

Wordle Plus

Group members: 郑子欣(2021012926), 都稼川(2021012917),
乔守航(2022013406), 左捷瑞(2022011529)

1. Overview

Wordle is a word-guessing game that went viral online recently. In fact, it is the most searched term according to Google's annual Year in search report in 2022. It has quite simple rules. The original and most basic ones require player to guess a 5-letter word within 6 guesses. After each guess, the system returns hints regarding which characters in the guess are correct. Restriction is that player cannot guess random letters, only words from a list. The player wins if he gets the correct answer within 6 guesses, otherwise he loses.

We have designed some original functionalities in addition to the classic ones online, these functionalities will be shown in the Program functionalities section of this report. During the setup users will have to download 2 txt files, "ALL.TXT" and "SOLUTION.TXT", so the program can execute properly. Further information regarding txt files is included in the Originality and external library section of this report.

2. Program structure

The program consists of 13 functions and 1 main function. Their functionalities are as follow.

int menu(void):

Function: As a hub, players choose to start the game, view help, or exit.

Code logic: First, use puts to show the function and operation of the menu to the player, and then use a larger array buffer to receive the player's input. Then use the variable scan and while loop to check whether the input is normal. After checking, the buffer will be compared with exit, help and new respectively. If the corresponding operation is triggered all the time, the help will open the function help(), while exit will return -1, new will return 1, and the return value will act as a condition in the main() function to trigger other operations.

The above whole (excluding puts) is in a while loop where the condition is true, that is, if the return is not triggered, the above operation will loop infinitely.

void random(void) :

Function: Randomly read a word stored in an array word from the existing word file SOLUTION as the answer to the game.

Code logic: First, use the defined variable an initialization of 0 with array s and the for loop of nested while loops to measure the total number of words of SOLUTION named i, and then use fseek to bring the file pointer back to the beginning. Then use rand to generate a random number from 0 to i named e. Then use the for loop of the nested while loop to adjust the file pointer to the beginning of the eth word, and then use the for loop to fill the eth word into the array work. Finally, use fseek to bring the file pointer back to the beginning.

int modeChooser(void):

Function: Choose the game mode, the single-player mode or the double-player mode

Code logic: based on the menu function. First pass the return value of the menu function to the variable gameId, Then the if-else conditional judgment, If the gameId is greater than 0. Then call the random and gameLoop_single function into the single-player mode. The modeChooser function returns a value of 1; If gameId equals 0, Then call the gameLoop_double function into the two-person mode, It also uses the clock function. Finally, the outcome is judged by the nesting of two if-else statements and the modeChooser function returns a value of 1; In other cases, the modeChooser returns a value of -1.

void playMusic(void):

Function: Play the music 'Go Home'

Code logic: open the '.wav' file with the PlaySound function.

Note: The music file must be '.wav' file.

int answerChecker(const char *guess):

Function: to judge the accuracy of the user input

Code logic: First, establish a variable copy of the same type as the variable word, and upload the same value as the word with the for-for loop. Then, judge each element : if an element in guess is the same as the element in the same position in copy, the element is output in green, and then replace the element in the position in copy with '_' to prevent

it from affecting subsequent; otherwise, enter the next layer for loop to judge, that is, if the element exists in copy, the yellow output is used; if the element does not exist in Guess, the element is output in red.

int toLower(char* str):

Function: Turn uppercase letters to lowercase

Code logic: first use the strlen function to get the string length, and then use a for loop to turn the elements in the string one by one with the toLower function to lowercase.

int allLetter(const char* word):

Function: examine whether the supplied word only contain lowercase letters.

