**OOPCG MICROPROJECT**

Title: Hangman

Problem Statement: Write a C++ program to make a word-guessing game in which system throws a random word and the player tries to guess it by suggesting the letters. The word to guess is represented by a row of dashes, giving the number of letters. If the guessing player suggests a letter which occurs in the word, the program writes it in all its correct positions.  
  
Prerequisite: Data Structures and problem solving, computer graphics.  
  
Objective: To use concepts of Object Oriented Programming and Computer Graphics to design a computerized version of traditional pen-paper Hangman.

Theory:

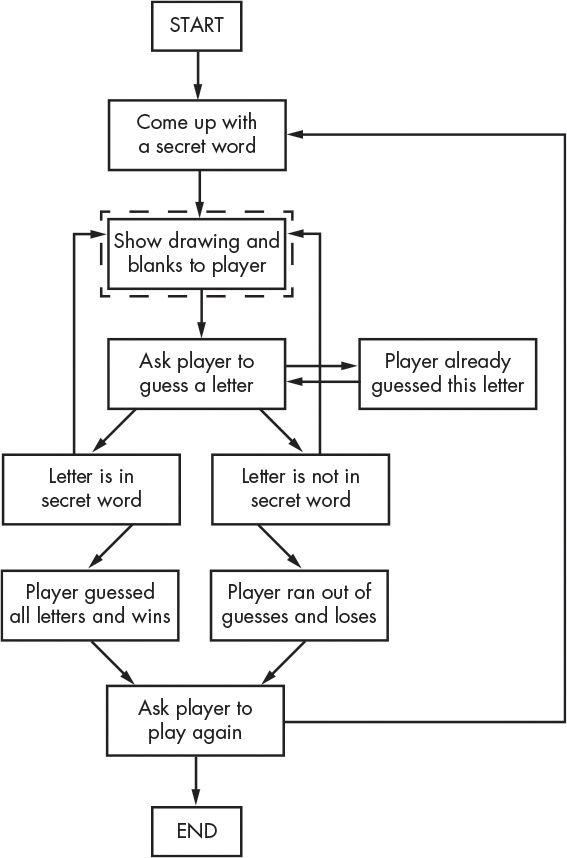
The overall architecture of the game can be thought of having following main points which consists of the functionalities mentioned below:  
1. Pick a random word.   
2. Take the player’s guess.   
3. Quit the game if the player wants to.   
4. Check that the player’s guess is a valid letter.   
5. Keep track of letters the player has guessed.   
6. Show the player their progress.   
7. Finish when the player has guessed the word.

The structured program is represented using the following pseudo-code:

Pick a random word

While the word has not been guessed {   
 Show the player their current progress  
 Get a guess from the player

If the player wants to quit the game {  
 Quit the game   
}   
Else If the guess is not a single letter {  
 Tell the player to pick a single letter  
 }   
Else {  
 If the guess is in the word {   
 Update the player's progress with the guess  
 }   
 }   
}   
Congratulate the player on guessing the word



By keeping a collection of blank spaces and filling them in as the player guesses correct letters using an array of blanks for each letter in the word. The main game takes place inside a while loop. In this loop we display the current state of the word being guessed (beginning with all blanks); ask the player for a guess (and make sure it’s a valid, single-letter guess); and update the answer array with the chosen letter, if that letter appears in the word.

Concepts used:  
Object Oriented Programming  
Class : A class in C++ is the building block that leads to Object-Oriented programming. It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class.

Eg: class **HANGMAN**{};

Access modifiers : They are used to implement an important aspect of Object-Oriented Programming known as data hiding.  
public - members are accessible from outside the class.

private - members cannot be accessed (or viewed) from outside the class.

protected - members cannot be accessed from outside the class, however, they can be accessed in inherited classes.

Eg: public:  
 int **logic**();

Scope resolution operator : The scope resolutionoperator is used to reference the global variable or member function that is out of scope.

It is used to access the hidden variables or member functions of a program.   
It defines the member function outside of the class using the scope resolution.  
It is used to access the static variable and static function of a class.  
The scope resolution operator is used to override function in the Inheritance

Eg: int **HANGMAN** :: **logic**(){}

Computer Graphics

* Outtextxy():

The header file graphics.h contains **outtextxy()** function which displays the text or string at a specified point (x, y) on the screen.

* Settextstyle()

The function **setstextstyle()** is used to set these three attributes of any text.  
settextstyle(int font , int direction , int charsize);

Conclusion : Thus a computerized version of traditional pen-paper Hangman was designed.