

# **TYPING SPEED TESTER**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**  
**BANGLADESH UNIVERSITY OF BUSINESS & TECHNOLOGY**  
**(BUBT)**  
**MIRPUR-2. DHAKA**

**NOVEMBER, 2020**

# **TYPING SPEED TESTER**

*A project*

*Submitted to the Department of Computer Science and Engineering  
Bangladesh University of Business and Technology (BUBT), Dhaka  
In partial fulfillment of requirements  
For the Software Development-1  
of*

## **BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING**

**BY-**

**Md. Firose Mahmud (ID: 19201103003)  
Md. Arafat Hossen (ID: 19201103039)  
Md. Habibur Rahman Shohel (ID: 19201103027)  
Intake – 43<sup>rd</sup> (sec 1)**

**SUPERVISED BY-**

**Md. Saifur Rahman**

Assistant Professor,

Department of Computer Science and Engineering (CSE)

Bangladesh University of Business and Technology (BUBT)

Mirpur-2, Dhaka-1216, Bangladesh

## **ABSTRACT**

Nowadays computers are used for almost all purposes. Typing is the most important part of computer uses. All the instructions of the computer are given through typing and the speed of work depends on the speed of typing. So typing speed is required to speed up the work. Regular practice is needed to increase the speed of typing and that needs to be accurate. Improving typing practice and measuring accuracy at the same time is difficult and time-consuming. Our typing speed tester will speed up typing. Our goal is to create a platform for typing practice in a simple and accurate way. The user will be able to practice typing easily and happily here. At the end of each practice, the user will be able to see his score and if he wants, he can see his previous score too. For this, he will get the feeling of playing games and his typing speed will improve day by day.

## DECLARATION

We hereby declare that the project entitled “**TYPING SPEED TESTER**” submitted in partial fulfillment by us for the degree B.Sc. Engineering in Computer Science and Engineering in the faculty of Computer Science and Engineering of **Bangladesh University of Business and Technology (BUBT)** under the guidance of our supervision of **Md. Saifur Rahman**, Assistant professor, department of Computer Science and Engineering is our own work and it contains no material which has been accepted for the award to the candidates of any other disciplines except few references which is taken from various books and authors to enrich our knowledge about the topic of our project.

-----  
Md. Firose Mahmud

ID: 19201103003

Intake: 43

-----  
Md. Arafat Hossan

ID: 19201103039

Intake: 43

-----  
Md.Habibur Rahman Shohel

ID: 19201103027

Intake: 43

# CERTIFICATE

## TO WHOM IT MAY CONCERN

That is to certify that Md. Firose Mahmud, Md. Arafat Hossan and Md. Habibur Rahman Shohel students of B.Sc. in CSE has completed their project work titled **“Typing Speed Tester”** satisfactorily in partial fulfillment for the requirements of B.Sc. in Computer Science and Engineering from Bangladesh University of Business and Technology in the year November, 2020.

-----  
Md. Firose Mahmud

ID: 19201103003

Intake: 43

-----  
Md. Arafat Hossan

ID: 19201103039

Intake: 43

-----  
Md. Habibur Rahman Shohel

ID: 19201103027

Intake: 43

-----  
**Project Supervisor**

**Md. Saifur Rahman**

Assistant Professor

Department of CSE,

Bangladesh University of Business and Technology (BUBT).

Mirpur-2, Dhaka-1216.

## DEDICATION

*Dedicated to our parents for all their love and inspiration.*

## ACKNOWLEDGEMENT

**“Task successful”** makes everyone happy. But the happiness will be gold without glitter if we didn’t state the persons who have supported us to make it a success. Success will be crowned to people who made it a reality but the people whose constant guidance and encouragement made it possible will be crowned first on the eve of success.

This acknowledgment transcends the reality of formality when we would like to express deep gratitude and respect to all those people behind the screen who guided, inspired and helped me for the completion of our project work. We consider ourselves lucky enough to get such a good project. This project would add as an asset to our academic profile.

We express our gratitude to the help of our supervisor **Md. Saifur Rahman**, for his constant supervision, guidance and co-operation throughout the project and for giving constant motivation and valuable help through the project work. We also would like to thanks to our honorable chairman **Dr Kamruddin Md Nur**, for his support and giving us support and giving us permission to use the computer lab whenever we needed.

We extend our sincere gratitude to our parents who have encouraged us with their blessings to do this project successfully. Finally we would like to thank to all our friends, all the teaching and non-teaching staff members of the CSE Department, for all the timely help, ideas and encouragement which helped throughout in the completion of project.

## **APPROVAL**

This project “**Typing Speed Tester**” submitted by Md. Firsoe Mahmud, Md. Arafat Hossan and Md. Habibur Rahman Shohel ID No. 19201103003, 19201103039 and 19201103027 students of B.Sc. in Computer Science and Engineering from Bangladesh University of Business and Technology in the year November, 2020 under the supervision of Md. Saifur Rahman, Assistant Professor, Department of Computer Science and Engineering has been accepted as satisfactory for the partial fulfillment for the requirements of B.Sc. in Computer Science and Engineering and approved as to its style and contents.

-----  
**Project Supervisor**

**Md. Saifur Rahman**

Assistant Professor  
Department of CSE,  
Bangladesh University of Business and Technology (BUBT).  
Mirpur-2, Dhaka-1216.

-----  
**Chairman**

**Dr Kamruddin Md Nur**

Associate Professor and Chairman (Acting)  
Department of Computer Science and Engineering (CSE)  
Bangladesh University of Business and Technology (BUBT)  
Mirpur-2, Dhaka-1216, Bangladesh



## **COPYRIGHT**

© Copyright by Md. Firose Mahmud (19201103003), Md. Arafat Hossan (13142103103), and Md. Habibur Rahman Shohel (13142103094)

All Right Reserved

## ABBREVIATIONS

### Synonyms and Acronyms

LPM

RAM

HDD

### Descriptions

Letter Per Minutes

Random Access Memory

Hard Disk Drive

## LIST OF FIGURE

2.2.3 Flowchart Symble.....	11
3.4.1 Welcome Note.....	17
3.4.2 Short Instruction.....	17
3.4.3 Main Menu.....	18
3.4.4 Test Speed.....	18
3.4.5 Achievement.....	19
3.4.6 User History.....	19
3.4.7 Help.....	20
3.4.8 Exit.....	20

