**A Micro Project Report**

**on**

**Problem Solving using C Language**

Submitted by

#### Devarapalli sri raja vardhan reddy(24475A0510)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET (AUTONOMOUS)**

**Accredited by NAAC with A+ Grade and NBA under Tier-1**

**NIRF rank in the band of 201-300 and is an ISO 9001:2015 certified Approved by AICTE, New Delhi, Permanently affiliated to JNTU Kakinada, Approved by AICTE, Accredited by NBA and accredited ’A+’ grade by NAAC Narasaraopet-522601, Palnadu(Dt.), Andhra Pradesh, India**

**2024-20****25**

**NARASARAOPETA ENGINEERING COLLEGE: NARASARAOPET**

**(AUTONOMOUS)**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**CERTIFICATE**

**This is to certify that** Devarapalli sri raja vardhan reddy **rollno : 24475A0510, a Second Year Student of the Department of Computer Science and Engineering, has completed the Micro Project Satisfactorily in “Problem Solving using C Language" for the Academic Year 2024-2025.**.

Project Co-Ordinator HEAD OF THE DEPARTMENT

**Dr. Rama Krishna. Eluri, M.Tech., Ph.D.** **Dr. S. N. Tirumala Rao,** **M.Tech., Ph.D. Asst. Professor Professor**

C programming code

**Aim :**

Develop mimicro project on login\_and\_registration\_system

**Code**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_USERS 100

#define MAX\_USERNAME\_LEN 50

#define MAX\_PASSWORD\_LEN 50

#define FILENAME "users.txt"

// Structure to store user details

typedef struct

{

char username[MAX\_USERNAME\_LEN];

char password[MAX\_PASSWORD\_LEN];

}

User;

// Function prototypes

int register\_user();

int login\_user();

int is\_user\_registered(const char\* username);

void store\_user(const User\* user);

int read\_users\_from\_file(User\* users);

int main()

{

int choice;

while (1)

{

printf("\n\*\*\* Login and Registration System \*\*\*\n");

printf("1. Register\n");

printf("2. Login\n");

printf("3. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice)

{

case 1:

register\_user();

break;

case 2:

login\_user();

break;

case 3:

printf("Exiting system...\n");

exit(0);

default:

printf("Invalid choice. Please try again.\n");

}

}

return 0;

}

// Function to register a new user

int register\_user()

{

User new\_user;

FILE \*file;

char confirm\_password[MAX\_PASSWORD\_LEN];

printf("\nEnter username: ");

scanf("%s", new\_user.username);

if (is\_user\_registered(new\_user.username))

{

printf("Username already exists. Please try again.\n");

return 0;

}

printf("Enter password: ");

scanf("%s", new\_user.password);

printf("Confirm password: ");

scanf("%s", confirm\_password);

if (strcmp(new\_user.password, confirm\_password) != 0)

{

printf("Passwords do not match. Please try again.\n");

return 0;

}

// Store the user in the file

store\_user(&new\_user);

printf("Registration successful!\n");

return 1;

}

// Function to check if a username already exists

int is\_user\_registered(const char\* username)

{

User users[MAX\_USERS];

int user\_count = read\_users\_from\_file(users);

for (int i = 0; i < user\_count; i++)

{

if (strcmp(users[i].username, username) == 0)

{

return 1; // Username found

}

}

return 0; // Username not found

}

// Function to store a new user into the file

void store\_user(const User\* user)

{

FILE \*file = fopen(FILENAME, "a");

if (!file)

{

printf("Error opening file.\n");

return;

}

fprintf(file, "%s %s\n", user->username, user->password);

fclose(file);

}

// Function to read all users from the file into an array

int read\_users\_from\_file(User\* users)

{

FILE \*file = fopen(FILENAME, "r");

if (!file)

{

return 0; // No users in the file

}

int i = 0;

while (fscanf(file, "%s %s", users[i].username, users[i].password) != EOF)

{

i++;

}

fclose(file);

return i; // Return number of users read from the file

}

// Function to handle user login

int login\_user()

{

char username[MAX\_USERNAME\_LEN], password[MAX\_PASSWORD\_LEN];

User users[MAX\_USERS];

int user\_count = read\_users\_from\_file(users);

printf("\nEnter username: ");

scanf("%s", username);

printf("Enter password: ");

scanf("%s", password);

// Search for the user in the file

for (int i = 0; i < user\_count; i++)

{

if (strcmp(users[i].username, username) == 0 &&

strcmp(users[i].password, password) == 0)

{

printf("Login successful! Welcome %s.\n", username);

return 1;

}

}

printf("Invalid username or password.\n");

return 0;

}

Registration Example 1

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 1

Enter username: john

Enter password: hello

Confirm password: hello

Registration successful!

Registration Example 2 (Existing Username)

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 1

Enter username: john

Username already exists. Please try again.

Registration Example 3 (Password Mismatch)

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 1

Enter username: jane

Enter password: hello

Confirm password: world

Passwords do not match. Please try again.

Login Example 1 (Successful Login)

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 2

Enter username: john

Enter password: hello

Login successful! Welcome john.

Login Example 2 (Invalid Credentials)

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 2

Enter username: john

Enter password: wrong

Invalid username or password

Exit Example

\*\*\* Login and Registration System \*\*\*

1. Register

2. Login

3. Exit

Enter your choice: 3

Exiting system...