

MINI PROJECT

MADE BY :-ABHISHEK KUMAR

REG NO :-RA2111027010195

SECTION :-Y1

DEPT :- CSE SEPICALIZATION IN BDA

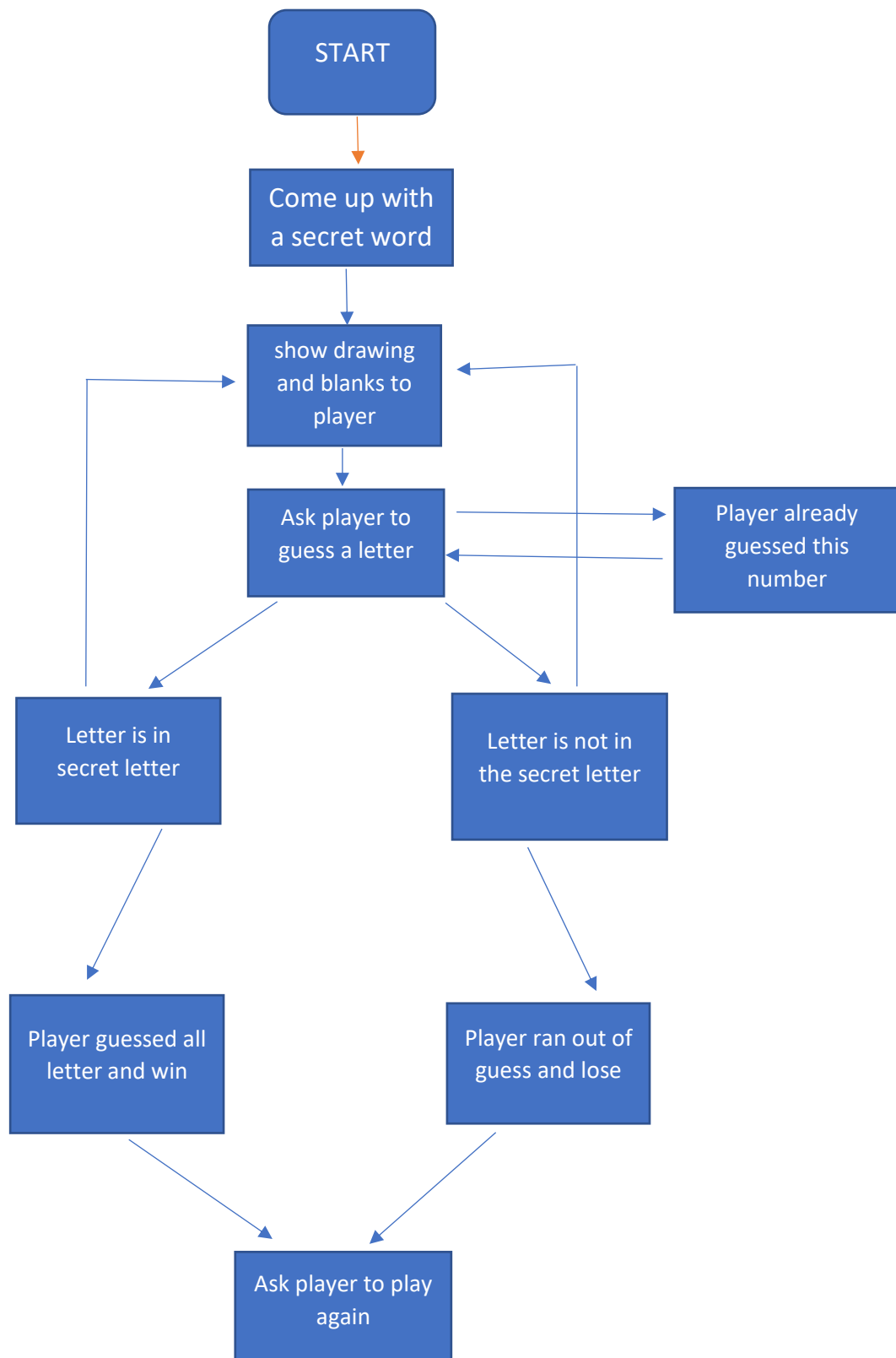
TOPIC :-Hangman game

DISCRIPTION

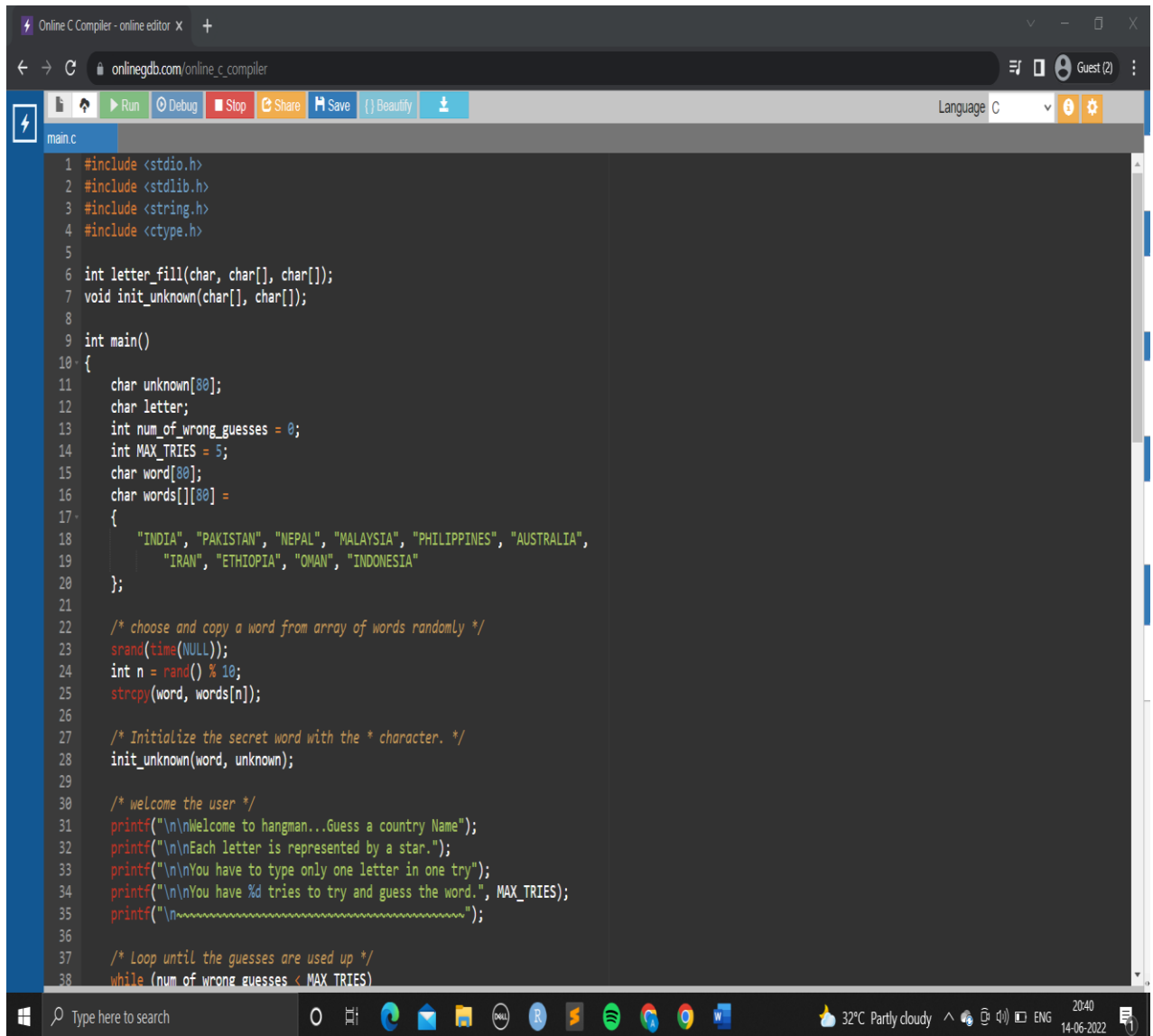
In the game of Hangman, the computer chooses a word at random from a given list of words. This word is the answer. The player then tries to guess the word, by guessing one letter at a time.

Whenever the user guesses a letter that is in the answer, all occurrences of that letter are revealed to the user. The game ends when the user has guessed every letter in the word, before he reaches the allowed number of strikes (usually 5). This program is an interactive Hangman game. The focus is to use and manipulate strings and loops.