# Table of Contents

	Page No
<b>Abstract.....</b>	<b>i</b>
<b>Declaration.....</b>	<b>ii</b>
<b>Certificate.....</b>	<b>iii</b>
<b>Dedication.....</b>	<b>iv</b>
<b>Acknowledgement .....</b>	<b>v</b>
<b>Approval.....</b>	<b>vi</b>
<b>Copyright.....</b>	<b>vii</b>
<b>Abbreviations.....</b>	<b>viii</b>
<b>Chapter-1: Introduction.....</b>	<b>01-04</b>
1.1 Introduction.....	01
1.2 Problem Statement.....	01
1.3 Motivation.....	02
1.4 Objectives.....	03
1.5 Contributions.....	03
1.6 Organization of Project Report.....	03
1.7 Conclusions.....	04
<b>Chapter-2: Existing System.....</b>	<b>04-13</b>
2.1 Introduction.....	04
2.2 Existing System.....	05
2.3 Supporting Literature.....	07
2.3.1 Algorithm.....	07
2.3.2 Flow Chart.....	10
2.3.3.C Language .....	12
2.4 Conclusions.....	13
<b>Chapter-3: Proposed Model.....</b>	<b>13-21</b>
3.1 Introduction.....	13
3.2 Algorithm/Flow Chart of the Proposed System .....	14
3.3 SystemRequirements.....	16
3.3.1 Hardware Requirements.....	16

3.3.2 Software Requirements.....	16
3.4 User Interfaces.....	17
3.5 Conclusions.....	21
<b>Chapter-4: Conclusions and Future Plan.....</b>	<b>21-22</b>
4.1 Conclusions.....	21
4.2 Future Plan.....	23
<b>List of Figures.....</b>	<b>viii</b>
<b>References.....</b>	<b>23</b>
<b>Appendix.....</b>	<b>23-30</b>

# Chapter – 1 (Introduction)

## 1.1 Introduction:

As it is important to get any computer-related work done accurately and quickly, So as to accurate and fast typing is equally important. Typing requires regular practice to speed up and the practice platform should be simple and enjoyable. By the help of this '**Typing Speed Tester**', we can practice regularly easily, and accurately.

## 1.2 Problem Statement:

We developed a software for beginner who want to improve their typing speed. When an user open the software for practice himself/herself to check his/her typing speed firstly, the user got a welcome message after pressing any key then got a short instruction about the software. Besides the user have to input his/her name. Secondly, the user got the list of main menu. In the menu list it has four options which are

- 1. Speed Test**
- 2. User Record**
- 3. Help**
- 4. Exit.**

If the user choose “**Speed Test**” he/she have to enter **1**. Then the user will get some letters for testing. After finishing input the user can see his/her achievement which are percentage of correct, LPM and total score.

If the user wants, he can check the typing speed again or go to the main menu. If the user choose “**User Record**” he/she have to enter **2**. And will find all users record here. Form main menu, when the user forget the instruction about the software he can check it again using press **3**. Finally, if the user want to exit from the software he/she have to enter **4**.

With this software the speed of typing can be determined by letters. Through typing practice in this software, the user will be able to increase his/her typing speed.

### **1.3 Motivation:**

We were looking for a project to select that didn't match the other groups. The “**Typing Speed Tester**” got the idea while searching for the project in internet. There we saw a lot of complete projects but they were in another high level language. There were no such projects using C language. And then we decided that we would complete this project using C language.

## **1.4 Objectives:**

By the end of this project, the basics in C language have become clearer. The mentality of developing software and working in a team has been created. By completing the project we learned some of the functions of C language that were unknown to us. For example Time function, rand function obviously srand function.

## **1.5 Contributions:**

Our typing speed tester will speed up typing. Our goal is to create a platform for typing practice in a simple and accurate way. The user will be able to practice typing easily and happily here. At the end of each practice, the user will be able to see his score and if he wants, he can see his previous score too. For this, he will get the feeling of playing games and his typing speed will improve day by day.

## **1.6 Organization of Project Report:**

In this report, “**Abstract**” discussed page no “**i**”, “**Declaration**” discussed page no “**ii**”, “**Certificate**” discussed page no “**iii**”, “**Dedication**” discussed page no “**iv**”, “**Acknowledgement**” discussed page no “**v**”, “**Approval**” discussed page no “**vi**”, “**Copyright**” discussed page no “**vii**”, “**Abbreviations**” discussed page no “**viii**”.

Besides it divided Four Chapters. In page no **01 – 04** discussed about **Introduction**, in page no **03 - 04** discussed about **Existing System**, in page no **04 - 07** discussed about **Proposed Model** and in page **09 – 10** discussed about **Conclusions and Future Plan**. Also added some figure in page no **12 - 12** and appendix in page no **13 - 10**.

## **1.7 Conclusion:**

Typing is very important for computer-related work. We created this software for typing practice. With this software, the user will be able to see the improvement and accuracy of his typing. We have learned something new while creating this software.

## **Chapter – 2 (Existing System)**

### **2.1 Introduction:**

In this chapter we discussed the construction of baseline models of existing systems. This activity relies on knowledge of the basic C programming, algorithm and flowchart associated with the system under study.



## **2.2 Existing System:**

We have found a similar project like our project which is available in internet. That project's name is "**Typing Tutor**". Here is the abstract of that project under bellow-

### **Typing Tutor Project Abstract:**

#### **Functions Used:**

The major user defined functions used in the source code of this project are:

- **void loadFiles(void):** This function has been used to load .DAT files from the computer which has been stored with the .c file of project. The .DAT files contain the statistics, name of some users etc. of Typing Tutor Project in C. You can change them by editing the .DAT files and the source code of the project.
- **int mainmenu(void):** This user defined function has been used to print the main menu of the project.
- **int listUser(void):** There are certain predefined user in the project. This function lists name of them which are stored in ulist.DAT file.
- **int userSelectMenu(void):** It is for selecting the name of users which are already defined in the source code.
- **void createUser(void):** If you don't like the provided name of user, you can create a new user name too. This task is done using this user defined function.

- **int listLesson(void):** In Typing Tutor Project in C, this function has been used to list the name of lessons for testing the typing speed or learning the typing.
- **void beginSession():** This function starts the session or lesson which has been selected by the user.

