

A Course Based Project Report on

HANGMAN

Submitted to the

Department of Computer Science and Engineering

in partial fulfillment of the requirements for the completion of course
PROGRAMMING FOR PROBLEM SOLVING LABORATORY (22ES1CS101)

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

B.ESHWARI	23071A05E7
B.CHANDRANEEL	23071A05E8
CH.ARJUN	23071A05E9
D.POOJA SREE	23071A05F0
D.RISHI VARMA	23071A05F1

Under the guidance of

DR.D N VASUNDHARA

(Course Instructor)

Assistant Professor, Department of CSE, VNRVJIET



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**VALLURUPALLI NAGESWARA RAO VIGNANA
JYOTHI INSTITUTE OF ENGINEERING &
TECHNOLOGY**

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS,
India

**VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI
INSTITUTE OF ENGINEERING AND TECHNOLOGY**

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA Accredited for CE, EEE, ME, ECE, CSE, EIE, IT B. Tech Courses, Approved by AICTE, New Delhi, Affiliated to JNTUH, Recognized as "College with Potential for Excellence" by UGC, ISO 9001:2015 Certified, QS I GUAGE Diamond Rated
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

This is to certify that the project report entitled "**HANGMAN GAME**" is a bonafide work done under our supervision and is being submitted by **DR D N VASUNDHARA,(23071A05E7)B.ESHWARI,(23071A05E8)B.CHANDRANEEL REDDY,(23071A05E9)CH.ARJUN,(23071A05F0)D.POOJASREE,(23071A05F1)D.RISHI VARMA** in partial fulfillment for the award of the degree of **Bachelor of Technology** in COMPUTER SCIENCE AND ENGINEERING, of the VNRVJIET, Hyderabad during the academic year 2023-2024.

DR.D N VASUNDHARA

Assistant Professor
Department of CSE

Mrs.NAGINI

Associate Professor & HOD
Department of CSE

Course based Projects Reviewer

**VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI
INSTITUTE OF ENGINEERING AND TECHNOLOGY**

An Autonomous Institute, NAAC Accredited with 'A++' Grade,
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



DECLARATION

We declare that the course-based project work entitled “HANGMAN GAME” submitted in the Department of Information Technology, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfilment of the requirement for the award of the degree of **Bachelor of Technology in COMPUTER SCIENCE AND ENGINEERING** is a bonafide record of our own work carried out under the supervision of **DR D N VASUNDHARA, Assistant Professor, Department of CSE, VNRVJIET.** Also, we declare that the matter embodied in this thesis has not been submitted by us in full or in any part thereof for the award of any degree/diploma of any other institution or university previously.

Place: Hyderabad.

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved President, Sri. D. Suresh Babu, VNR Vignana Jyothi Institute of Engineering & Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our beloved Principal, Dr. C.D Naidu, for permitting us to carry out this project.

We express our deep sense of gratitude to our beloved Professor, Associate Professor and Head of the Department of COMPUTER SCIENCE AND ENGINEERING, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad-500090 for the valuable guidance and suggestions, keen interest and through encouragement extended throughout the period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved Guide, DR D N VASUNDHARA, Assistant Professor in COMPUTER SCIENCE AND ENGINEERING, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, for his/her valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion of our project work.

B.ESHWARI	23071A05E7
B.CHANDRANEEL	23071A05E8
CH.ARJUN	23071A05E9
D.POOJA SREE	23071A05F0
D.RISHI VARMA	23071A05F1

ABSTRACT

Hangman is a project which aims in developing a computerized version of the popular word guessing game. In this project, Hangman game code is developed using C programming language. This hangman game is about guessing letters (A-Z) to form the words. This game is a common word guessing game in which the player must guess one letter at a time to complete a missing word. In this project, design is straightforward and clean, making it easy for users to learn, use, and navigate.

It is a game that people can play during their free time. It can also be used for educational purposes.

This hangman game source code in c language ends after a certain number of incorrect guesses, and the player loses. If the player correctly recognizes all of the letters in the missing title, the game is also over. In this project design is straightforward and clean, making it easy for users to learn, use, and navigate. It is game which people can play during their free time. It can also be used for educational purposes.