Code logic: First determine whether the length of the supplied word is correct. If so, use for loop to examine whether every element is a character, use variable sum to record the number of letters in the supplied word. If the value of sum is equal to the correct length, it means the supplied word is consists of letters, and the function will return 1. Otherwise, it will return 0.

int inList(const char* word):

Function: examine whether the supplied word is contained in the list

Code logic: if the word has a correct length and consists of letters, we will search whether the word is in the list ALL. The search will start from the words with the same initial letter. And it will not search beyond the words with the same initial letter. Then

we compare the supplied word with words in the list. Variable high/low indicates the probable position, i.e., if the supplied word is contained, it must be after/before the word we now use to compare. If the two matches, we will return 1, means the supplied word is in the list.

void wordleLoop_single(); int wordleLoop_double();

Function: get the game running and operate in a loop

Code logic: Players can only guess at most 5 times. Invalid words or word with right length but not contained in the list will not use the guess times, and player will receive “answer is not in list!”. If the player input word with wrong length, the function will remind him and tell him the correct length. If the character in guess is also contained in the answer, the character will be removed from the alphabet.

Once the guess is right, the game will end, and print “You win!”(single) or “Success”(double) , otherwise it will print “You lose”(single) or “Failure”(double) and tell the players the right answer.

void help(void);

Function: Output the game description

Code logic: use ‘puts’ and ‘printf’ to output the introduction to Wordle.

void Answer(char *word, int Len);

Function: present the given ‘answer’ with ‘*****’ and meanwhile ,input a value to

`'answer'`

Code logic: Use the function `'getch()'` to achieve the function that the answer should be hidden in double-player mode.

`int pickAnswer(char * answer, int index);`

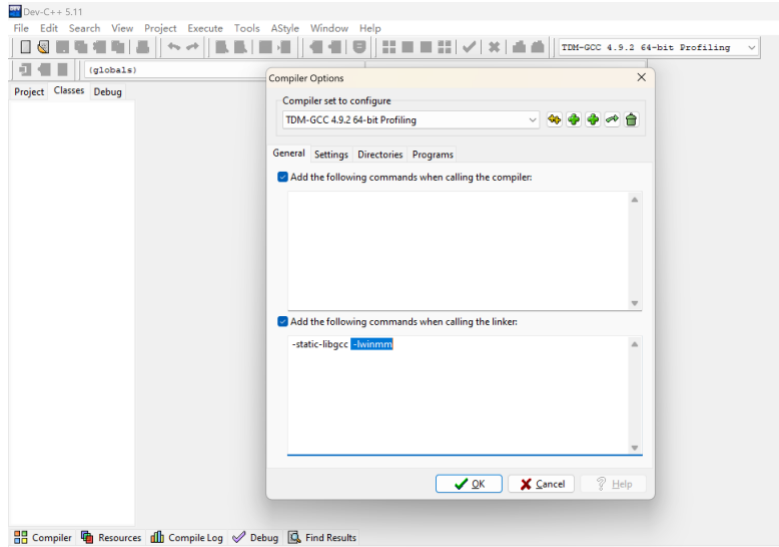
Function: Pick the given word in the file `'SOLUTION.TXT'`

Code logic: Define `'i'` to compare with `'index'` which is the location of the answer in the file. While `'i'` is smaller than `'index'`, using `'++i'` and do the above until `'i'` equals `'index'`. Finally, return `'index'`.

3. Language and IDE

The programming language used is C language, and the IDE used for program development is Dev-C++.

Notify that to enable the music function, you need to set your Dev-C++ compiler options to adding command `" -lwinmm"` when calling the linker. The following screenshot shows how to complete the setting.



4.Demonstration

First, the music “Go Home” is played when the program is executed to relax you for the game.

Then, there will be a menu for you to choose functions from.

```
now playing : Go Home

Please read the game description in 'introduction' carefully before your first game.

Menu:
  Please choose:
    single: single player mode
    double: double player mode
    help:  information about the game
    exit:  end game

Please enter your choice:
```

If you are new for this game, you are recommended to choose the ‘help’ first .And you will see the following:

```

help
-----
A brief introduction to 'Wordle'

There are two modes to be chosen:
1.Single player mode :
    The computer picks a word at random as the 'word'
    Guess the 5 letters of the word within 6 tries,and then you win.
2.Double player mode :
    One player( Wordle-maker ) enter a word as the 'word' , the other( Wordle-guesser ) guess the word
    Then the two players switch roles.
    If one of them guess out the word successfully while the other fail,the one who succeed is the winner.
    If they both guess out the word ,then check their number of attempts ,the player with less attempts is the
winner.
    If they both guess oyt the word with the same number of attempts ,then check their time ,the player using l
ess time is the winner. Otherwise the game is tied
You can choose one mode.

After every guess,each character will be shown in a certain color.
They will be presented like this:
    red      means Character not found at all
    green    means Character found and position correct
    yellow   means Character found but position wrong
Unused letters of the alphabet are shown next to the hint
The game prefers valid positions over invalid positions,
And it handles double letters properly.
Guessing "RATES" when the word is "TESTS" shows "RATES"

```

After that, you have already known the basic playing method of the game.