METHODOLOGY



CODE:-



The screenshot displays a web browser window with the URL `onlinegdb.com/online_c_compiler`. The browser's address bar shows the user is logged in as 'Guest (2)'. The online compiler interface includes a toolbar with buttons for Run, Debug, Stop, Share, Save, and Beautify. The language is set to C. The code editor shows a file named `main.c` with the following C code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <ctype.h>
5
6 int letter_fill(char, char[], char[]);
7 void init_unknown(char[], char[]);
8
9 int main()
10 {
11     char unknown[80];
12     char letter;
13     int num_of_wrong_guesses = 0;
14     int MAX_TRIES = 5;
15     char word[80];
16     char words[][80] =
17     {
18         "INDIA", "PAKISTAN", "NEPAL", "MALAYSIA", "PHILIPPINES", "AUSTRALIA",
19         "IRAN", "ETHIOPIA", "OMAN", "INDONESIA"
20     };
21
22     /* choose and copy a word from array of words randomly */
23     srand(time(NULL));
24     int n = rand() % 10;
25     strcpy(word, words[n]);
26
27     /* Initialize the secret word with the * character. */
28     init_unknown(word, unknown);
29
30     /* welcome the user */
31     printf("\n\nWelcome to hangman...Guess a country Name");
32     printf("\n\nEach letter is represented by a star.");
33     printf("\n\nYou have to type only one letter in one try");
34     printf("\n\nYou have %d tries to try and guess the word.", MAX_TRIES);
35     printf("\n\n~~~~~");
36
37     /* Loop until the guesses are used up */
38     while (num_of_wrong_guesses < MAX_TRIES)
```

Online C Compiler - online editor X

onlinegdb.com/online_c_compiler

Run Debug Stop Share Save Beautify

Language C

main.c

```
40     printf("\n\n%s", unknown);
41     printf("\n\nGuess a letter: ");
42     scanf("%c", &letter);
43     letter = toupper(letter);
44     fflush(stdin);
45
46     /* Fill secret word with letter if the guess is correct,
47     otherwise increment the number of wrong guesses. */
48     if (letter_fill(letter, word, unknown) == 0)
49     {
50         printf("\nWhoops! That letter isn't in there!\n");
51         num_of_wrong_guesses++;
52     }
53     else
54     {
55         printf("\nYou found a letter! Isn't that exciting!\n");
56     }
57
58     /* Tell user how many guesses has left. */
59     printf("You have %d", MAX_TRIES - num_of_wrong_guesses);
60     printf(" guesses left.\n");
61
62     /* Check if they guessed the word. */
63     if (strcmp(word, unknown) == 0)
64     {
65         printf("%s\n", word);
66         printf("Yeah! You got it!");
67         break;
68     }
69 }
70 if (num_of_wrong_guesses == MAX_TRIES)
71 {
72     printf("\nSorry, you lose...you've been hanged.\n");
73     printf("The word was : %s\n", word);
74 }
75 getch();
76 return 0;
```

Type here to search

32°C Partly cloudy 20:40 14-06-2022

Online C Compiler - online editor × +

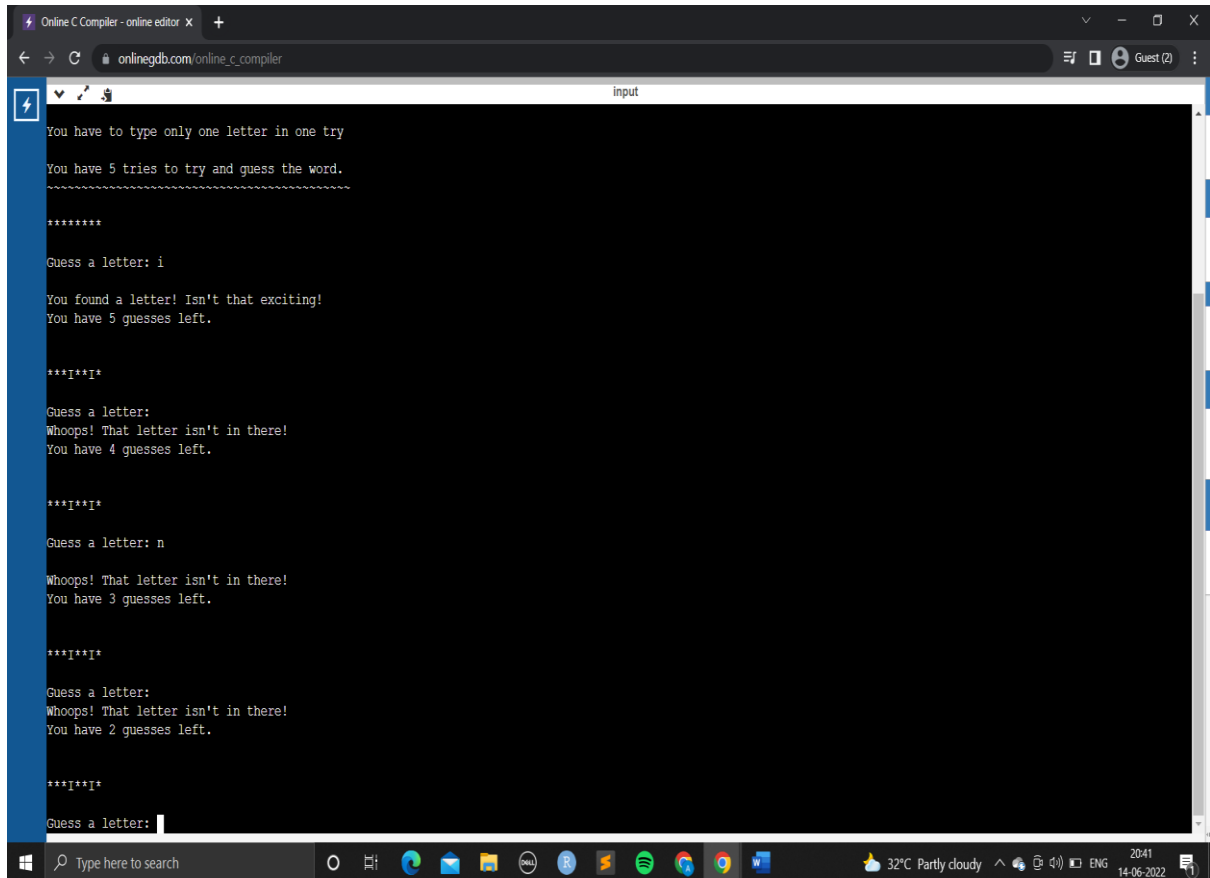
onlinegdb.com/online_c_compiler

Run Debug Stop Share Save {} Beautify

main.c