`void viewRecords(void)`, `void about(void)`, `void sortSession(session list[],int n)`, `void addRecord(session cstr, void viewStat(void)`, `void DrawMenu(int opt,char *mi[],int n)`, `void box(int x,int y,int width,int height,int fc,int bc)` are some other user defined function in Typing Tutor Project in C. In order to make the source code user friendly, the name of the function simply indicates its use in the project.

## Features:

- This project provides certain name as user by default and there a facility to create a new user too. You can enter your name as user.
- In the project, you can view the total statistics.
- The record option in the project is for displaying the data of user which has already used the application. you can view typing speed, accuracy etc.
- If you want to terminate the program without any use, you can select exit option in the main menu.

## **2.3 Supporting Literature:**

As computing becomes a mainstream discipline embedded in the school curriculum and acts as an enabler for an increasing range of academic disciplines in higher education, the literature on introductory programming is growing. Although there have been several reviews that focus on specific aspects of introductory programming, there has been no broad overview of the literature exploring recent trends across the breadth of introductory programming. This paper is the report of an ITi CSE working group that conducted a systematic review in order to gain an overview of the introductory programming literature. Partitioning the literature into papers addressing the student, teaching, the curriculum, and assessment, we explore trends, highlight advances in knowledge over the past 15 years, and indicate possible directions for future research.

### **2.3.1 Algorithm:**

Algorithm is a step-by-step procedure, which defines a set of instructions to be executed in a certain order to get the desired output. Algorithms are generally created independent of underlying languages, i.e. an algorithm can be implemented in more than one programming language.

From the data structure point of view, following are some important categories of algorithms –

**Search** – Algorithm to search an item in a data structure.

**Sort** – Algorithm to sort items in a certain order.

**Insert** – Algorithm to insert item in a data structure.

**Update** – Algorithm to update an existing item in a data structure.

**Delete** – Algorithm to delete an existing item from a data structure.

## Characteristics of an Algorithm

Not all procedures can be called an algorithm. An algorithm should have the following characteristics –

- **Unambiguous** – Algorithm should be clear and unambiguous. Each of its steps (or phases), and their inputs/outputs should be clear and must lead to only one meaning.
- **Input** – An algorithm should have 0 or more well-defined inputs.
- **Output** – An algorithm should have 1 or more well-defined outputs, and should match the desired output.
- **Finiteness** – Algorithms must terminate after a finite number of steps.
- **Feasibility** – Should be feasible with the available resources.
- **Independent** – An algorithm should have step-by-step directions, which should be independent of any programming code.

## How to Write an Algorithm?

There are no well-defined standards for writing algorithms. Rather, it is problem and resource dependent. Algorithms are never written to support a particular programming code.

As we know that all programming languages share basic code constructs like loops (do, for, while), flow-control (if-else), etc. These common constructs can be used to write an algorithm.

We write algorithms in a step-by-step manner, but it is not always the case. Algorithm writing is a process and is executed after the problem domain is well-defined. That is, we should know the problem domain, for which we are designing a solution.

### Example

Let's try to learn algorithm-writing by using an example.

Problem – Design an algorithm to add two numbers and display the result.

Step 1 – START

Step 2 – declare three integers a, b & c

Step 3 – define values of a & b

Step 4 – add values of a & b

Step 5 – store output of step 4 to c

Step 6 – print c

Step 7 – STOP

### **2.3.2 Flowchart:**

#### **Overview**

A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

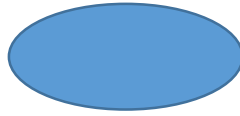
#### **Discussion**

Flowcharts display the steps in code as shapes connected together with arrows. The main goal is to create a rough draft of a solution to a coding problem. The type of shapes seen in the flowchart depends on what statements the programmer wants to create. For example, an “if” statement (a part of the code that only works if a certain condition is true) is represented by a diamond, while the looping statements (ones that allow a section of code to repeat itself as needed) are represented by hexagons. Flowcharts may also color code the different types of statements as well, making the code easier to read.

Common flowcharting symbols and examples follow. When first reading this section, focus on the simple symbols and examples. Return to this section in later chapters to review the advanced symbols and examples.

## Simple Flowcharting Symbols

**Terminal:** The rounded rectangles, or terminal points, indicate the flowchart's starting and ending points.



**Flow Lines:** To save time arrowheads are often only drawn when the flow lines go contrary the normal.



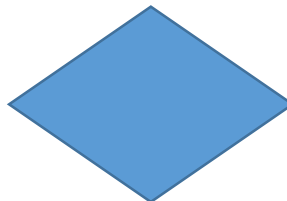
**Input/Output:** The parallelograms designate input or output operations.



**Process:** The rectangle depicts a process such as a mathematical computation, or a variable assignment.



**Decision:** The diamond is used to represent the true/false statement being tested in a decision symbol.



### **2.3.3 C Programming:**

The C programming language is a computer programming language that was developed to do system programming for the operating system UNIX and is an imperative programming language. C was developed in the early 1970s by Ken Thompson and Dennis Ritchie at Bell Labs. It is a procedural language, which means that people can write their programs as a series of step-by-step instructions. C is a compiled language.

Because the ideas behind C are kept close to the design of the computer, the compiler (program builder) can generate machine code/native code for the computer. Programs built in machine code are very fast. This makes C a good language for writing operating systems. Many operating systems, including Linux and UNIX, are programmed using this language. The language itself has very few keywords, and most things are done using libraries, which are collections of code for them to be reused.

C is available for many different types of computers. This is why C is called a "portable" language. A program that is written in C and that respects certain limitations can be compiled for many different platforms.

The syntax of C has also influenced many other programming languages, such as C++, C#, and Java, and many more programming languages we use nowadays.



Here is an example of a program written in C. When built and run it will show "Hello world!", followed by a new line on the computer screen.

```
#include <stdio.h>

int main() {
    printf("Hello world!\n");
    return 0;
}
```

## **2.4 Conclusion:**

Algorithm or Flowchart helps to create a program easily. It is step-wise representation of a solution to a given problem, which makes easy to understand. Every step in an algorithm has its own logical sequence. So it is easy to debug.

## **Chapter - 3 (Proposed Model)**

### **3.1 Introduction:**

This is a small project for beginner who are interested to increase the speed of typing. By using this software an user can see their achievement after complete every level. It shows the duration of time, percentage of right and give a score. The score stored in a file and when everyone want to see the total history who are the user of the software and their score. It helps the user to increasing speed of typing obviously. Because of it is a offline software, everyone used it easily.

