CS5351 Software Engineering
2024/2025 Semester B
Exercise on Modern Code Review (MCR)

I. Background on MCR

1. MCR is a lightweight and tool-based code review of *code change* (not the whole piece of code!)
   - It is the norm for many software development projects
2. Characteristics of MCR
   - Informal;
   - Tool-based logistics;
   - Asynchronous;
   - Focused on reviewing code changes;
   - Lighter weight than code inspection.
3. Main Ideas
   - Author A makes a patch[1] P on a code block C to address some problems and sends the patch P via e-channel (e.g., email, Whatsapp, WeChat, or review tool) to a set of reviewers R
   - Each reviewer R evaluates P: either deem P good or reject P
   - Author A commits P; post-commit reviewer by other Rs possible.
4. Authors and reviewers exchange ideas, find bugs, and discuss alternative solutions to better design the structure of a submitted code change.
5. A case study and more information about MCR can be found in the following paper[2].

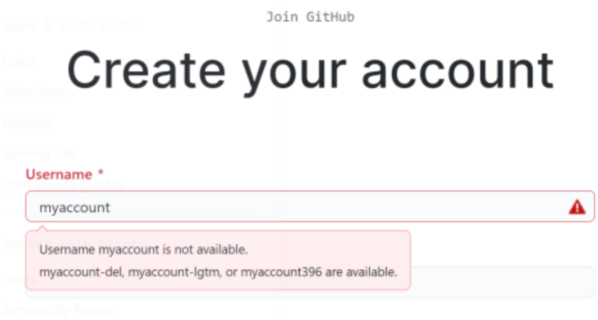
[1] Patch refers to the diff file of two pieces of code. See the example in the following link: https://www.cyberciti.biz/faq/appy-patch-file-using-patch-command/

[2] Sadowski, Caitlin, Emma Söderberg, Luke Church, Michal Sipko, and Alberto Bacchelli. "Modern code review: a case study at google." In Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice, pp. 181-190. 2018.

II. In-Class Activity MCR

1. **Each student has a role as follows:**
   - Author A of the code: 1 student
   - Reviewers R: no pre-defined role
   - Use GitHub for review, and Author A should do the following step **from 2 to 6 to** prepare for the MCR activity.

2. [**all students**] **Step to sign up a Github account if that student has an account yet**
Create a GitHub account (https://github.com) (If have, just login)



Join GitHub

Create your account

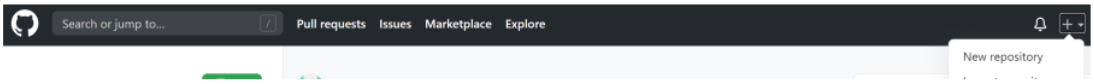
Username *

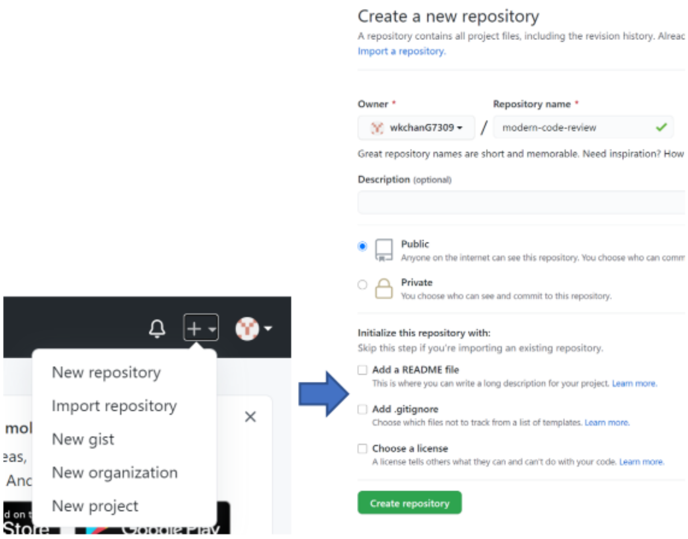
myaccount

Username myaccount is not available.
myaccount-del, myaccount-lgtm, or myaccount396 are available.

3. [**Author A**] **Step for one student of the team to create a project**

- Create a project for the exercise





Create a new repository

A repository contains all project files, including the revision history. Alreac
Import a repository.

