GROUP MEMBERS:

HARIS JAVED

UZAIR HUSSAIN

SHAHWAIZ SATTAR

HAMZA HAMEED

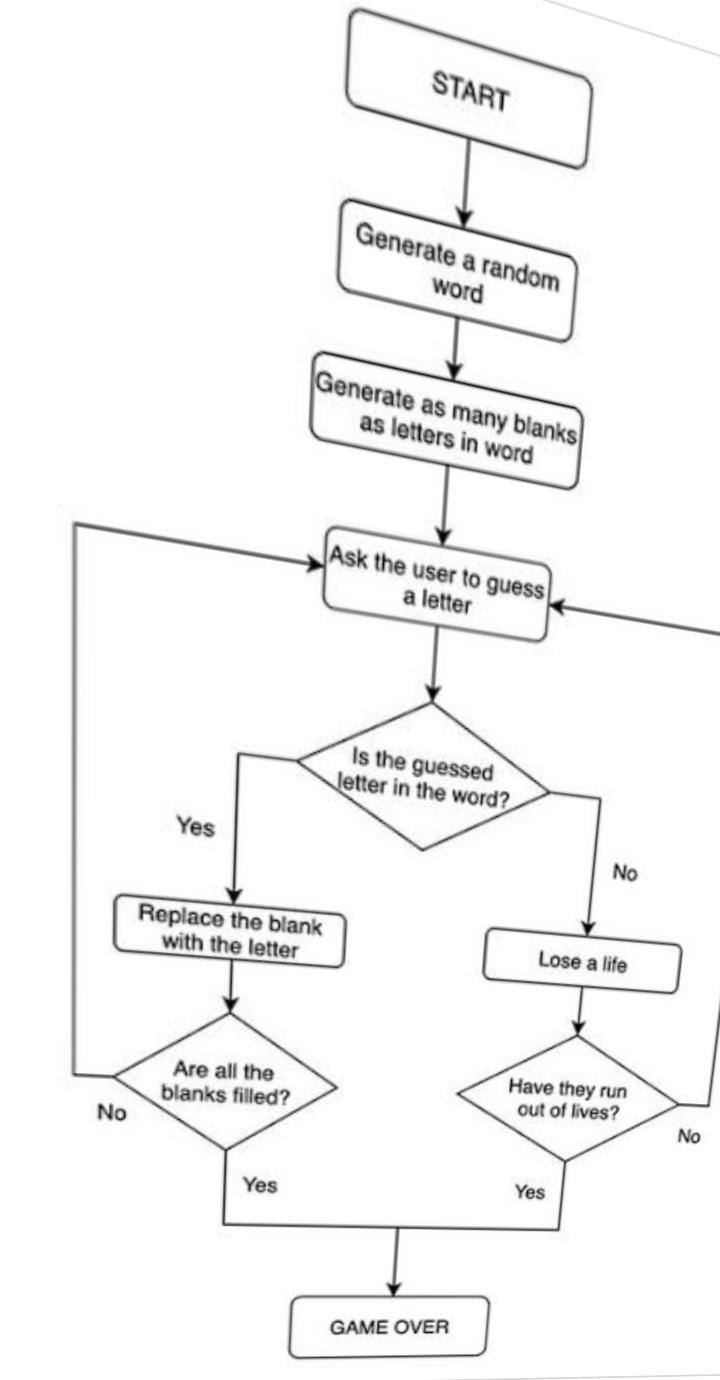
CP LAB REPORT: **HANGMAN GAME**

INTRODUCTION OF PROJECT:

Simplified hangman game. In this version the computer program randomly chooses a target string and ask the user to suggest letters that occurs in the string after each guess the program shows user the a version of a target string that replace letters that have not been guessed with the stars (\*) along with a count of a wrong guess that have been made so far .the game is over when the users wins correctly by guessing the string or losing by making more than assigned incorrect guessing.

OBJECTIVE AND SCOPE:

We are designing a world popular hangman game .it’s purpose is to turn this paper game into a computerized version. In it User has option to play game. The hangman will be drawn automatically if your chose an incorrect word. If you finish the first level, you will be taken to the next level with a new word to be guessed.

FLOW CHART OF C++hangman:

WORKING OF PROJECT:

Idea was to working c++program to play a hangman word guessing game. Program had to be able to show how long the word was, takes in letter guessed, recognize if the letter is correct or not, recognizing a repeated letter, and know what to do and when the word was guessed.

Program include these major parts:

* Getting a random word from the file already created.
* Turning this word to a hidden string made up of character ‘\*’.
* Setting the useful variables including Guess variable (which performs certain if statements) and Score/Lives variable (which performs and displays the Hangman figure according to the switch case).
* Taking input from the user as guessing of letter and checking if it’s correct or incorrect.
* Displaying output (in term of stick figure ASCII art) unless out of lives (GAME OVER) or guessed the whole word (WON).

