NYUAD Computer Systems Programming PROJECT #2

Project Deadline: Apr 30, 2014, 11:59pm, No late submission

In this project, you are going to write a **word search puzzle solver**. Your program will read a 20x20 2D character array as input and a list of 30 words to search for. You will search the given words in the puzzle. Words can be any mix of uppercase and lowercase. The words may reside in vertical, horizontal, or diagonal directions, either left-right, up-down or vice versa (**See sample run below for more explanation**).

You will output, for each word, whether the word is found in the puzzle and its length. If the word is found in the puzzle, you will also output its start position (row and column number), its direction (vertical, horizontal, diagonal), and whether it is reversed or not. In your program, this information should be contained in a **structure you will define**.

NOTES:

- For horizontal and vertical cases, assume that a word is not reversed if it is in left-right or up-down direction, and reversed if it is in right-left or down-up direction, respectively. For diagonal cases assume that a word is not reversed if it is in down-right or up-right direction and reversed if it is in down-left or up-left direction.
- If the same word occurs twice, you can print only one of the solutions.
- The start position is the starting row and column number of the word, where the first row and column is numbered as 1 and goes up to 20 (unlike C arrays, where the indicies start with 0 and goes up to N-1)

Check the sample run in the next page for more information.

GRADING:

- Indentation: 10 points
- Meaningful variable naming: 10 points
- Comments: 10 points (do not write too much comments, but write necessary ones)
- Correct structure definition type (variables, access functions): 30 points
- Correct execution of the program: 40 points

SUBMISSION:

You'll email the source code (the entire project using zip format) to me at <yfang@nyu.edu>.

SAMPLE RUN:

Text in **bold** is your program's input, plain text are your program's outputs.

Input:

```
XEQMRJKOWRGHWLKOBMET
AHEKGCACNANRUTEREHCH
TSLVEYEMQPXHLRNUNUYF
LDBZRHRBAFDEFINERSJE
UUUYGABFUOUBUTPTNIMC
AUOWFJQNSLJXVKSQDORO
FXDXVRCWCOTDINTEGERM
ESNEOTGVBNPKXLTROHSP
DPGDISIYRGBFCRIQIFPI
GYNOETKUARREDULCNIEL
FXNFUASNNRVUCCGFTPWE
CTGENTTSWMRXNUANPYCR
FGCDIIUIQUNAUBLSIMOI
IHTETCNGCKLEFSIZERPB
ECXPNRANMPETVFFHQOTN
DTDYOJTEEWQVEUYOIGPS
NIKTCAADZQHLFRPNEJVO
AWXDPCOWMJSIOBTMSZKS
KSFOJNLEWEXGLEXNNOIF
WLTWJWFNNVBWREJCKDMS
FLOAT
DOUBLE
INTEGER
ARRAY
FUNCTION
LOOP
WHILE
IFELSE
SWITCH
CASE
STRUCT
POINTER
STRING
CHARACTER
BREAK
CONTINUE
BOOLEAN
INCLUDE
DEFINE
COMPILER
LINKER
INTERPRETER
RETURN
TYPEDEF
UNSIGNED
DEFAULT
SIZEOF
STATIC
LONG
SHORT
```

Your output:

```
FLOAT: found, 5, (20,7), vertical, reversed
DOUBLE: found, 6, (7,3), vertical, reversed
INTEGER: found, 7, (7,13), horizontal, not reversed
ARRAY: found, 5, (13,12), diagonal, reversed
FUNCTION: found, 8, (4,10), diagonal, reversed
LOOP: not found
WHILE: found, 5, (16,10), diagonal, not reversed
IFELSE: found, 6, (14,15), diagonal, reversed
SWITCH: found, 6, (19,2), vertical, reversed
CASE: found, 4, (11,14), diagonal, not reversed
STRUCT: found, 6, (6,15), diagonal, not reversed
POINTER: found, 7, (14,19), diagonal, reversed
STRING: found, 6, (16,20), diagonal, reversed
CHARACTER: not found
BREAK: found, 5, (5,7), vertical, reversed
CONTINUE: found, 8, (17,5), vertical, reversed
BOOLEAN: not found
INCLUDE: found, 7, (10,18), horizontal, reversed
DEFINE: found, 6, (4,11), horizontal, not reversed
COMPILER: found, 8, (5,20), vertical, not reversed
LINKER: not found
INTERPRETER: not found
RETURN: found, 6, (2,16), horizontal, reversed
TYPEDEF: found, 7, (17,4), vertical, reversed
UNSIGNED: found, 8, (10,8), vertical, not reversed
DEFAULT: found, 7, (9,1), vertical, reversed
SIZEOF: found, 6, (20,20), diagonal, reversed
STATIC: found, 6, (9,6), vertical, not reversed
LONG: found, 4, (6,10), vertical, not reversed
SHORT: found, 5, (8,19), horizontal, reversed
```