

Joseph Ingenito

CSC 230-02

Project 1

Spring 2020

Struct Node:

Char chr

Bool solution

Int main(int argc, char **argv):

Int x,y

Input x, y

Vector<vector<Node>> matrix

Matrix.resize(x)

For i in range(x)

Matrix[i].resize(y)

For j in range(y)

Node n

Input n.chr

n.solution = false

matrix[i][j] = n

for each word in argv

findWord(matrix, word)

printMatrix(matrix)

return 0

```
void findWord(vector<vector<Node>> &matrix, char* word):
```

```
    for i in range(matrix.size())
```

```
        for j in range(matrix[0].size())
```

```
            //check all possible directions
```

```
            if checkWord(matrix,word,i,0,j,1)
```

```
                drawWord(matrix,word,i,0,j,1)
```

```
            if checkWord(matrix,word,i,0,j,-1)
```

```
                drawWord(matrix,word,i,0,j,-1)
```

```
            if checkWord(matrix,word,i,1,j,0)
```

```
                drawWord(matrix,word,i,1,j,0)
```

```
            if checkWord(matrix,word,i,-1,j,0)
```

```
                drawWord(matrix,word,i,-1,j,0)
```

```
            if checkWord(matrix,word,i,1,j,1)
```

```
                drawWord(matrix,word,i,1,j,1)
```

```
            if checkWord(matrix,word,i,1,j,-1)
```

```
                drawWord(matrix,word,i,1,j,-1)
```

```
            if checkWord(matrix,word,i,-1,j,-1)
```

```
                drawWord(matrix,word,i,-1,j,-1)
```

```
            if checkWord(matrix,word,i,-1,j,1)
```

```
                drawWord(matrix,word,i,-1,j,1)
```

```
void drawWord(vector<vector<Node>> &matrix, char* word, int x, int delX,
```

```
    int y, int delY):
```

```
    int count = 0
```

```
    while count < length(word)
```

```
        matrix[x][y].solution = true
```

```
        count++
```

```
        x += delX
```

```
        y += delY
```

```

bool checkWord(vector<vector<Node>> matrix, char* word, int x, int delX,
               int y, int delY):
    int count = 0
    while count < length(word)
        if x<0 or x>= matrix.size() or y<0 or y>=matrix[0].size()
            return false
        if matrix[x][y].chr != *(word + count)
            return false
        count++
        x += delX
        y += delY

    return true

void printMatrix(vector<vector<Node>> matrix, char* word, int x, int delX,
                 int y, int delY):
    for i in range(matrix.size())
        for j in range(matrix[0].size())
            if matrix[i][j].solution
                print (matrix[i][j].chr, " ")
            else
                print("* ")
    print("") //new line

```