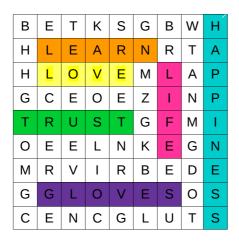
CSE108 LW 07

•Using mobile phones, flash disks, internet and any other record or communication media is strictly forbidden during lab sessions. Throughout a lab session, all such media must be kept turned off and in a closed environment. Violation of this rule is punished with a grade 0, -100 or worse. Before doing anything else, make sure that your computer is not attached any such media.

•Make sure that you have deleted all of your work PARMANENTLY before leaving the first



GOAL;

In this lab, You will create a Nx9 puzzle and fill it from the file called "puzzle.txt". In main ,ask the player for the word , the location as the starting point (x,y) and direction (across 'a' or down 'd'). If the word is found in the specified location, replace the word with '*' characters on the puzzle. Repeat the process while the player wants to continue.

PART-I (2 PT)

•Write a function that reads the matrix elements from the file called "puzzle.txt". Open and close the file in this function.

void fill_puzzle_matrix(char m[][SIZE], int* n);
//or int fill puzzle matrix(char m[][SIZE]);

PART-II (1 PT)

•Write a function that prints the matrix elements passed as input parameter. **void print_puzzle(const char m[][SIZE], int n);**

PART-III (2 PT)

•Write a function that checks whether the word is in the puzzle or not. The function takes puzzle matrix **m**, the size of the matrix n, the string **word**, the location on x axis, location on y axis and the direction d ('a' represents across and 'd' represents down) as input parameters. You should define an enumeration type bool for true and false. The function should return a value of type bool. You may **strlen** function because it's not allowed.

bool is same word(const char m[][SIZE], int n, const char word[], int x, int y, char d);

PART-IV (2 PT)

Write a function that replaces the characters of the word with the character '*'. The function takes puzzle matrix \mathbf{m} , the size of the matrix \mathbf{n} , the location on \mathbf{x} axis, location on \mathbf{y} axis, the direction d ('a' represents across and 'd' represents down) and number of characters num_c as input parameters. It should replace num_c characters starting from location (\mathbf{x} , \mathbf{y}) on the direction d with the character '*'.

•void mark_puzzle(char m[][SIZE], int n, int x, int y, char d, int num_c);

PART-V (2 PT)

Test your functions in main by;

- creating an puzzle matrix taking the elements from the file called "puzzle.txt"
- ullet asking the player ford the word,the location as the starting point(x,y) and direction
- If the word is found in speficified location
 - replace the word with '*' characters on the puzzle.txt
 - give info about you found the word and replace it
 - print the matrix
- Otherwise
 - warn the player about you havent found the word in specified location
 - print the matrix
 - Repeat the process while the player wants to continue.

PART-VI - BONUS PART (2 PT)

•Write a function that checks whether the word is in the puzzle or not. The function takes puzzle matrix **m**, the size of the matrix n, the string **word** as input parameters. The difference of this part from the Part3 is that the function should founds the location point and direction ,and an integer pointer that has the point : x, y, and direction d (0 represents across and 1 represents down) respectively.

void is_found(const char m[][SIZE], int n, const char word[],int* params);

Enter the word:Learn
Enter the starting point of the word on

Enter the starting point of the word on puzzle matrix:1

1

Enter the direction [across:a, down: d]:a The word is found and replace with '*' character.

В	E	T	K	S	G	В	W	Н
Н	*	*	*	*	*	R	T	A
M	С	T	D	U	T	L	A	P
G	С	E	O	E	Z	I	N	P
T	R	U	S	T	G	F	M	I
O	E	E	L	N	K	E	G	N
M	R	V	I	R	В	E	D	E
G	G	L	O	V	E	S	O	S
С	E	N	С	G	L	U	Т	S

Do you want to continue ? [yes(y) or no(n)] : y

Enter the word:Happiness

Enter the starting point of the word on puzzle matrix:1

1

Enter the direction [across:a, down: d]: d
The word is not found.

В	E	T	K	S	G	В	W	Н
Н	*	*	*	*	*	R	Т	A
M	С	T	D	U	T	L	A	P
G	С	E	O	E	Z	I	N	P
Т	R	U	S	T	G	F	M	I
О	E	E	L	N	K	E	G	N
M	R	V	I	R	В	E	D	E
G	G	L	O	V	E	S	O	S
С	E	N	С	G	L	U	Т	S

Do you want to continue ? [yes(y) or no(n)]: n