## **3.2 Algorithm:**

Start

Step 1: Call Heading Function

Print welcome message

Call Instruction Function

Print sort instruction for project

Set user name

Step 2: Call mainMenu Function

Print Menu list

Input user choice

If : choice == 1, Then

Call Speed\_Test Function

Generate 10 Random Number

Initialize start with time Function

Initialize Ten string and Assign Some  
Letters

Print String For user to type

Declare three string for input

Input character by character

If: char != string[index], then

wrong++, (stored globally)

Initialize end with time Function

Assign duration by difftime function

(duration variable is declare Globally)

Call Calculation Function

Declare percent, right, ch, LPM

Assign ch by char '%'

Calculate; right = (total ltr - wrong)

Calculate; percent = (100.0\*right)/total ltr

If: 100>= percent >=90, then

Score=100; (stored Globally)

Else if: 90> percent >=80

Score = 90

Else if: 80> percent >=70

Score = 80

Else if: 70> percent >=60

Score = 70

Else if: 60> percent >=50

Score = 60

```

        Else if: 50> percent >=40
            Score = 50
        Else if: 40> percent >=30
            Score = 40
        Else if: 30> percent >=20
            Score = 30
        Else if: 20> percent >=10
            Score = 20
        Else if: 10> percent >=1
            Score = 10
        Else
            Score = 0
        Calculate; LPM = (total ltr * 60.0)/duration
        Clear Screen
        Print User name, percent, score, LPM
        Open File, append user_name and score
        Close File
        Input 1 or 2
            If: Input ==1
                Call Speed_Test Function again
            Else if: Input ==2
                Call mainMenu Function
    Else if: Choice == 2, then
        Clear Screen
        Declare variable a string and int
        Open the file by read mood
        Use fgets and store them into string
        Print them line by line
        Close the file
        Input 0
            If: Input == 0
                Call mainMenu Function
    Else if: Choice == 3, then
        Call instruction Function
        Input 0
            If: Input == 0
                Call mainMenu Function
    Else if: Choice == 4, then
        Clear the screen
        Print Ending Message

```

End

### **3.3 System Requirements:**

The requirements for the program to run or for the code to be in effect are basic and can be found on any simple computer. However some of the essential hardware and software requirements are described as follows:

#### **3.3.1 Hardware Requirements:**

The hardware required for the program to run is same as the hardware that can be found on simple personal computer (Desktop, Laptop, etc.). So there is no such special hardware requirement for the implementation of this project.

- Pentium IV or higher, (PIV-300GHz recommended)
- 256 MB RAM
- 1 Gb HDD space

#### **3.3.2 Software Requirements:**

As the C-programming language is used for the coding purpose, the main software required is also the same. The installation of this software is not much complicated and it does not occupy much space. The system requirement for the program to run is also basic and not hard to be found.

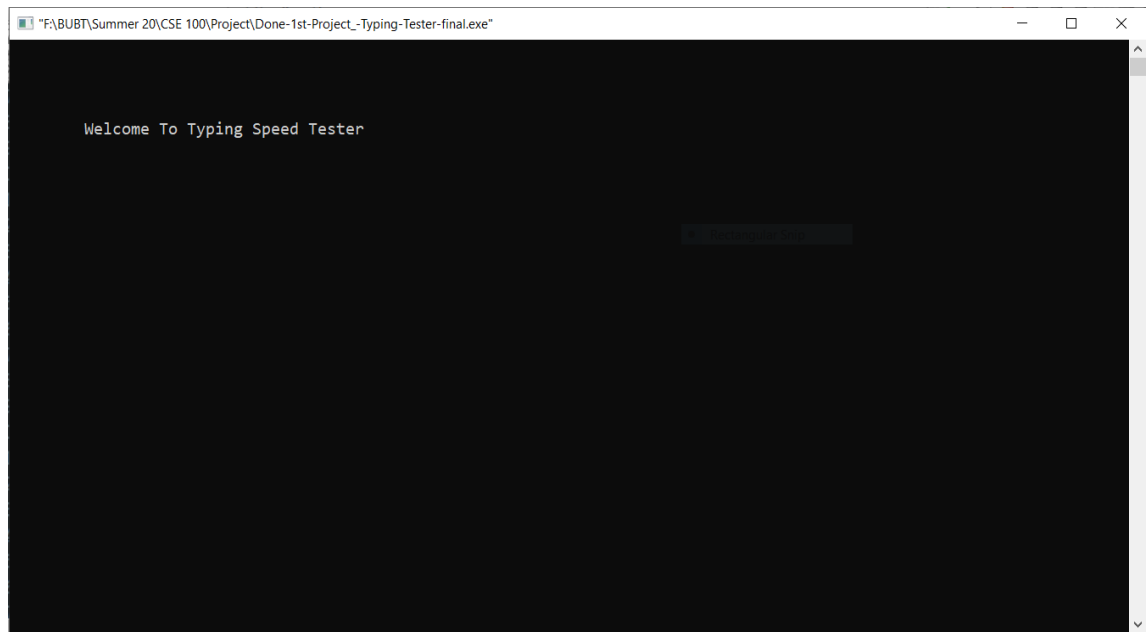
**Operating System:** Windows XP / Windows7/ higher