Owner * wkchanG7309 / Repository name * modern-code-review ✓

Great repository names are short and memorable. Need inspiration? How

Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can comm

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

□ Add a README file
This is where you can write a long description for your project. Learn more.

□ Add .gitignore
Choose which files not to track from a list of templates. Learn more.

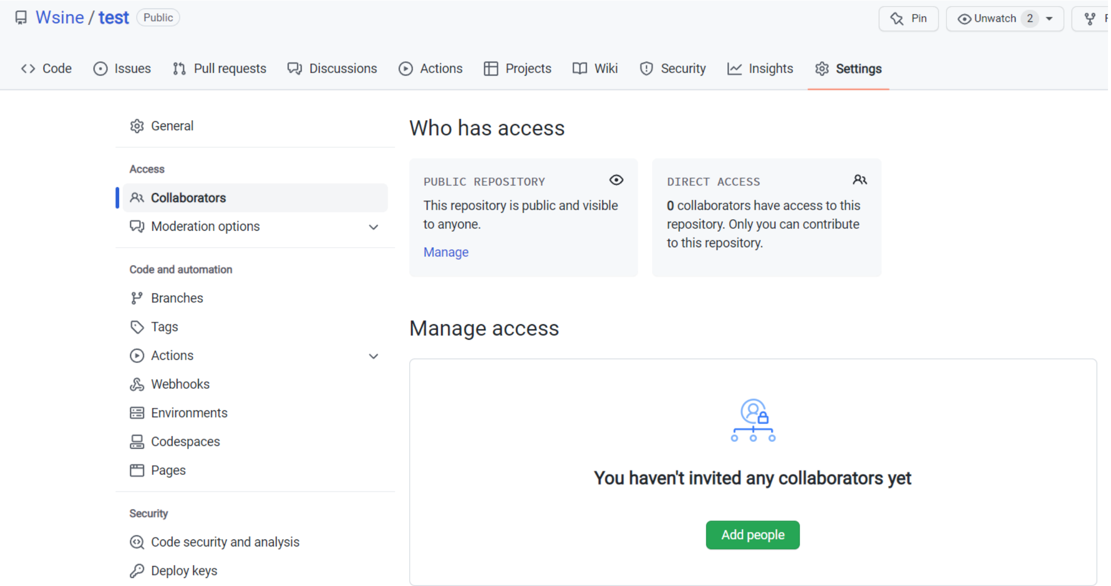
□ Choose a license
A license tells others what they can and can't do with your code. Learn more.

Create repository

4. **Step for Author A to invite other students to join the project**

- [**All but Author A**] **Inform Student A about your GitHub ID**
  - **Tell Student A set you as a collaborator.**

- [**Author A**] **Invite all your teammates as the project's collaborators.**
  a. Press [Settings]
  b. Select [Collaborators]
  c. Press [Add people]



Who has access

PUBLIC REPOSITORY

This repository is public and visible to anyone.

DIRECT ACCESS

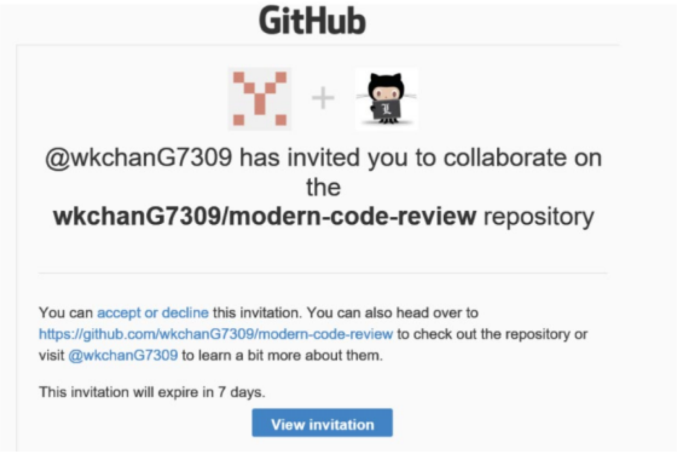
0 collaborators have access to this repository. Only you can contribute to this repository.

Manage access

You haven't invited any collaborators yet

5. [**All but Author A**] **Step for other students to check their mailboxes to join the project**

- Check your email and accept the collaboration invitation from Author A.



