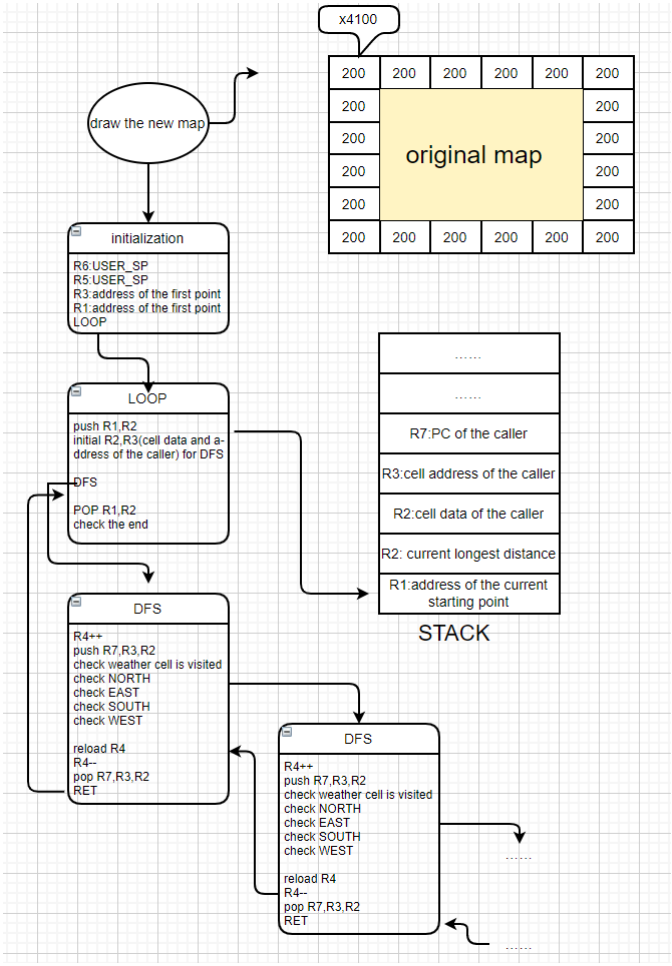
1.algorithm



2.essential parts

```
loop    ADD     R0,R1,#1        ;check every point to get the longest distance
        ADD     R1,R1,#1        ;R1:address of the current starting point
        ADD     R6,R6,#-1       ;push R1,R2
        STR     R1,R6,#0
        ADD     R6,R6,#-1
        STR     R2,R6,#0
        LDR     R2,R0,#0        ;initialize R2,R3（DFS）（cell data and address of
the caller）
        ADD     R3,R0,#0
        JSR     DFS
        LDR     R2,R6,#0        ;pop R2,R1
        ADD     R6,R6,#1
        LDR     R1,R6,#0
        ADD     R6,R6,#1
        LDR     R7,R1,#0        ;check weather the address is the address after
end of the map
        ADD     R7,R7,#2
        BRz     END1
        BR      loop
```

```
DFS     ADD     R4,R4,#1        ;the recursion
        ADD     R6,R6,#-1       ;push R7,R3,R2
        STR     R2,R6,#0
```

```
        ADD     R6,R6,#-1
        STR     R3,R6,#0
        ADD     R6,R6,#-1
        STR     R7,R6,#0
        ADD     R3,R0,#0
        LDR     R2,R3,#0        ;if data of the current address is 200,return
DONE
        LD      R1,VALUE1
        NOT     R1,R1
        ADD     R1,R1,#1
        ADD     R1,R1,R2
        BRz     DONE
        AND     R2,R2,#0        ;put breadcrumb in the current cell
        ADD     R2,R2,#-1

NORTH   LDI     R7,M            ;tmp variable:R7,R0,R1
        ADD     R7,R7,#2
        NOT     R7,R7
        ADD     R7,R7,#1
        ADD     R7,R7,R3        ;R7:address of the north cell
        LDR     R0,R7,#0        ;R0:data of the north cell
        BRn     EAST            ;if a breadcrumb is in the north cell,check the
EAST
        NOT     R0,R0
        ADD     R0,R0,#1
        LDR     R1,R3,#0        ;R1: data of the current cell
        ADD     R0,R0,R1
        BRnz    EAST
        ADD     R0,R7,#0        ;put the address of the north cell in R0
        JSR     DFS
```

3.Q&A

Q:Your program fail to pass the 2*25 random test at first, so could you please give detailed explanation of that problem and how you solved that?

A: To judge weather the next piont jumped to is legal, I redraw a new map which set boarder around the old map and load the address after the map with 0.However,if the data of current starting point is 0,my program will take it as the end of the searching and the value stored in R2 will be 1. To avoid misunderstanding, I load the address after the map with -2. Only when the data of the starting address is -2,the program will halt.