TABLE OF CONTENTS

S NO.	TOPIC	PAGE NUMBER
1	ABSTRACT	1
2	INTRODUCTION	3
3	SOURCE CODE	4
4	TEST CASES / OUTPUT	12
5	CONCLUSION	15
6	REFERENCES	16

INTRODUCTION

1.1 PROBLEM DEFINITION

The Hangman game poses challenges in creating an engaging and educational experience. Designing an effective word selection mechanism, implementing a user-friendly interface, and balancing difficulty levels are key challenges. Ensuring the game remains enjoyable while promoting language learning and critical thinking skills presents a multifaceted problem requiring thoughtful solutions.

1.2 OBJECTIVE

The objective of the Hangman game is to provide an entertaining and educational experience for players. Focused on enhancing language skills, the game aims to challenge users in guessing a hidden word or phrase by suggesting letters. The primary goal is to strike a balance between enjoyment and cognitive engagement, fostering language learning and problem-solving abilities.

CHAPTER-2

SOURCE CODE

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

#include <ctype.h>

int main() {

    char words[][21] = {"brindaavanam", "malliswari", "oxygen", "aswathama",
        "murari", "intersteller", "anabelle", "titanic", "joker", "orphan", "raees",
        "roy", "krishh", "bhootnath", "baazigar", "missamma", "ishq", "adbhutam",
        "policeodu", "rakshasudu", '\0'};

    int i, len, movie, mistakes = 0;

    char game[50], guessed[26], tracker[50], alpha;

    srand(time(0));          // Brings the cursor to 0, everytime executing the program

    movie = rand() % 21;      // Generates a random number

    len = strlen(words[movie]);

    // Displaying Dashes

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

        printf("_\t");
```



```

// Take input from the user

strcpy(game, words[movie]);

for (i = 0; i < len; i++){

    tracker[i] = '_';

}

while (mistakes < 7) {

    printf("\nEnter a letter:\n ");

    scanf(" %c", &alpha);

    alpha = tolower(alpha); // Convert to lowercase


    // Check if the letter is already guessed

    int alreadyGuessed = 0;

    for (i = 0; i < strlen(guessed); i++) {

        if (alpha == guessed[i]) {

            printf("Already Guessed\n");

            alreadyGuessed = 1;

            break;

        }

    }

    if (alreadyGuessed) {

        continue;

    }

```

```

int flag = 1;

// Check if the guessed letter is correct
for (i = 0; i < len; i++) {
    if (alpha == game[i]) {
        tracker[i] = alpha;
        flag = 0;

        if (strcmp(tracker, game) == 0) {
            printf("Congratulations! You guessed the word: %s\n", game);
            return 0; // Exit the program after successful guess
        }
    }
}

// Print the current state of the word
for (i = 0; i < len; i++) {
    printf("%c\t", tracker[i]);
}

printf("\n");

// Handle wrong guess
if (flag == 1) {
    printf("%c is not present in the word\n", alpha);
    mistakes++;
}

```

```
switch (mistakes){
```

```
case 1:
```

```
    printf("-----\n");  
    printf("|          |\n");  
    printf("|          |\n");  
    printf("|\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    printf("\\n");  
    break;
```

```
case 2:
```

```
    printf("-----\n");  
    printf("|          |\n");  
    printf("|          |\n");  
    printf("|          @  @\n");  
    printf("|          !\n");  
    printf("|          -----\n");  
    printf("\\n");
```

```

printf("\n");

printf("\n");

printf("\n");

printf("\n");

break;

```

case 3:

```

printf("-----\n");

printf("          |\n");

printf("          |\n");

printf("          @  @\n");

printf("          !\n");

printf("          -----\n");

printf("          |\n");

printf("          |\n");

printf("          |\n");

printf("          | \n");

printf("\n");

break;

```

case 4:

```

printf("-----\n");

printf("          |\n");

printf("          |\n");

printf("          @  @\n");

```

```

printf("|          !\n");
printf("|          -----\n");
printf("|          |\n");
printf("|          // |\n");
printf("|          |\n");
printf("|          | \n");
printf("|\n");
break;

```

case 5:

```

printf("-----\n");
printf("|          |\n");
printf("|          |\n");
printf("|          @  @\n");
printf("|          !\n");
printf("|          -----\n");
printf("|          |\n");
printf("|          // | \\\n");
printf("|          |\n");
printf("|          | \n");
printf("|\n");
break;

```

case 6:

```

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

```

```

printf("|          |\n");
printf("|          @  @\n");
printf("|          !\n");
printf("|          -----\n");
printf("|          |\n");
printf("|          // | \\\n");
printf("|          |\n");
printf("|          // | \n");
printf("\n");
break;
case 7:
printf("-----\n");
printf("|          |\n");
printf("|          |\n");
printf("|          @  @\n");
printf("|          !\n");
printf("|          -----\n");
printf("|          |\n");
printf("|          // | \\\n");
printf("|          |\n");
printf("|          // | \\\n");
printf("\n");
break;
}

```

```

        // printf("HANGMAN: ");

        // for (i = 0; i < mistakes; i++) {

        //     printf("%c", "HANGMAN"[i]);

        // }

        // printf("\n");


        if (mistakes == 7) {

            printf("You Lost!!!!\n");

            printf("Exceeded number of chances\n");

            printf("The correct word was: %s\n", game);

            return 0; // Exit the program after reaching the maximum number of mistakes

        }

    }

    // Keep tracking the alphabets guessed

    guessed[strlen(guessed)] = alpha;

    guessed[strlen(guessed) + 1] = '\0';


    printf("Guessed so far: %s\n", guessed);

}

return 0;

}

```

CHAPTER-3

TEST CASES/ OUTPUT

[illegible]


```
t
-----
|         |
|         |
|   @   @ |
|   !     |
| ----- |
|         |
|         |
|         |
|         |
|         |
Guessed so far: aifsk

Enter a letter:
t
i _      t _          s    t _ _ _ _ _ 
Guessed so far: aifskt

Enter a letter:
m
i _      t _          s    t _ _ _ _ _ 
m is not present in the word
```

```

-----
|                                     |
|                                     |
|          @      @                  |
|          !                  |
|-----|                          |
|        |                          |
|       // |                       |
|        |                          |
|                                     |
Guessed so far: aifsktm

Enter a letter:
o
_ _ _ _ _ s _ t _ _ _ _ _
o is not present in the word

```

```

-----
|
|
| @ | @
| !
| -----
| // | \\
|
|

```

Guessed so far: aifsktmo

```
Enter a letter:
r
i _ t r s t _ _ _ _ r
Guessed so far: aifsktmor

Enter a letter:
e
i _ t e r s t e _ _ e r
Guessed so far: aifsktmore

Enter a letter:
l
i _ t e r s t e l l e r
Guessed so far: aifsktmorel

Enter a letter:
d
i _ t e r s t e l l e r
d is not present in the word
```

```
-----
|                                     |
|           |                       |
|           @   @                   |
|           |                       |
|           |                       |
|-----|-----|
|           |                       |
|          // |  \\                  |
|           |                       |
|          // |                     |
|-----|-----|
Guessed so far: aifsktmoreld
Enter a letter:
n
Congratulations! You guessed the word: interstellar
```

CHAPTER-4

CONCLUSION

In conclusion, the Hangman code written in C reflects the robustness and versatility of the programming language. Through effective logic and user input handling, the code creates an engaging and functional rendition of the classic word-guessing game. Its simplicity and clarity demonstrate the power of C in developing interactive and entertaining applications, making it a valuable exercise for both learning and application development.

REFERENCES

- [1]. [https://en.wikipedia.org/wiki/Hangman_\(game\)](https://en.wikipedia.org/wiki/Hangman_(game))
- [2]. "Traditional Games of England, Scotland, and Ireland (Vol. I), by Alice Bertha Gomme"., via Project Gutenberg
- [3]. "A White Cap Party". The Philadelphia Inquirer. 16 February 1902. p. 39 – via Newspapers.com.
- [4]. Kapoor, Leena; Prakashan, Goyal Brothers (2021-01-01). Solutions to GET Smart Book for Class 4. Goyal Brothers Prakashan.
- [5]. Griffin, Merv & Bender, David (2007). Merv: Making the Good Life Last. New York: Simon and Schuster. pp. 99–100. ISBN 978-0-7434-5696-8.
- [6]. McLoone, Jon (13 August 2010). "25 Best Hangman Words". Wolfram Research. Archived from the original on 15 August 2010. Retrieved 14 March 2015
- [7]. Foreign English Teacher Under Attack For Playing Hangman Game] "Foreign English Teacher Under Attack for Playing Hangman Game | Japan Probe". Archived from the original on 2010-12-25. Retrieved 2011-03-11., Japan Probe, July 23, 2010