**IDE:** Codeblocks/ Visual Studio

**Compiler:** GCC

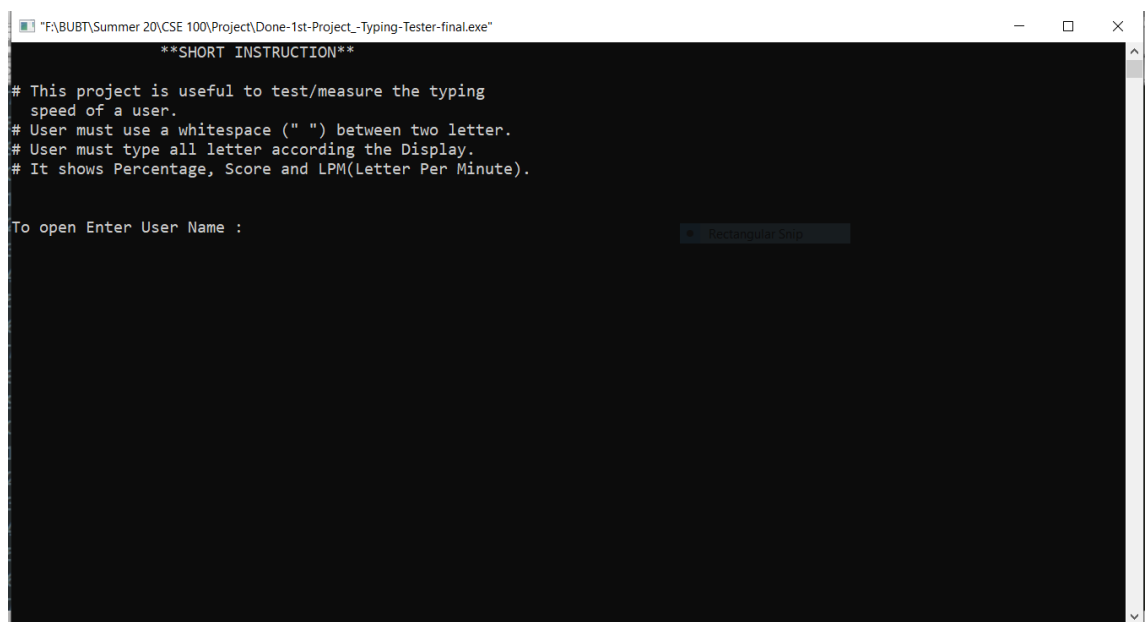
## 3.4 User Interfaces:

### 3.4.1 Welcome Note:



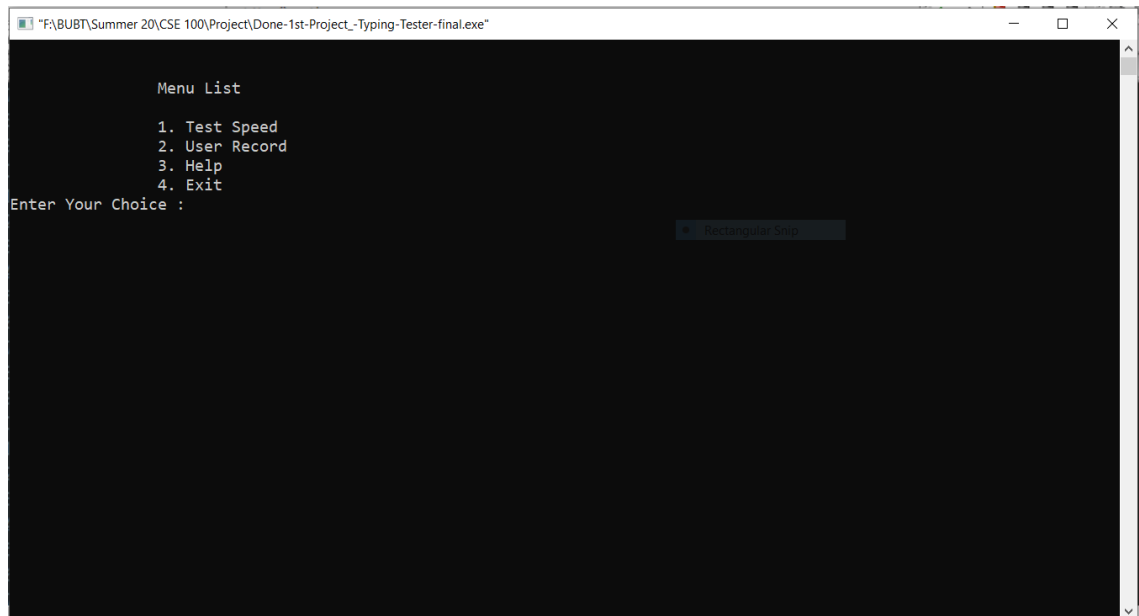
### 3.4.2 Short Instruction:

When Press any Key, the short instruction showed.



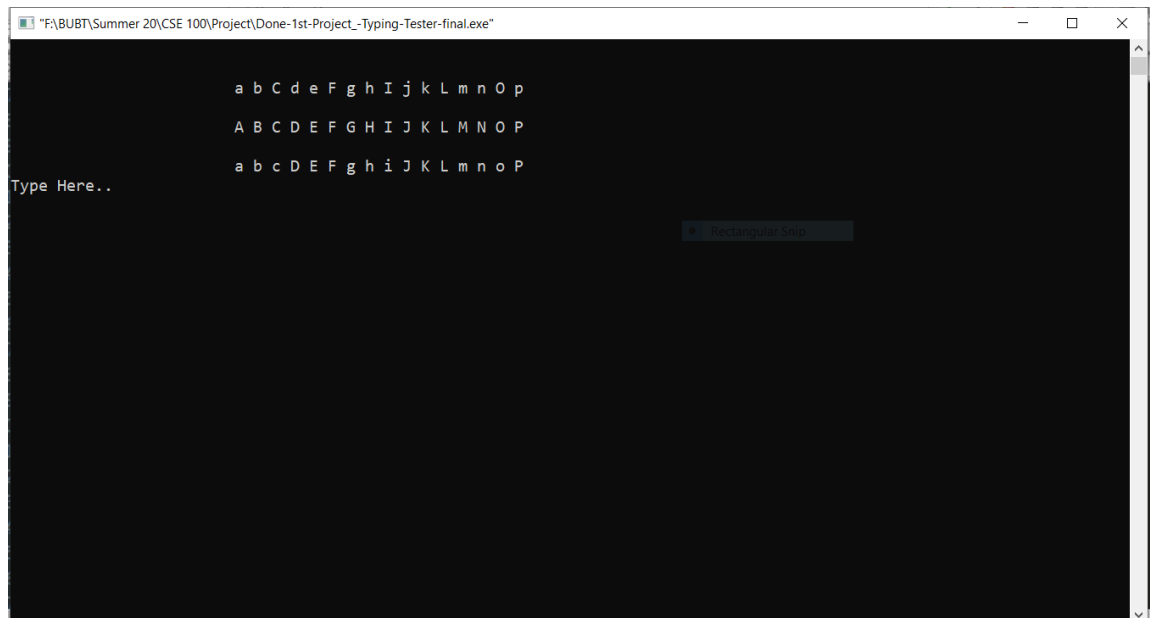
### 3.4.3 Main Menu:

After Enter the User Name Main Menu showed.



