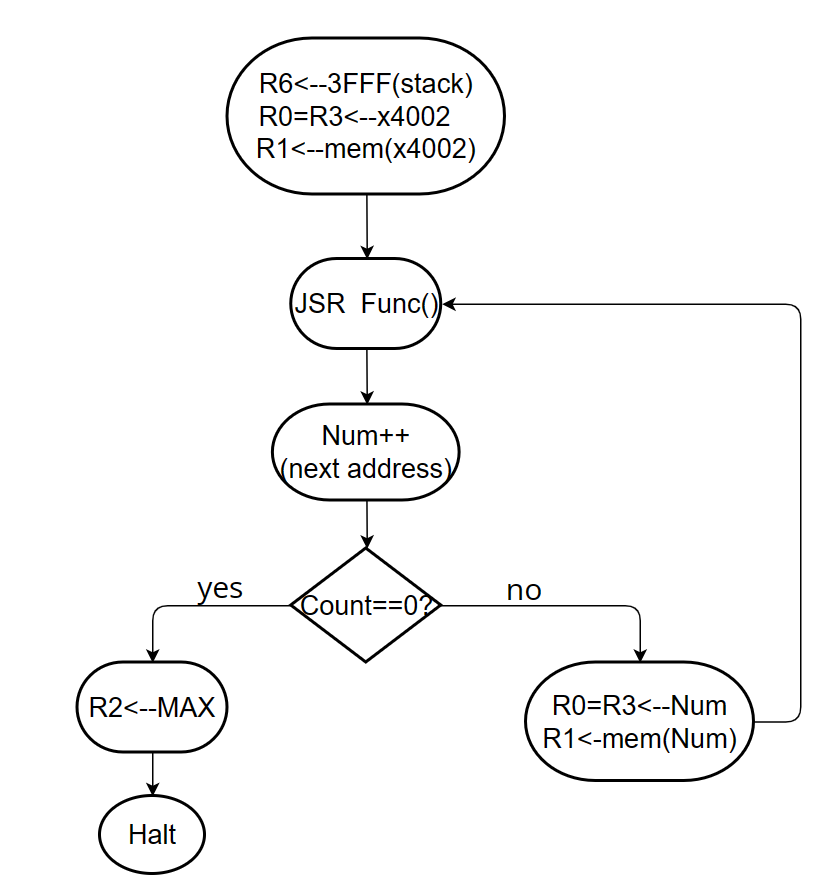
1. Algorithm

① ②

①①

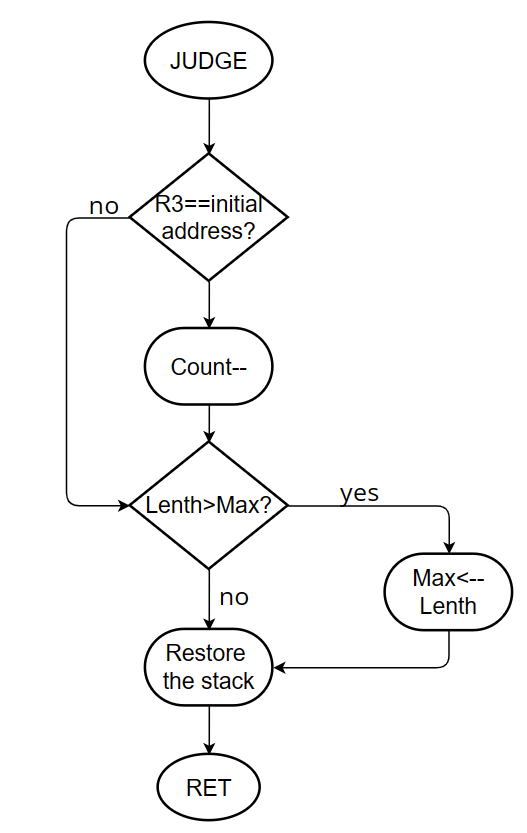


F() function:

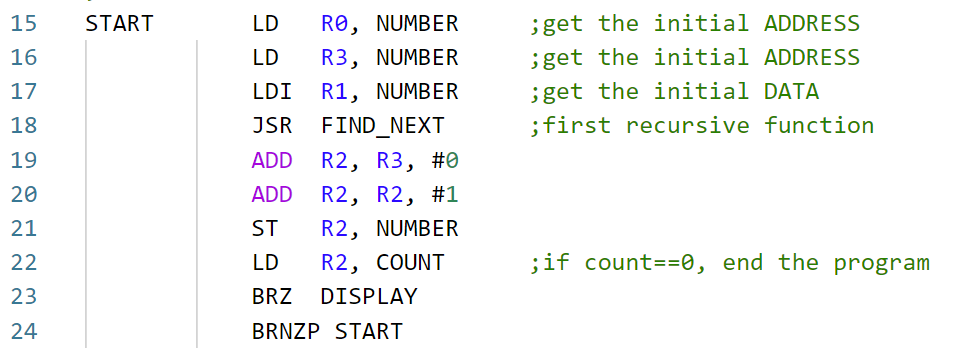


1. Essential parts
2. Q&As

③

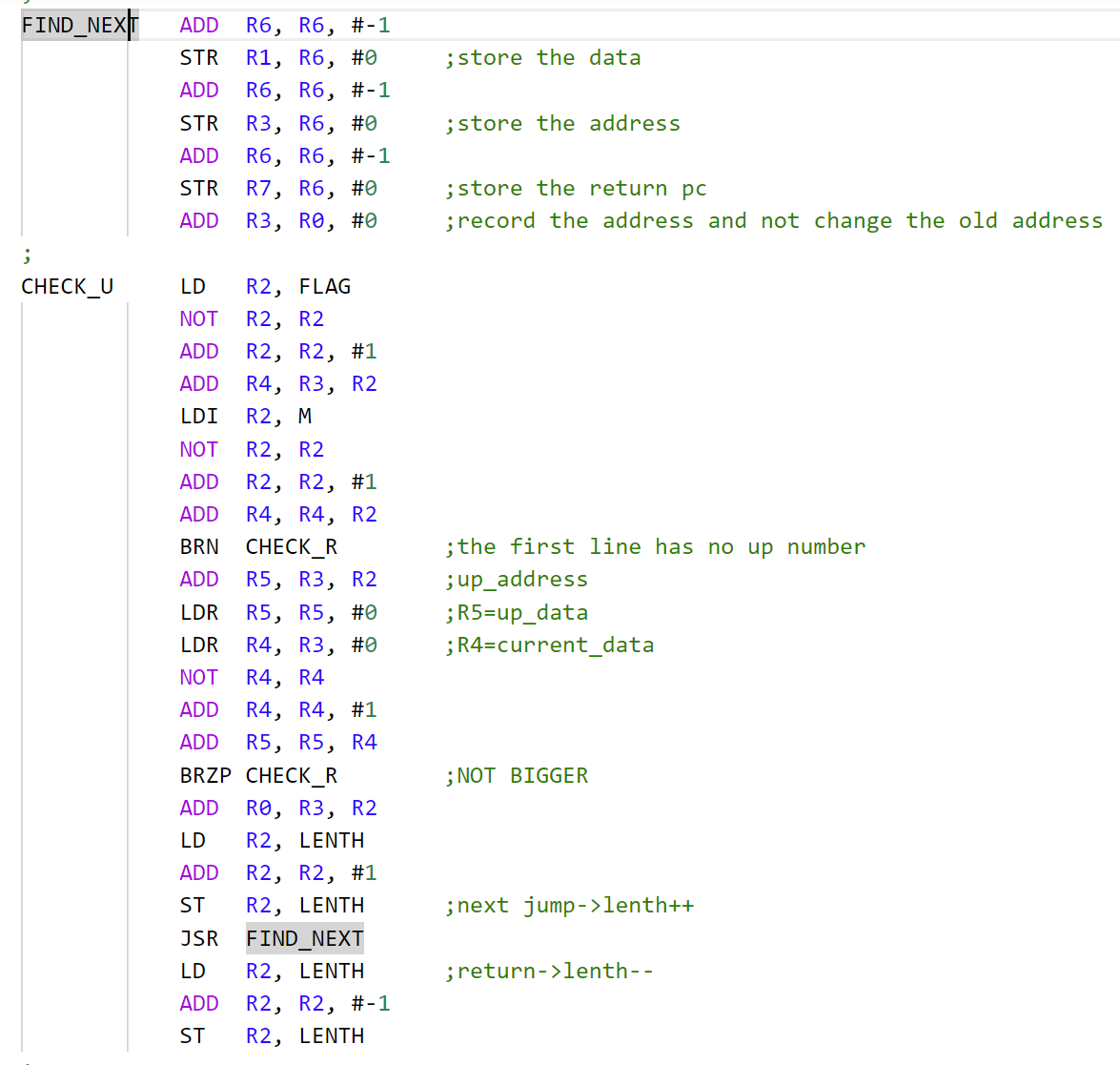


1. Essential parts

(1)

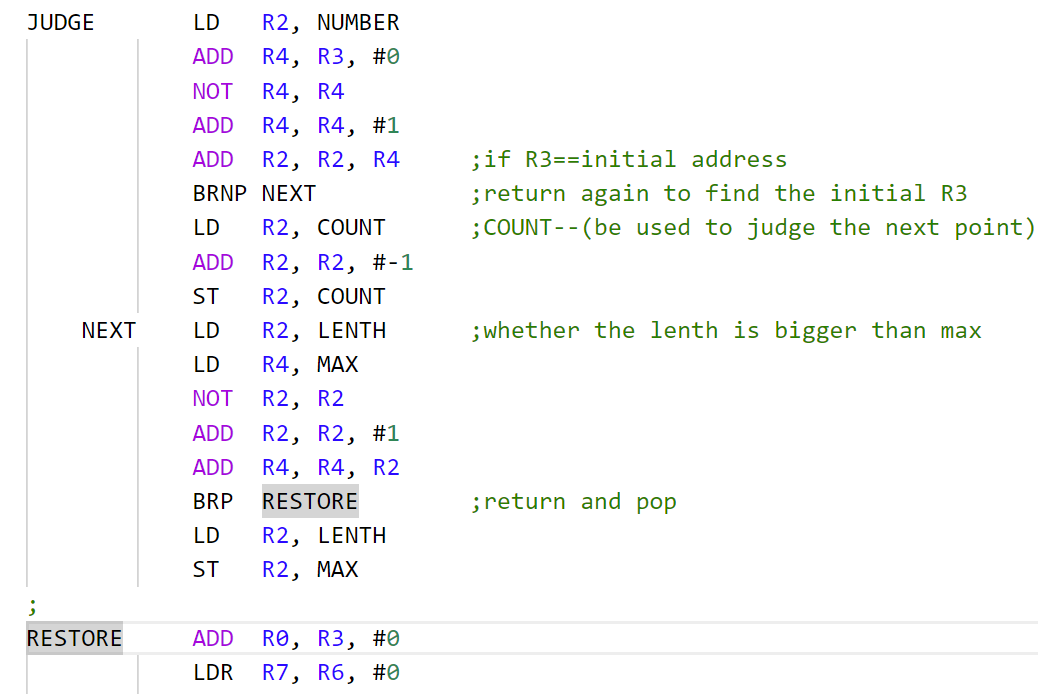
The register is initialized before the recursive function starts, R3 stores the address, R1 stores the data, R3 is saved in stack memory before each recursive call. Since there are N\*M starting points to judge, a Count is designed to record the number of starting points that have been considered.

1. Function parts



For example, the CHECK\_U command to check whether the current address is in the top row. If yes, go to the CHECK\_R command to check whether the current address is in the rightmost row; If no, check whether the current data is larger than the data stored in the previous address. If it is not, skip to check whether it is in the rightmost row. If so, modify R0 to the new address, call the F function recursively, and the corresponding ‘Lenth’ increment.

CHECK\_R, CHECK\_L, and CHECK\_D are similar, but when judging the leftmost row, jump to the JUDGE section to update the maximum length 'Max' and determine if the starting point needs to be updated.



1. Q&As

Q : What is the maximum number in stack memory?

A : Because I push 3 numbers when calling the function, the maximum number in stack is 50\*3=150.