#include<stdio.h> /\*Header file declaration\*/

#include<string.h> /\*<string.h> for strcmp();,strlen(); functions use\*/

#include<stdlib.h>

void showHangman(int);

int main(void)

{

char hangmanWord[100], tempWord[100]; /\*\*hangmanWord[] array for the original word and tempWord[] array to get the alphabet from user and compare it with original word\*\*/

char hangmanOutput[100]; /\*\*This array will show the remaining blanks and correct inputs\*\*/

int wrongTry = 6 , matchFound = 0; /\*\*player will get 5 chance, so we use wrongTry as chance counter\*\*/

/\*\*matchFound to search the alphabet, if the alphabet from user does not exist

in the original word it will remain 0, upon finding the word, matchFound will

be set as 1\*\*/

int counter = 0 , position = 0, winner, length , i;

char alphabetFromUser;

system("cls"); /\*\*for clearing the screen\*\*/

printf("\n\n Enter any word in small case and hit >>ENTER<<");

printf("\n\n\t Enter HERE ==> ");

scanf("%s",hangmanWord); /\*\*get the string from opponent\*\*/

printf("\n\n Now give the COMPUTER to your friend and see if he/she can CRACK it!!!");

printf("\n\n\tHIT >>ENTER<<");

getchar(); /\*\*hold the computer screen\*\*/

length = strlen(hangmanWord); /\*\*get the length of the word\*\*/

system("cls");

printf("\n\n !!!!!!!!!!!!!!!!!!!Welcome to the HANGMAN GAME!!!!!!!!!!!!!!!!!\n\n\n"); /\*\*Brief description of the game\*\*/

printf("\n\n You will get 5 chances to guess the right word");

printf("\n\n So help the Man and get...set...GO..!!");

getchar();

printf("\n\n\tHIT >>ENTER<< ");

getchar();

system("cls");

printf("\n\t||===== "); /\*\*show the HANGMAN\*\*/

printf("\n\t|| | ");

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\n The word has %d alphabets \n\n",length); /\*\*tell the user how many alphabets the word has\*\*/

for( i = 0; i < length ; i++)

{

hangmanOutput[i] = '\_';

hangmanOutput[length] = '\0';

}

for(i = 0 ; i < length ; i++)

{

printf(" ");

printf("%c",hangmanOutput[i]); /\*\*Show the Word With n(length of the original word) number of underscores (\_)\*\*/

}

while(wrongTry != 0) /\*\*while loop for exiting the program when no try left\*\*/

{

matchFound = 0;

printf("\n\n enter any alphabet from a to z and please use small case!!");

printf("\n\n\t Enter HERE ==> ");

fflush(stdin);

scanf("%c",&alphabetFromUser); /\*\*get alphabet from user\*\*/

if(alphabetFromUser < 'a' || alphabetFromUser > 'z') /\*\*In case player gives input other than 'a' to 'z' the console will ask again\*\*/

{

system("cls");

printf("\n\n\t Wrong input TRY AGAIN ");

matchFound = 2;

}

fflush(stdin);

if (matchFound != 2)

{

for(counter=0;counter<length;counter++) /\*\*for loop to check whether player input alphabet exists or not in the word\*\*/

{

if(alphabetFromUser==hangmanWord[counter])

{

matchFound = 1;

}//end of if()

}//end of for()

if(matchFound == 0) /\*\*in case of wrong guess\*\*/

{

printf("\n\t :( You have %d tries left ",--wrongTry);

getchar();

showHangman(wrongTry);

getchar();

}//end of if()

else

{

for(counter = 0; counter < length; counter++)

{

matchFound = 0;

if(alphabetFromUser == hangmanWord[counter])

{

position = counter ;

matchFound = 1;

}//end of if

if(matchFound == 1)

{

for(i = 0 ; i < length ; i++)

{

if( i == position)

{

hangmanOutput[i] = alphabetFromUser; /\*\*Put the alphabet at right position\*\*/

}

else if( hangmanOutput[i] >= 'a' && hangmanOutput[i] <= 'z' ) /\*\* If the position already occupied

by same alphabet then no need to

fill again EASY!! and continue \*/

{

continue;

}

else

{

hangmanOutput[i] = '\_'; /\*\* Put a blank at not guessed alphabet position \*\*/

}

}

tempWord[position] = alphabetFromUser; /\*\*put the alphabet in another char array to check with the original word\*\*/

tempWord[length] = '\0'; /\*\*put the NULL character at the end of the temp string\*\*/

winner = strcmp(tempWord,hangmanWord); /\*\*upon True comparison it will return 0\*\*/

if(winner == 0) /\*\*if the player guessed the whole word right then he/she is the WINNER\*\*/

{

printf("\n\n\t \t YAHOO!!!!! You are the WINNER !!!!!");

printf("\n\n\t The Word was %s ",hangmanWord);

printf("\n\n\n\n\t\tEASY HUH???\n\n");

getchar();

return 0;

}//end of inner if

}//end of outer if

}//end of for loop

}//end of else

}// end of if(matchFound != 2) condition

printf("\n\n\t");

for(i = 0 ; i < length ; i++)

{

printf(" ");

printf("%c",hangmanOutput[i]); /\*\*Show the original Word With blanks and right Input alphabet\*\*/

}

getchar();

}//end of while loop

if(wrongTry <= 0) /\*\*if the player can not guess the whole word in 5 chaces\*\*/

{

printf("\n\n\t The Word was %s ",hangmanWord);

printf("\n\n\t The man is dead you IDIOT!!!!!");

printf("\n\n\t Better luck next!!!");

}

getchar();

return 0;

}//end of main();

void showHangman(int choice) /\*\*This function show the hangman after each wrong try\*\*/

{

switch(choice)

{

case 0:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| %cO/",'\\');

printf("\n\t|| | ");

printf("\n\t|| / %c",'\\');

printf("\n\t|| ");

break;

case 1:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| %cO/",'\\');

printf("\n\t|| | ");

printf("\n\t|| %c",'\\');

printf("\n\t|| ");

break;

case 2:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| %cO/",'\\');

printf("\n\t|| | ");

printf("\n\t|| ");

printf("\n\t|| ");

break;

case 3:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| %cO/",'\\');

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\t|| ");

break;

case 4:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| %cO ",'\\');

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\t|| ");

break;

case 5:

system("cls");

printf("\n\t||===== ");

printf("\n\t|| | ");

printf("\n\t|| O ");

printf("\n\t|| ");

printf("\n\t|| ");

printf("\n\t|| ");

break;

}//end of switch-case

return;

}