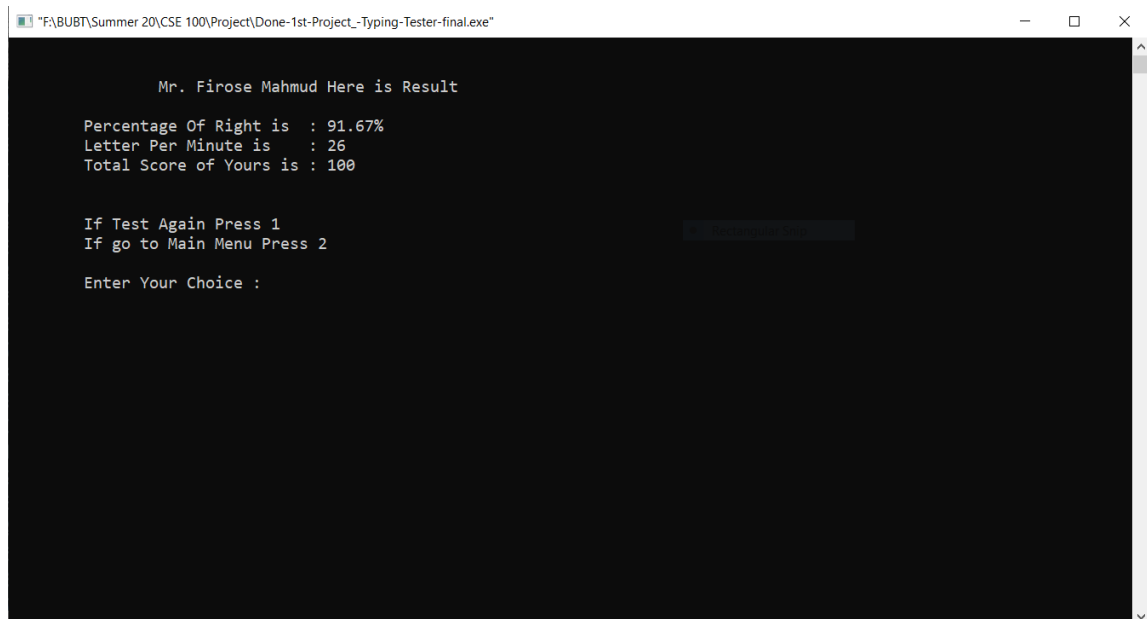
### 3.4.4 Test Speed:

When Enter the choice 1, then it showed-



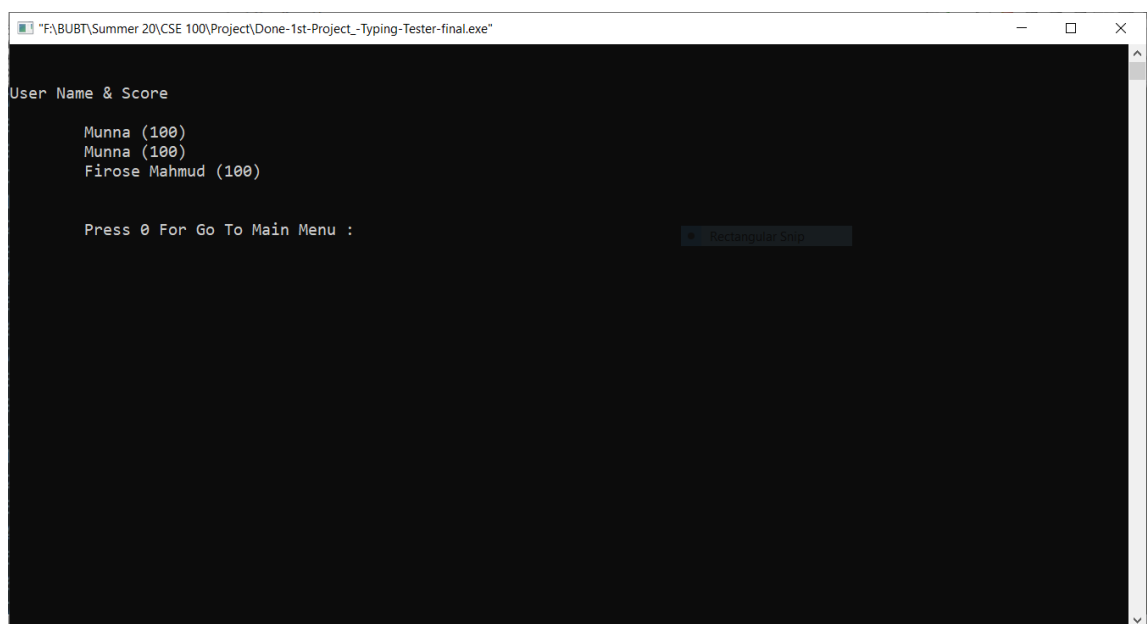
### 3.4.5 Achievement:

After finishing input, the achievement showed-



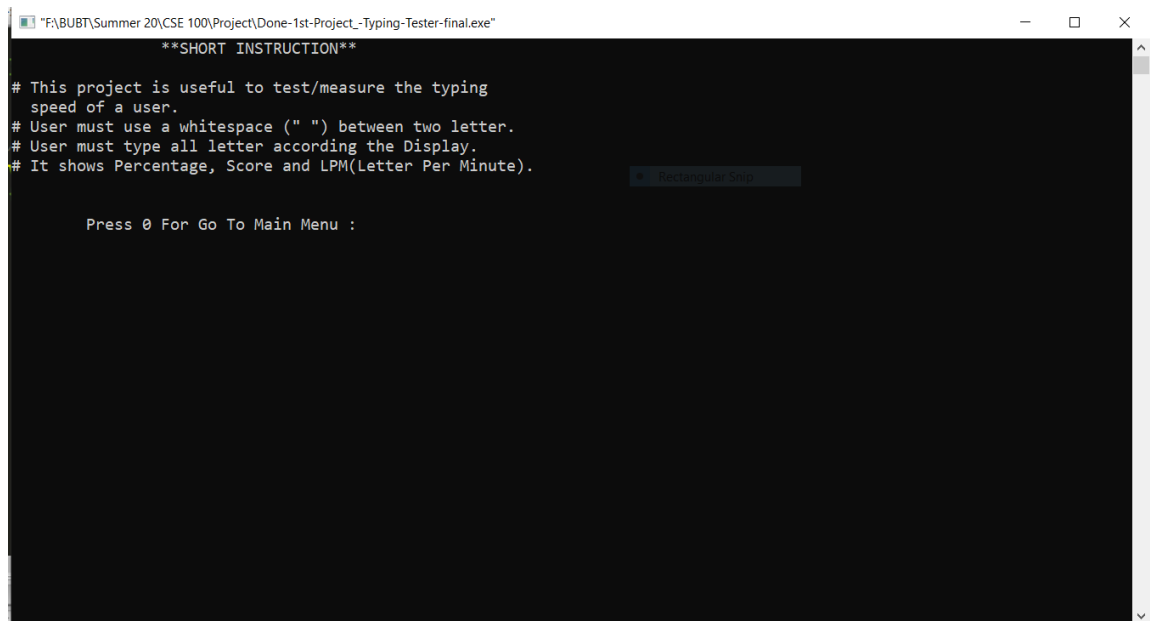
### 3.4.6 User History:

Input 2 for go to Main Menu again. And press 2 for show history all user.