Let's start the game!

If you are alone to play this game, then choose the 'single' for single-player mode.

```

Please enter your choice:
single

answer unused alphabet

Guess 1:

```

Then enter your guess. You only have 6 chances. So be careful!

After your each guess, the program will get back the result:

each character will be shown in a certain color:

- red-not in the answer;
- yellow-found in the answer but position wrong;
- green-found in the answer and position right;

Meanwhile, the alphabet will be presented. If a character is in red, that means it has been used. Otherwise, it has not.

```

answer unused alphabet

Guess 1: about
about abcdefghijklmnpqrstuvwxy
Guess 2:

```


Finally, if you guess out the answer successfully:

```
answer unused alphabet
Guess 1: about
about abcdefghijklmnopqrstuvwxy
Guess 2: binge
binge abcdefghijklmnopqrstuvwxy
Guess 3: catch
catch abcdefghijklmnopqrstuvwxy
Guess 4: caste
caste abcdefghijklmnopqrstuvwxy
Guess 5: cabin
You win!
time = 102.000000 seconds
```

If you fail to guess out the answer:

```
answer unused alphabet
Guess 1: topic
topic abcdefghijklmnopqrstuvwxy
Guess 2: break
break abcdefghijklmnopqrstuvwxy
Guess 3: board
board abcdefghijklmnopqrstuvwxy
Guess 4: blind
blind abcdefghijklmnopqrstuvwxy
Guess 5: bites
bites abcdefghijklmnopqrstuvwxy
Guess 6: biota
biota abcdefghijklmnopqrstuvwxy
You lose. The answer was bigot
time = 219.000000 seconds
-----
```

The right answer will also be shown to you, like the ‘bigot’ in the above picture.

And if you are looking for a game for double player, this one is the best choice, too.

We have ‘double’ mode for you.

```
Please enter your choice:
double
Player 1 , please enter your answer.
|
```

Then just follow the remind: Player 1 should enter a 5-character word as the ‘answer’.

And the input will be presented as ‘*****’ on the screen to protect it from the sight of

Player 2 .

```
Player 1 , please enter your answer.
*****
Player 2 , now start your game!
answer unused alphabet
Guess 1: |
```

And it’s turn for Player 2.

The game loop is like the process mentioned above:

```

Please enter your choice:
double
Player 1 , please enter your answer.
*****
Player 2 , now start your game!
answer unused alphabet

Guess 1: break
break abcdefghijklmnopqrstuvwxyz
Guess 2: weigh
weigh abcdefghijklmnopqrstuvwxyz
Guess 3: abase
abase abcdefghijklmnopqrstuvwxyz
Guess 4: abeit
answer is not in list!

Guess 4: about

Success
time1 = 41.000000 seconds
Player 2 , please enter your answer.

```

Then ,after guessing ,the Player 2 should enter a word as the ‘answer’, after which is Player 1’s turn. As the following picture:

```

Player 2 , please enter your answer.
*****
Player 1 , now start your game!
answer unused alphabet

Guess 1: weigh
weigh abcdefghijklmnopqrstuvwxyz
Guess 2: about
about abcdefghijklmnopqrstuvwxyz
Guess 3: ccraft

Invalid answer. Must be 5 characters

Guess 3: craft
craft abcdefghijklmnopqrstuvwxyz
Guess 4: crash
crash abcdefghijklmnopqrstuvwxyz
Guess 5: crack

Success
time2 = 85.000000 seconds

```

And as you can notice, the used time will be presented after the game for each player.

The timing function is realized to decide who is the winner.