```
74 }
75 getch();
76 return 0;
77 }
78
79 /* Take a one character guess and the secret word, and fill in the
80 unfinished guessword. Returns number of characters matched.
81 Also, returns zero if the character is already guessed. */
82
83 int letter_fill(char guess, char secretword[], char guessword[])
84 {
85     int i;
86     int matches = 0;
87     for (i = 0; secretword[i] != '\0'; i++)
88     {
89         /* Did we already match this letter in a previous guess? */
90         if (guess == guessword[i])
91             return 0;
92         /* Is the guess in the secret word? */
93         if (guess == secretword[i])
94         {
95             guessword[i] = guess;
96             matches++;
97         }
98     }
99     return matches;
100 }
101
102
103 /* Initialize the unknown word */
104 void init_unknown(char word[], char unknown[])
105 {
106     int i;
107     int length = strlen(word);
108     for (i = 0; i < length; i++)
109         unknown[i] = '*';
110     unknown[i] = '\0';
111 }
```

RESULT :-



The screenshot shows a web browser window with the address bar displaying "onlinegdb.com/online_c_compiler". The browser tab is titled "Online C Compiler - online editor". The main content area is a dark-themed code editor with a light blue sidebar on the left. The code in the editor is a C program for a word-guessing game. The program prompts the user to type a letter, checks if it's correct, and updates the number of guesses left. The game has 5 tries. The user has guessed 'i', 'n', and 't' so far, all of which are incorrect. The program is currently waiting for the next guess.

```
input
You have to type only one letter in one try
You have 5 tries to try and guess the word.
~~~~~
*****
Guess a letter: i
You found a letter! Isn't that exciting!
You have 5 guesses left.

*****
Guess a letter:
Whoops! That letter isn't in there!
You have 4 guesses left.

*****
Guess a letter: n
Whoops! That letter isn't in there!
You have 3 guesses left.

*****
Guess a letter:
Whoops! That letter isn't in there!
You have 2 guesses left.

*****
Guess a letter: 
```

CONCULTION :-

**THE MISSION OF THIS PROJECT IS TO
PERMIT USER TO LEARN NEW WORD IN A
FUN MANNER**