### 3.4.7 Help:

If any forget the instruction he/she can press 3 from Main Menu then find the short instruction again-



```

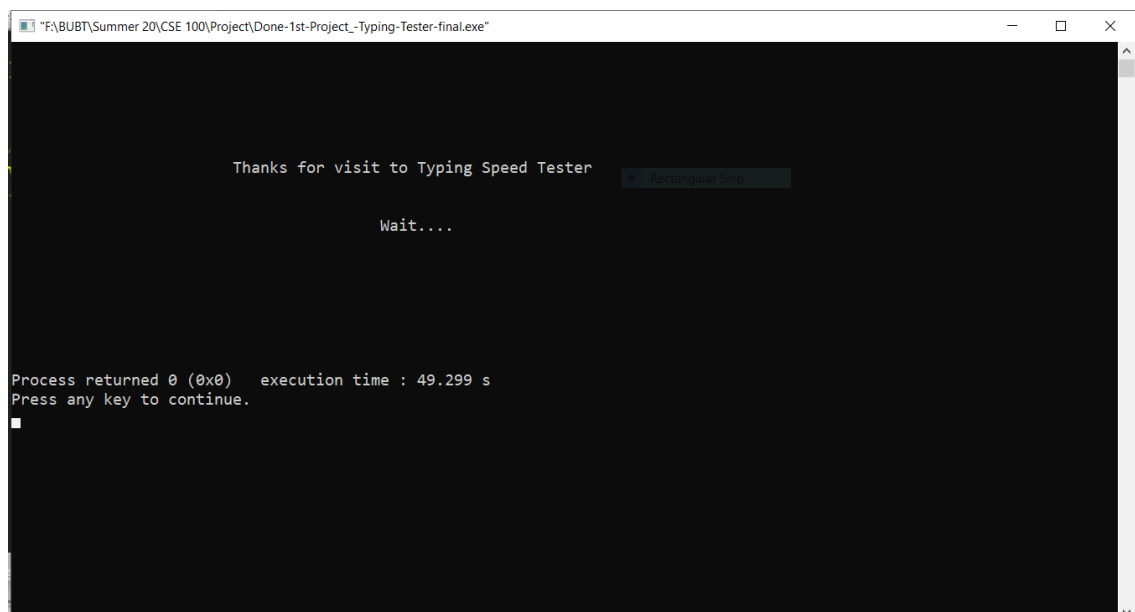
**SHORT INSTRUCTION**

# This project is useful to test/measure the typing
speed of a user.
# User must use a whitespace ( " ") between two letter.
# User must type all letter according the Display.
# It shows Percentage, Score and LPM(Letter Per Minute).

Press 0 For Go To Main Menu :
```

### 3.4.8 Exit:

Press 4 for Exit the software. Then it showed-



```

Thanks for visit to Typing Speed Tester

Wait....

Process returned 0 (0x0) execution time : 49.299 s
Press any key to continue.
█
```



### **3.5 Conclusion:**

Therefore we can conclude that the project has been successful for what it aims to perform. This program under the project can be used in individual user to increasing typing speed.

## **Chapter – 4 (Conclusion and Future Planning)**

### **4.1 Conclusion:**

We started this project when we had very few concepts about the C language and its different usage. During the project preparation we were introduced with the new concepts and new elements of C language along with the progression of the course. We were able to apply all those theoretical C concepts taught to us in our application to make it more and more functional. Over time, we've worked to make our project more compact and more user-friendly. During this period we learned various C programming concepts and now in position to design in the error free program successfully.

Overall the team feels that the project is pretty successful although we did face a few problems here and there but we came to know the various C programming techniques. We learned to make the variables and constants either global or local. Then we also learned the practical way of implementing the structures in the program and access the members of structures, compare them, initialize them and so on. The implementation

of techniques like string comparison, function calling, recursion, array storage, using file for storage, and calling by value or reference has broaden my knowledge regarding C language.

The major technical learning we had from the project is that how different functions can be associated to form the single program and how the data flows from one unit (function) to another and produce the desired output. We were introduced with the validations techniques to make the program error free and reliable. We had also learned to design the problem, develop the algorithms, and test the program to measure the efficiency and accuracy.

### **Limitations:**

Any project is not perfect. Every project has got some limitations. Our project has also has got some limitations. these limitations may be because it is a mini project and we are also not and expert. Some of the main limitations of the project or program are listed as follows-

- The display attraction of the program is poor. No interesting graphics are used as there is no use of <graphic.h> in the program.
- No use of date function.
- No option to create user account and details.

## **4.2 Future Planning:**

With the rise in technology use and of contemporary communication in media, typing fast is a skill we should all learn. More people are using computers for both work-related and personal activities, such as writing emails, reports and other documents, sending instant messages, and chatting on social media. More corporate, promotional, educational, and web-based materials are also being created. Plus, more audio and video recordings are being transcribed. This calls for the ability to type quickly and accurately.

So, our future plan about this project that is to make sure security to the every user of this app and we will store the user data like as, Individual id, name and highest score. To increase the typing skill we will add word and sentence in the database of our sample interface.