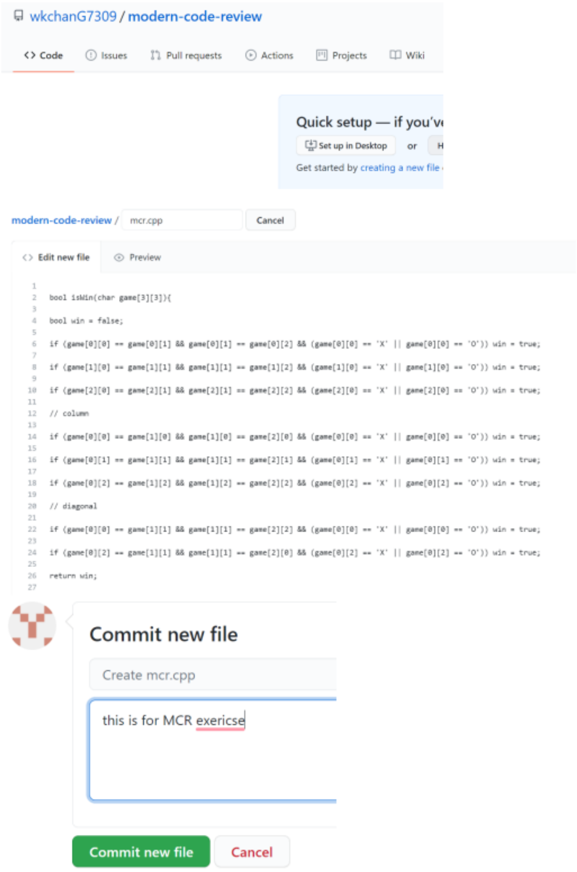
@wkchanG7309 has invited you to collaborate on the **wkchanG7309/modern-code-review** repository

You can accept or decline this invitation. You can also head over to https://github.com/wkchanG7309/modern-code-review to check out the repository or visit @wkchanG7309 to learn a bit more about them.

This invitation will expire in 7 days.

6. **Step to check in the original code**

- [**Author A**] **Create a new file in the GitHub project and check in the original code with the following steps.**
  - Go to [Code] menu
  - Press [creating a new file]
  - Name your file as mcr.py
  - Copy and paste the code listing from the Canvas/Code/mcr.py for Suthor A to the editor in GitHub.
  - Press [Commit new file]



```

bool isWin(char game[3][3]){

bool win = false;

if (game[0][0] == game[0][1] && game[0][1] == game[0][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;

if (game[1][0] == game[1][1] && game[1][1] == game[1][2] && (game[1][0] == 'X' || game[1][0] == 'O')) win = true;

if (game[2][0] == game[2][1] && game[2][1] == game[2][2] && (game[2][0] == 'X' || game[2][0] == 'O')) win = true;

// column

if (game[0][0] == game[1][0] && game[1][0] == game[2][0] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;

if (game[0][1] == game[1][1] && game[1][1] == game[2][1] && (game[0][1] == 'X' || game[0][1] == 'O')) win = true;

if (game[0][2] == game[1][2] && game[1][2] == game[2][2] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;

// diagonal

if (game[0][0] == game[1][1] && game[1][1] == game[2][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;

if (game[0][2] == game[1][1] && game[1][1] == game[2][0] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;

return win;

```

Commit new file

Create mcr.cpp

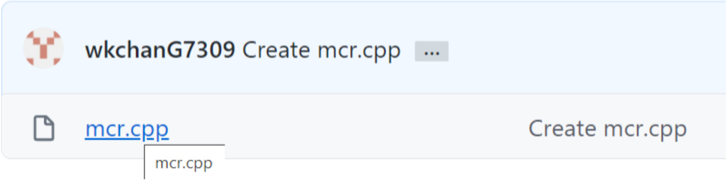
this is for MCR exericse

7. [00-10 minutes] **Author A prepares the patch P for the code C, and sends the patch P with some explanation to the e-channel**