CONCLUSION OF PROJECT:

This project was challenging and fun for us and we learned few new things and new ways of implementations of code in C++.The code include most library and methods already taught in the semester course for example

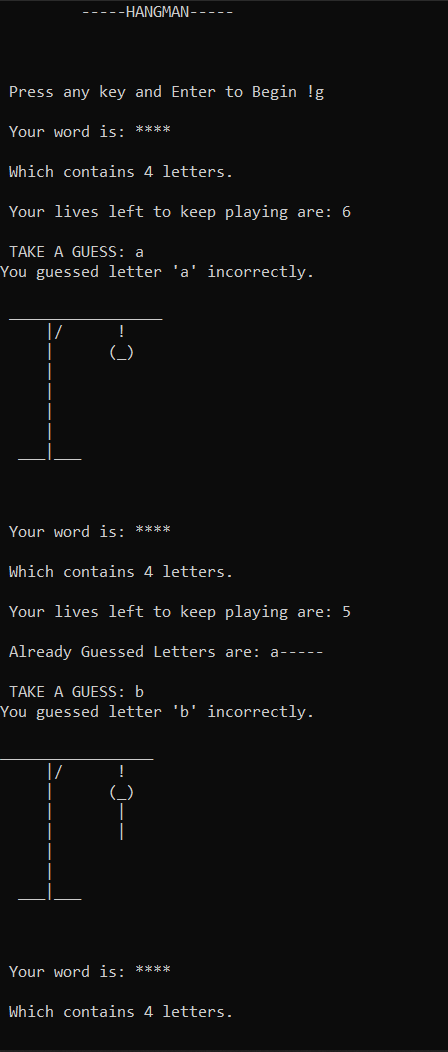
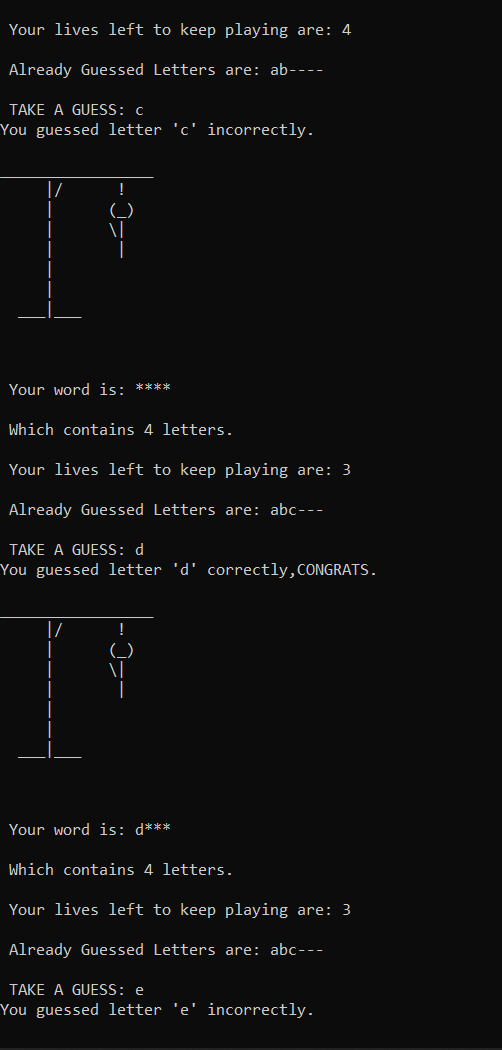
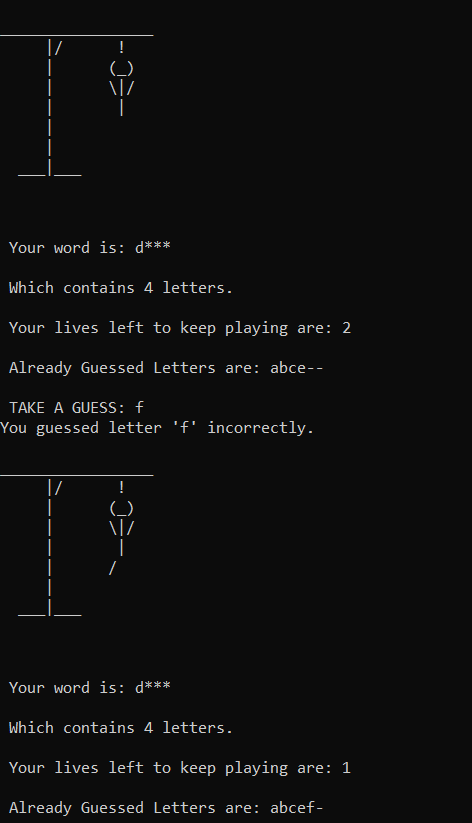
* Reading a file of type .txt using library fstream.
* Using condition statements like if, else if, else and also switch cases.
* Making use of array and strings.
* Creating function by reference.
* Using for loops for writing and reading especially arrays.

