REPORT

GROUP MEMBERS:

- *Utku Cavit Dener – 19332652043*
- *Canan Dursun – 19332652016*

Firstly we made all the article together. We mostly met from the discord application and made decisions about how we can do it first. We made a distribution of tasks between us for each article. And then we checked each other's work and we told each other about our work from the Discord application. We got help from each other. In this way, we finished in a short time.

1. *We compared the two versions.*
   - *Version 1 has more than 25 control statements but Version 2 has almost 10 control statements. In version two, there are less control statements comparing to version one. Because it's more readable and has high-performance.*
   - *Version 1 has 1 void and 1 int but Version 2 has 2 void and 2 int. In version two, the amount of function number is more than comparing to version one. That provides us to control structure of program and also provides to other programmers to read easily.*
   - *Version 1 used system ("cls") function get cleaner console view. In Version 2, the current board is drawn after each move so each move can be seen when and where. So Version 2 does not have such a thing.*
   - *In Version 2 has pointers and that provides flexibility for using. Version 1 does not have pointer.*
   - *In Version 2, a square can be moved infinitely, and when an invalid move is entered, the turn moves to the other player. In Version 1, a single move can be made to a square, and when an invalid move is entered, the same player is asked for a valid move. Version 2 can enter endless loop. Version 1 is better than Version 2 because the rules in the original game are the same as in Version 1.*
   - *In both versions, you can view the available squares on the game board. Version 2 tells available squares with each move, but can make infinite moves to unavailable squares. So Version 2 is unnecessary and that's why Version 1 is better than Version 2.*
2. *In the versions folders, you can find the pseudocodes for both of the versions.*

3. *‘conio.h’ is a header file mostly served by MS-DOS compilers for console input / output processing. It is neither mentioned in the C Programming Language book, nor part of the C standard library or ANSI C, nor is it defined in POSIX. Most C compilers tahta target UNIX and Linux do not have this header and do not supply the library functions. Some embedded systems or cc65 use a conio-compatible library.*

- *Some functions from 'conio.h' library used in Version 1:*
  kbhit: kbhit() is present in ‘conio.h’ and used to determine if a key has been pressed or not. If a key has been pressed then it returns a non zero value otherwise returns zero.
  getch: getch() also reads a single character from the keyboard. But it does not use any buffer, so the entered character is immediately returned without waiting for the enter key. We found out just getch() function. And we can use scanf() function instead of getch() function.

4. *a)*

- *In Version 1,we created 2 source files and 2 header files. Their names header_1 and board_header. Checkwin function is in header_1. We explained a string in that funtion. Board funciton is in board_header. We explained same string in that function, too.*
- *In version 2,we created 2 source files and 2 header files. Their names check_win and void_func. There are 4 function in them. CheckForWinner function is in check_win. We explained a string, a counter and pointer in that function. drawGrid, takeTurn, swapPlayer functions are in void_func. We explained a string in drawGrid function. We explained a string and a pointer in takeTurn function. We explained a pointer in swapPlayer function.*

b) We write on the loops and some functions.When you look codes on codeblocks.You can see.

c) You can see in codeblocks.

5. *a)We use getche function to take choice.And swapPlayer and takeTurn functions improved.*

b) Generally, we added the option to play against the computer and set a range for the rand () function and let the computer randomly play in that range. You can see in codeblocks.

6. *As known firstly ,we had to decscribe file.With "if else" sturcture we checked file is null or not.And than we opened file with fopen funtion and took grid from file to start where it left off.In while loop we open file again to write ,printed grid array to record what we played and closed the file.At the End of the main function,we opened file to write again ,printed grid to return to go back to beginning and closed the file with fopen.*

7. *We chose the ' Modify Version 2 so that a user cannot over-write previous moves' option. Generally we provided this with the do-while loop. You can see in codeblocks.*