The rules for double-player mode:

- If one of them guess out the answer successfully while the other fails, the one who succeed is the winner.

For example:

```

Player 2 , now start your game!
answer unused alphabet

Guess 1: about
about abcdefghijklmnopqrstuvwxy
Guess 2: crack
crack abcdefghijklmnopqrstuvwxy
Guess 3: crash
crash abcdefghijklmnopqrstuvwxy
Guess 4: craft
craft abcdefghijklmnopqrstuvwxy
Guess 5: quick

Success
time1 = 20.000000 seconds

Player 2 , please enter your answer.
*****
Player 1 , now start your game!
answer unused alphabet

Guess 1: about
about abcdefghijklmnopqrstuvwxy
Guess 2: break
break abcdefghijklmnopqrstuvwxy
Guess 3: craft
craft abcdefghijklmnopqrstuvwxy
Guess 4: crash
crash abcdefghijklmnopqrstuvwxy
Guess 5: crack
crack abcdefghijklmnopqrstuvwxy
Guess 6: crawl
crawl abcdefghijklmnopqrstuvwxy
Failure. The answer was crazy
time2 = 30.000000 seconds

Player 2 win the game!

```

- If they both guess out the answer, then check their number of attempts they have used, the player with less attempts is the winner.

For example:

```

Please enter your choice:
double
Player 1 , please enter your answer.
*****
Player 2 , now start your game!
answer unused alphabet

Guess 1: break
break abcdefghijklmnopqrstuvwxy
Guess 2: abase
abase abcdefghijklmnopqrstuvwxy
Guess 3: about
about abcdefghijklmnopqrstuvwxy

Success
time1 = 12.000000 seconds

Player 2 , please enter your answer.
*****
Player 1 , now start your game!
answer unused alphabet

Guess 1: about
about abcdefghijklmnopqrstuvwxy
Guess 2: break
break abcdefghijklmnopqrstuvwxy
Guess 3: crack
crack abcdefghijklmnopqrstuvwxy
Guess 4: crash
crash abcdefghijklmnopqrstuvwxy
Guess 5: quick

Success
time2 = 27.000000 seconds

Player 2 win the game for fewer attempts!

```

- If they both guess out the answer with the same number of attempts, then check their time, the player using less time is the winner.

For example:

```

Please enter your choice:
double
Player 1 , please enter your answer.
*****
Player 2 , now start your game!
answer unused alphabet

Guess 1: break
break abcdefghijklmnopqrstuvwxyz
Guess 2: abase
abase abcdefghijklmnopqrstuvwxyz
Guess 3: about

Success
time1 = 8.000000 seconds

Player 2 , please enter your answer.
*****
Player 1 , now start your game!
answer unused alphabet

Guess 1: craft
craft abcdefghijklmnopqrstuvwxyz
Guess 2: crash
crash abcdefghijklmnopqrstuvwxyz
Guess 3: crack

Success
time2 = 10.000000 seconds

Player 2 win the game for using less time!

```

- Otherwise the game is tied.

For example,

```

Guess 1: crack
crack abcdefghijklmnopqrstuvwxyz
Guess 2: break
break abcdefghijklmnopqrstuvwxyz
Guess 3: abase
abase abcdefghijklmnopqrstuvwxyz
Guess 4: crash
crash abcdefghijklmnopqrstuvwxyz
Guess 5: craft
craft abcdefghijklmnopqrstuvwxyz
Guess 6: weigh
weigh abcdefghijklmnopqrstuvwxyz
Failure. The answer was about
time1 = 28.000000 seconds

Player 2 , please enter your answer.
*****
Player 1 , now start your game!
answer unused alphabet

Guess 1: about
about abcdefghijklmnopqrstuvwxyz
Guess 2: crack
crack abcdefghijklmnopqrstuvwxyz
Guess 3: crash
crash abcdefghijklmnopqrstuvwxyz
Guess 4: cabin
cabin abcdefghijklmnopqrstuvwxyz
Guess 5: break
break abcdefghijklmnopqrstuvwxyz
Guess 6: about
about abcdefghijklmnopqrstuvwxyz
Failure. The answer was weigh
time2 = 27.000000 seconds

The game is tied!