## **REFERENCES**

Algorithm Statement Took from here-

<https://www.efaculty.in/c-language/algorithm-in-c-language>

Flowchart Statement Took from here-

[https://en.wikibooks.org/wiki/Programming\\_Fundamentals/Flowcharts](https://en.wikibooks.org/wiki/Programming_Fundamentals/Flowcharts)

C Language Statement Took from here-

[https://simple.wikipedia.org/wiki/C\\_\(programming\\_language\)](https://simple.wikipedia.org/wiki/C_(programming_language))

## APPENDIX

```
//header file..
#include<stdio.h>
#include<string.h>
#include<time.h>
#include<conio.h>
#include<stdlib.h>

//file pointer..
FILE *fp;

//prototype of all function..
void heading();
void instruction();
void mainMenu();
void Speed_Test();
void calculation();

//globally declare
int duration, wrong=0, score;
char user_name[30];

//This is our main function...
int main()
{
    heading();
    mainMenu();

    return 0;
}

//define mainMenu function...
void mainMenu()
{
    system("cls");

    int choice;
    char c_message[] = {"Enter Your Choice : "};
```

```

printf("\n\n\t\tMenu List");
printf("\n\n\t\t1. Test Speed\n");
printf("\t\t2. User Record\n");
printf("\t\t3. Help\n");
printf("\t\t4. Exit\n");

for(int i=0; i<20; i++){
    printf("%c", c_message[i]);
}
//input choice..
fflush(stdin);
scanf("%d", &choice);

if(choice ==1 ){
    Speed_Test();
}
else if(choice == 2){
    system("cls");
    char uname[50];
    int c_choice;
    char footer[]={"Press 0 For Go To Main Menu : "};

    fp=fopen("score.txt","r");
    printf("\n\nUser Name & Score\n\n");
    while(fgets(uname,50,fp)!= NULL){
        printf("\t%s",uname);
    }
    fclose(fp);

    printf("\n\n\t");
    for(int i=0; i<30; i++){
        printf("%c", footer[i]);
    }
    scanf("%d", &c_choice);

    if(c_choice == 0)
        mainMenu();
}

```

```

else if(choice == 3){
    int c_choice;
    char footer[]={"Press 0 For Go To Main Menu : "};
    instruction();

    printf("\n\n\t");
    for(int i=0; i<30; i++){
        printf("%c", footer[i]);
    }
    scanf("%d", &c_choice);

    if(c_choice == 0)
        mainMenu();
}
else if(choice == 4){
    system("cls");
    char lastm[] = {"Thanks for visit to Typing Speed Tester"};
    printf("\n\n\n\n\n\n\n\t\t\t");
    for(int i=0; i<39; i++){
        printf("%c", lastm[i]);
    }
    printf("\n");

    char lastn[9] = {"Wait...."};
    printf("\n\n\t\t\t\t\t\t\t");
    for(int i=0; i<8; i++){
        printf("%c", lastn[i]);
    }
    printf("\n\n\n\n\n\n\n\n");
    for(int i=1; i<1e9; i++){
    }
    exit(0);
}
}
//Define heading function..
void heading()
{
    int i;
    char message[] = {"Welcome To Typing Speed Tester"};

```

```

printf("\n\n\n\n\t");
for(i=0; i<30; i++){
    printf("%c", message[i]);
}
getch();
instruction();
printf("\n\nTo open Enter User Name : ");
fflush(stdin);
gets(user_name);
}
//Define instruction function...
void instruction()
{
    system("cls");
    printf("\t\t**SHORT INSTRUCTION**\n\n");

    printf("# This project is useful to test/measure the typing\n");
    printf(" speed of a user.\n");
    printf("# User must use a whitespace (\n) between two
letter.\n");
    printf("# User must type all letter according the Display.\n");
    printf("# It shows Percentage, Score and LPM(Letter Per
Minute).\n");
}
//define sample function...
void Speed_Test()
{
    int chk[10];
    srand(time(0));
    for(int i=0; i<3; i++){
        chk[i] = rand()%10;
    }
    system("cls");

    time_t start, end;
    start = time(NULL);

    //INITIALIZE 3 LINE TO TEST..
    char line1[] = {"A B C D E F G H I J K L M N O P"};

```

```

char line2[] = {"A b C d E f G h I j K l M n O p"};
char line3[] = {"a b C d e F g h I j k L m n O p"};
char line4[] = {"a b c D E F g h i J K L m n o P"};
char line5[] = {"P o N m L k J i H g F e D c B a"};
char line6[] = {"P O N m L k J I H g F E D c B a"};
char line7[] = {"a B C d e g F h I j k l m P n O"};
char line8[] = {"h i J K L m a B C d a e D c B F"};
char line9[] = {"g h i d e h i J K L m L m n O p"};
char line10[] = {"m L k J i H g h I j k F g h I j"};

```

```
//SHOW THIS WHEN USER START TEST...
```

```
//Print 3 line according random number
```

```
printf("\nType Here..\n\n");
```

```
//INPUT FROM USER...
```

```
char testL1[32];
```

```
char testL2[32];
```

```
char testL3[32];
```

```
//1st line input from user...
```

```
// input as letter and compare with random line....if false then wrong++
```

```
end = time(NULL);
```

```
duration = difftime(end, start);
```

```
calculation();
```

```
}
```

```
//define calculation function....
```

```
void calculation()
```

```
{
```

```
double percent, right;
```

```
int LPM, a_choice;
```

```
char s_ch = '%';
```

```
right = 48-wrong;
```

```
percent = (100.0*right)/48;
```

```
// set score accordingly percentage..
```



```

        LPM = (48*60)/duration;
//printing users result...
        system("cls");
        char s_message[]={"Mr. "};
        char r_message[]={" Here is Result"};

        printf("\n\n\t");
        for(int i=0; i<4; i++){
            printf("%c", s_message[i]);
        }
        for(int i=0; i<strlen(user_name); i++){
            printf("%c", user_name[i]);
        }
        for(int i=0; i<15; i++){
            printf("%c", r_message[i]);
        }

        printf("\n\n\tPercentage Of Right is : %.2lf%c", percent, s_ch);
        printf("\n\tLetter Per Minute is : %d", LPM);
        printf("\n\tTotal Score of Yours is : %d\n", score);

        fp=fopen("score.txt","a");
        fprintf(fp, "%s (%d)\n", user_name,score);
        fclose(fp);

        char footer[]={"If Test Again Press 1If go to Main Menu Press 2Enter
Your Choice :"};
        printf("\n\n\t");
        for(int i=0; i<21; i++){
            printf("%c", footer[i]);
        }
        printf("\n\t");
        for(int i=21; i<47; i++){
            printf("%c", footer[i]);
        }
        printf("\n\n\t");
        for(int i=47; i<66; i++){
            printf("%c", footer[i]);
        }

```

```
scanf("%d", &a_choice);  
if(a_choice == 1){  
    wrong =0;  
    Speed_Test();  
}  
  
else if(a_choice ==2)  
    mainMenu();  
}
```