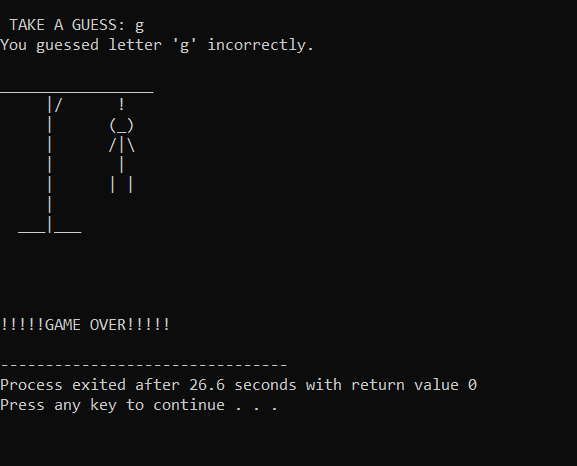
Something new we added to the program was the use of <ctime> library in other to generate a random computer clock time to get a random number each time the code was executed.

<Cstdlib> we use in order to define rand () and srand () functions.

PROJECT PICTURE:

Output Example 1:



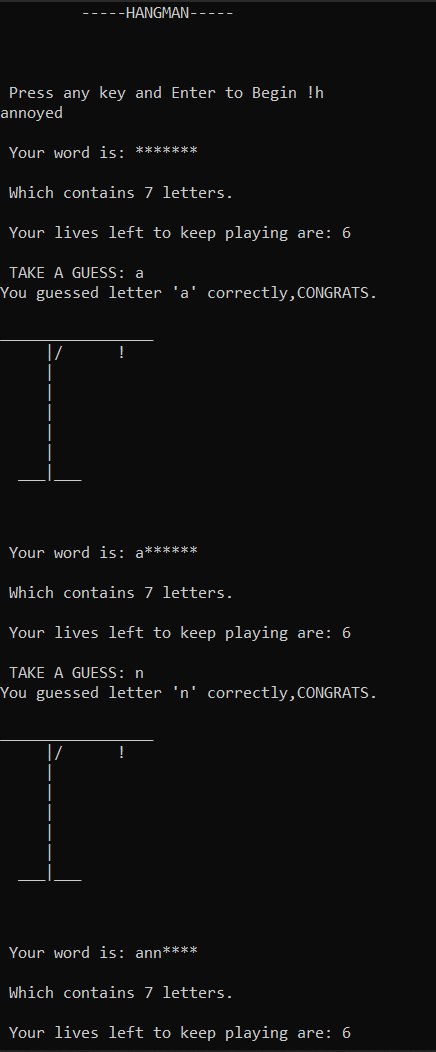
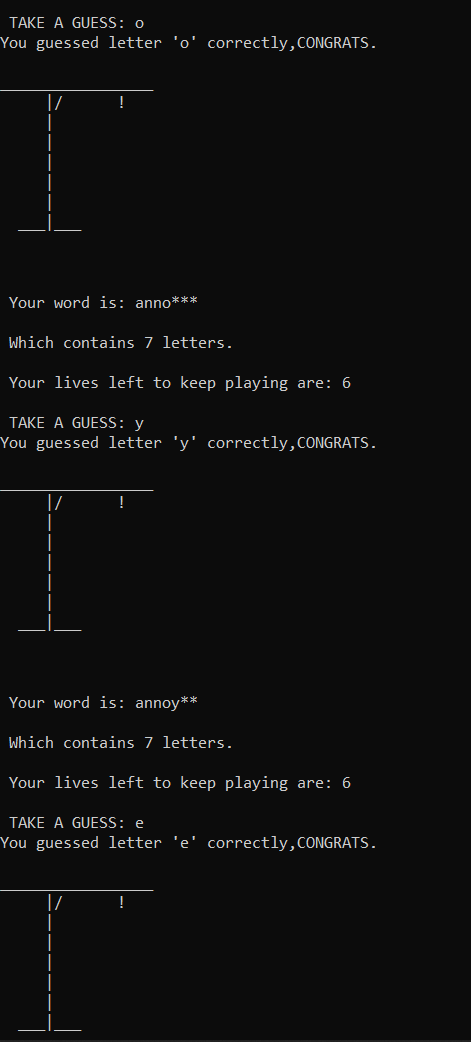
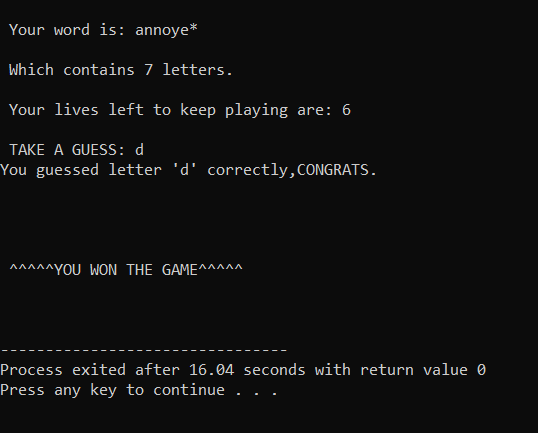
Output Example 1 EXPLANATION:

At first code ask the user to press a key and enter to begin after which the user is displayed a randomly generator word from the 100 word file in the background, showing it in hidden form and the words characteristic length and your lives u will take start with.

With these the user will be asked to enter the guess and the rest of the code runs according to the user guess being correct or in correct.

In this example use guessed the letter wrong, losing all his lives and subsequently losing and being hung while displaying GAME OVER message.

Output Example 2:

Output Example 2 EXPLANATION:

Similar to example one initial setup is ditto however the difference is of user given guesses. The user enter correct guesses consecutively (PS: with cheating in other to give example) and the game displayed WON message in the end.

***CODE:***

#include<iostream>

#include<fstream>

#include<ctime> //Used for getting random word from the file.

#include <cstdlib>// to mae srand() and rand() defined.

#include<string>

#include<string.h>

using namespace std;

void GetWord(string &word){

fstream myfile;

string wordlist[100];

int random;

myfile.open("Word List.txt");

for(int i=0;i<100;i++){

myfile >> wordlist[i];

}

srand(time(0)); //Takes computers clock time and proforms an alorithm to get a random num.

random=rand()%100;

word=wordlist[random];

myfile.close();

}

int main(){

cout<<"\t -----HANGMAN-----\n\n";

char start{};

cout <<"\n\n Press any key and Enter to Begin !";

cin >> start;

int lives=6,x;

string word;

char guess;

char guessed[7]; //one for than guess to prevent garbage value and use for displaying wrong guesses.

guessed[0]='\0';

for(int i=1;i<6;i++){

guessed[i]='-';

}

guessed[6]='\0';

GetWord(word);

//cout << word<<endl; //can be used for checking/cheating.

x=word.length();

string unknown(x,'\*');

//cout << unknown << endl;// is used later.

while(lives>=0){

bool correct=false;

cout << "\n Your word is: "<< unknown << endl;

cout << "\n Which contains " << x<<" letters.\n";

cout << "\n Your lives left to keep playing are: " << lives<< endl;

if(guessed[0]!='\0'){

cout <<"\n Already Guessed Letters are: "<< guessed <<endl;

}

cout << "\n TAKE A GUESS: ";

cin >> guess;

//CHECKING GUESS//

for(int i=0;i<x;i++){

if(word[i]==guess){

unknown[i]= guess;

correct=true;

}

}

if(correct==true){

cout <<"You guessed letter '" << guess << "' correctly,CONGRATS. \n";

}

else{

cout <<"You guessed letter '" << guess << "' incorrectly. \n";

lives--;

}

if(word==unknown){

cout<<"\n\n\n\n ^^^^^YOU WON THE GAME^^^^^\n\n\n";

break;

}

switch(lives){

case 6:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | \n";

cout<<" | \n";

cout<<" | \n";

cout<<" | \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

break;

}

case 5:{

cout<<"\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | \n";

cout<<" | \n";

cout<<" | \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

if(correct==false){

guessed[0]=guess;

}

break;

}

case 4:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | | \n";

cout<<" | | \n";

cout<<" | \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

if(correct==false){

guessed[1]=guess;

}

break;

}

case 3:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | \\| \n";

cout<<" | | \n";

cout<<" | \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

if(correct==false){

guessed[2]=guess;

}

break;

}

case 2:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | \\|/\n";

cout<<" | | \n";

cout<<" | \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

if(correct==false){

guessed[3]=guess;

}

break;

}

case 1:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | \\|/\n";

cout<<" | | \n";

cout<<" | / \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

if(correct==false){

guessed[4]=guess;

}

break;

}

case 0:{

cout<<" \n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" |/ !\n";

cout<<" | (\_)\n";

cout<<" | /|\\\n";

cout<<" | | \n";

cout<<" | | |\ \n";

cout<<" |\n";

cout<<" \_\_\_|\_\_\_\n\n\n";

cout<<"\n\n!!!!!GAME OVER!!!!!\n";

lives=-1;

break;

}

}

}

return 0;

}