```

In addition, if you want to exit the program, just input 'exit' when choosing function from the menu.

```

now playing : Go Home

Please read the game description in 'introduction' carefully before your first game.

Menu:
    Please choose:
    single: single player mode
    double: double player mode
    help: information about the game
    exit: end game

Please enter your choice:
exit

-----
Process exited after 5.282 seconds with return value 0
请按任意键继续. . .

```

The Valid Input is presented above in the pictures.

Examples for Invalid Input are as below.

- Invalid Input when choosing functions:

```
Please enter your choice:
ssingle
Invalid input
Please enter your choice:
```

You will be asked to input again.

- Invalid Input during the game:

Word is not in the list:

```
Guess 1: aaaaa
answer is not in list!
```

The word length is illegal:

```
Guess 1: weight
Invalid answer. Must be 5 characters
```

And you will be asked to guess without the increase of the number of attempts.

- Invalid Input when enter 'answer' in the double-player mode:

```
Please enter your choice:
double
Player 1 , please enter your answer.
*****
Invalid answer!!!
Restart the game.

Please read the game description in 'introduction' carefully before your first game.

Menu:
Please choose:
single:  single player mode
double:  double player mode
help:   information about the game
exit:   end game

Please enter your choice:
|
```

You will be asked to choose a function again.

5.Format for input and output

Input: 1. Enter <single> or <double> or <exit> or <help> on the menu page to trigger different functions.

2. Enter five-letter words during the game to guess the correct answer, and adjust the input repeatedly through the feedback received.

3. All your input is supposed in lowercase.

No space is expected between the characters.

(Standard Input is also mentioned clearly in the above pictures)

Output: After comparing the words guessed by the player with the correct answer, the player will give feedback to the player the relevant information in different forms. The details are as follows:

After each guess, each corresponding letter will be displayed in a certain color.

They will be presented as follows:

Red indicates that the character was not found at all, green indicates that the character and location are correct, and yellow indicates that the character is found but the position is wrong.

(Note: If the character is found but the character appears more than the number of times, the answer is also red.)

The unused letters in the alphabet are displayed next to the prompt, and the used letters are marked in red.

```

answer unused alphabet
Guess 1: about
about abcdefghijklmnopqrstuvwxy
Guess 2: binge
binge abcdefghijklmnopqrstuvwxy
Guess 3: catch
catch abcdefghijklmnopqrstuvwxy
Guess 4: caste
caste abcdefghijklmnopqrstuvwxy
Guess 5: cabin
You win!
time = 102.000000 seconds

```

6. External library

```
#include <stdlib.h>
```

We use function `exit()` to stop the game from running. `EXIT_SUCCESS` indicates the game is successfully exit, `EXIT_FAILURE` indicates the reverse.

```
#include <time.h>
```

We use the function `srand (time (NULL))` random seed to get random numbers so that words can be randomly fetched and calculate the time taken by both parties in the double mode.

```
#include <ctype.h>
```

Used for the function 'toLower' and 'Answer'.

```
#include <windows.h>
```

Used to present the outcomes with different colors and play background music

```
#include <conio.h>
```

Used for `getch()` in double-player mode.

```
#include<mmsystem.h>
```

Used to play background music

ALL.TXT

We set a range for the valid word, i.e., all the valid guesses must come from this list.

SOLUTION.TXT

All possible words that can be a solution.

7. Program highlights

(1) we divide the program into several parts and do function encapsulation. It benefits the reuse of code, also improve the conciseness and readability of the program.

(2) we set up a menu. Players can choose which mode of game they want to play. Player can play the wordle singly. It is also available for 2 players to compete in the game. Tries and time they use will be used to determine who is the winner. If they have any problem, they can click “help” to get more information about the game.

(3) To improve the experience, player can listen to music “Go Home” while playing the game. Also, we use different colors to tell the players how close their guesses are. These designs improve the visual and auditory perception of players.

8.Reference

Wordle-solver <https://github.com/Shreenabh664/Wordle-solver>

We use for reference when designing the game logic and functions we want to achieve.