



❖ THE HANGMAN DEVELOPER DOCUMENTATION

I. OVERVIEW OF THE DESIRED PROGRAM

The player must guess a word composed of a define number of character. The word is represented by dashes. Each turn the player should suggests a letter. If the letter occurs in the word it should take the place of one of the dashes or else it will be considered as failed turn. The process repeat until all the character are guessed and the word is fully clear

II. PROGRAM EXPLANATION

The game consist of three main part followed by the choice to replay if wanted via the use of recursion into main function

I. Settings

It consist of one function that return the desired difficulty (number of error allowed) named respectively difficulty

II. Word choice

In this part we choose the word to be guessed. We begin by asking the user to provide the desired length

Of the word by a function named `length_of_word` .

then we proceed by randomly selecting a word with appropriate size from a file named `words` .the selection is done by a function named `wordf` that contain in itself a `randomize` function. then we proceed by the use of `dashes_creation` function to print the masked word into the screen.

III. Playing

Using the function `playing` we keep asking the player to guess one letter at a time until either the word is completely guessed or his number of error exhausted . to do so ,we ask of player to provide a character .in the `letter` function

we check if it is a part of the alphabet and not a special character or a symbol then with lowercase function we make sure its a lowercase character or else we transform it into one.

We check afterward if the letter has already been used by the function use then we store it in a list to avoid reusing it .finally with the help of check function we reveal the letter in all it correct positions or add one to the number of misses

III. EXPECTED BEHAVIOUR FOR ANY POSSIBLE USER INPUT

The player should provide the number of misses acceptable if it is reached the following text will appear « game over »

The player should provide the length of the word and an equivalent number of dashes will appear

With each guess if it is true the program should reveal the letter in all it correct positions else add one to the number of misses

If the player guess a special letter or a symbol the following text will appear « Your guess can not be accepted. Please chose a letter from alphabet »

IV. EXISTING FUNCTION IN THE PROGRAM

```
int main() return itself if player want to replay{
    int difficulty() return d number of error allowed
    int length_of_word() return length of the word
    void dashes_creation(char *dashes) create dashes string
    void wordf(int length,char*word)pick a word randomly from file
    {
        int randomize(int length) pick a random number between
        1..20 }
    .
    int use (listuse *head,char c) check if the letter was used before
    void playing(int *x,int *y,char*word,char*dashes){
```

```

char letter() return a letter {
    char lowercase(char c)input an alphabet letter return the
    lowercase equivalent
}
int use (listuse *head,char c) check if the letter was used before
int store(listuse **head ,char c )store the letter in a file
int check(char *s1,char *s2,char c,int*y) reveal the letter (c) in all it
correct positions or add one to the number of misses (y)
}
void deleteliste ( listuse * head )delete list of used letters to free
memory{
    listuse * deleteletter ( listuse * head ) delete a letter from list
}
}

```