Project Title: WAR OF DIVISORS

Project Handler:

- ♣ **Khairul Anam Mubin** (ID: 17182103286)
- ♣ **Afsana Jahan Toma** (ID: 17182103291)

Instructor:
Hisham Siddique,
Lecturer, Dept. of CSE (BUBT)

Project Duration: 2 months.

Amount Requested: None.

Objective of The Project:

- To create a project (console game) using **C/C++** programming language and their features.

- To implement features like control statement, conditional statement, 2D array, loops statement, random function etc.

- To make the program easy while running it.

- To concise the memory of program as far as possible.

- To get an idea about making a simple project using **C and C++.**

- To be able to solve problems by Compiling and Debugging.

Project Goals:

- ➢ User friendly
- ➢ Reduced cost
- ➢ Less effort
- ➢ Easy maintenance

Project Implementation and Management Plan:

- **Expected project results**
  System will be easy to use, user friendly environment, very little maintenance cost, runnable in any supported device (High, medium and low end device).

- **Project activities and work plan**

Steps

I. Main Menu
   1. **Play multiplayer mode**
      a) **Play again**
      b) **Return to main menu**

   2. **About data**
      a) **Show About game**
         A. **Return to main menu**
      b) **Show About developers**
         A. **Return to main menu**
   3. **Exit**

Project Monitoring:

Project will be monitored to ensure that activities occur as planned. Appropriate corrective action will be taken if required.

Hard Ware Requirement:

   ✓ Intel Processor core i3 3rd gen

Soft Ware Requirement:

   ✓ Windows 10
   ✓ Code blocks

Game Rules:

There are two mode:

   1. Single Player

   2. Double Player

For single player:

   1 Run the game & press 2 for playing in single mode.

   2 Choose the game level. 1 for easy 2 for medium and 3 for hard.

   3 By choose a number from 3 to 9 to start game and going through a toss.

4   Who win the toss the removing is start for him. Thus he have to remove the multiple of that number only.

5   If opponent have no multiple left then the others win.

For Multiplayer:

1. Run the game & press 1 for playing in Multiplayer mode.

2. Insert the player 1 and player 2 name.

3. Chose numbers and after it go through a toss which is going to decide who will remove the first multiple.

4. If opponent have no multiple left others win.

*   After finishing game, you can choose to play again or exit.

*   Press 1 for play again

*   Press any key for exit.

Code and Functions Details:

There we have used a few numbers of **built in functions**.

char toupper(**char** c): which return the upper case of that character 'c' if the 'c' is in lowercase.

srand(time(0)): This function work time base and set an algorithm which can generate random number .

rand() : This is the random function which generate the number.

system("cls");This function clear the console.

headers: #include <time.h> , #include <stdlib.h> , #include <conio.h>

These headers are much needed to use these above functions.

 We have used our own function to short the code, efficient and maintain time complexity. They are also readable and portable for other developers for future developing.

1.**void** description(); this function describes the description and the rules of the game.

2. **void** single_player(); This functions executes when the user select the single player of the game . Actually this stand between the user and the Computer.

3. **void** Multiplayer(); This is the multiplayer function. When its called in the code or program the multiplayer game executes.

4. **void** vsmode();This is the main function while all program executes in multiplayer battle.

5**. void** easy();

6**.void** medium();

7**.void** hard();

 5,6,7 These are the level function which is in only used in single player mode.

8. **bool** move_choice(); This function is only for easy and medium mode. its decide the move choice of the user and the computer.

9**.bool** move_choice_hard();Its in hard mode move choice for single player battle.

10**.bool** move_choice_mul();While playing multiplayer battle then this functions take the move choice.

11. **void** gen_grid(**in**t n, **int** x);This is the grid generator function. This function is used for generate the numbers in grid/ matrix.

12.**void** show_grid(**int** n); every time this function called it shows the grid in the console display.

13. **bool** toss();

14.**bool** toss_mul();

13,14 these two functions randomly choose who is going to remove the first multiple from the grid.

15**.bool** rmv_pl1(**int** n);

16**.bool** rmv_pl2(**int** n);

17.**void** rmv_easy_pc(**int** n);

18.**void** rmv_hard_pc(**int** n);

 15,16,17,18 these functions are used for remove the multiple of their given number. 18, void rmv_hard_pc(int n) this function works with some logical algorithm . That's why we named it rmv_hard_pc. If there is a number which is the multiple of the computers choosen number and its the also the multiple of users choosen number then it will remove by computer .As its the main tricks of this game. For this, The computer play or remove the number optimally from the grid.

19.**int** win(**int** n,**int** dd); it count the number of divisors left in computers and users side. If the counting result is zero for anyone then the others win.This function declares the winning.

20.**void** sieve();This function generates the prime numbers which is help to remove the prime number from the grid.

Appendix:

As a whole, the project will be a good learning experience for us. We will be gained knowledge about the various aspects of C and C++. At the same time, we will be developed a deep understanding about the application developing interface.