CS161 Lectures 14, 15

Check
Select location
Select first player
Rules

Main

Local

Local

Local

Local

Local

Local

Local
main

Assume 3 rows setup values 1→9

Rules

Drawboard

Select First player
X, O value

Select Location
Location 1-9

Check to see if Avail
T/F

Avail?

NO (FALSE)

YES (TRUE)
1. Location, player (x, 0)
2. Play position
3. Drawboard
4. Winner? (no)
5. Message
6. Catches
7. Say to player
8. Switch player
9. NO
10. YES
Select Location

x,0

Prompt 1-9

Read

\#

1\rightarrow9? NO

yes

\#

Return \#
Check row

Row

if x \neq 0?

\text{if row, index } \equiv \text{ # index } \div 0, 1, 2

\text{if } x \neq 0? \text{? NO TRUE}

\text{yes} \rightarrow \text{FALSE (DON'T ALLOW!)}
Play position

Location player (x, 0)

1 → 3
- yes: row 1
  - store x, 0 in the array
- no:
  - 4 → 6
    - yes: row 2
      - store x, 0
    - no: row 3
      - store x, 0
else {
  player = X,
}

if (player == X) {
  void switchplayer(char &player)

player = O;
player = O;

x, y = player;
Alternate Main

→ Rules

→ Select first player

play game

→ Stop?

- NO
  - Switch play

- YES
  - play game
void play_game (char player) {
    if (winner (player) && winner (opponent) && cornerstone(c, player))
        switch (player (player));
    else
        do
            player = first (player);
            welcome (player);
            char player;
        until (main (c, player));
}
Play game

\[
\text{player} \downarrow \\
\text{draw board} \\
\downarrow \\
\text{Select Location} \\
\downarrow \\
1-9 \\
\downarrow \\
\text{Check to see if AVAL} \\
\downarrow \\
T/F \\
\downarrow \\
\text{AVAL} \\
\downarrow \\
\text{play position} \\
\]
```java
{ void play_game (char player) {
    int loc = select_loc (player); // 1-9
    do {
        if (board[loc] != '.') {
            loc = select_loc (player);
        } else {
            board[loc] = player;
            if (check_row (loc)) {
                break;
            } else if (check_column (loc)) {
                break;
            } else if (check_diagonal (loc)) {
                break;
            }
        }
    } while (true);
}
```
"Dream main"

- Welcome
- Set first player
- Play game
- Do you want to switch users?
- Yes → Finished
- No → Switch user
bool done_yet(char player);

int main()
{
    char player;
    Welcome();
    player = firstplayer();
    do
    {
        play_game(player);
        switchplayer(player);
    } while (!done_yet(player));
    cin.get();
    return 0;
}
Other functions...

void switchPlayer(char &player)

// Switch from an X to an O

if (player == 'X')
{
    player = 'O';
} else
{
    player = 'X';
}

if (minimum == 8 & 1 constructor)
bool done_yet(char player)
{
  bool done = false;
  if (winner())
  {
    done = true;
    cout << "CONGRAT'S!
      player " << player
      << " won " << endl;
  }
  else if (catscratch())
  {
    done = true;
    cout << "Try again!
      again!" << endl;
  }
  return done;
}
`{ return false; }`  

`{ return true; }`  

`cost > "Try again"`  

`if (character) {`  

`... return true;`  

`... cost > "Congratulations!"`  

`if (winner) {`  

`bool done_yet (check player) Another way`
\[ \emptyset < \{ 1, 2 \} \leq 2 \] 
\[ \emptyset > \{ 1, 2 \} \leq 2 \]

Same content

\[ \emptyset = (\{ 1, 2 \}) \cap S \cap \emptyset \]
```c
strncpy (str1, str2, 10);
```

Instead, we call `strncpy` from the string library:

```
strncpy (str1, str2, STR2); // STR1 = STR2;
```

Instead of assigning the string `str2` to `str1` directly, we first use `strncpy` to copy the string from `str2` to `str1` up to the 10th character.

This is illegal because an array is really a constant pointer to the constant string.

```
str1 = str2;
```

The assignment operator (`=`) assigns the address of `str2` to `str1`, which is illegal because the string length is not guaranteed to be less than or equal to 10.

We also cannot copy strings using the assignment operator:

```
char str1[10], str2[10];
```

We cannot change the location address of the first element of the array. We cannot change the location address of the first element of the array.

COPYING ARRAYS OF CHARACTERS
Welcome

Read a word

the

The

Yes

Add 1

NO

'@'

NO

Display Result