- **Note: patch is the diff file of two pieces of code**

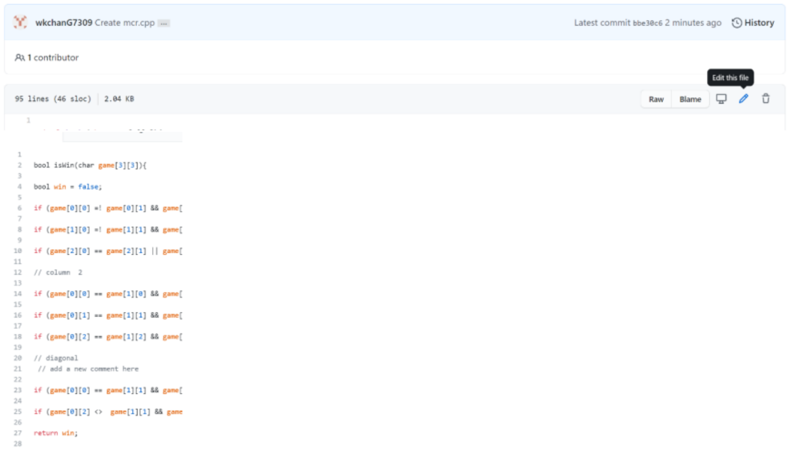
7.1 [**Author A**] Make changes and Check out the code

- Click the mcr.py file in GitHub.
- Press [edit file] according the reviewers' comments.
- Modify the mcr.py according to the change required.
- Go to the bottom of this page
- **Important**: Check [Create a **new branch** for this commit and start a pull request.] **in order to create a PATCH P**.
- Press [Propose changes]



wkchanG7309 Create mcr.cpp

mcr.cpp Create mcr.cpp



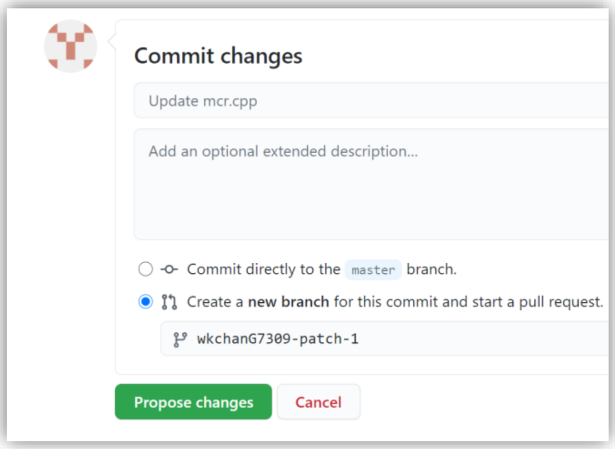
wkchanG7309 Create mcr.cpp

Latest commit bbe30c6 2 minutes ago

1 contributor

95 lines (46 sloc) | 2.04 KB

```

bool isWin(char game[3][3]){

bool win = false;

if (game[0][0] =! game[0][1] && game[

if (game[1][0] =! game[1][1] && game[

if (game[2][0] == game[2][1] || game[

// column  2

if (game[0][0] == game[1][0] && game[

if (game[0][1] == game[1][1] && game[

if (game[0][2] == game[1][2] && game[

// diagonal
 // add a new comment here

if (game[0][0] == game[1][1] && game[

if (game[0][2] <>  game[1][1] && game

return win;

```



Commit changes

Update mcr.cpp

Add an optional extended description...

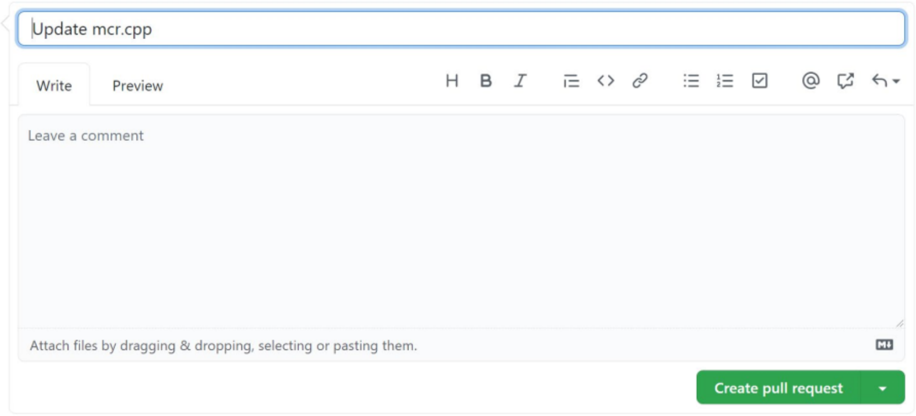
○ Commit directly to the `master` branch.

● Create a **new branch** for this commit and start a pull request.

wkchanG7309-patch-1

7.2 [**Author A**] Check in the revised code

- Press [**Create pull request**]



Update mcr.cpp

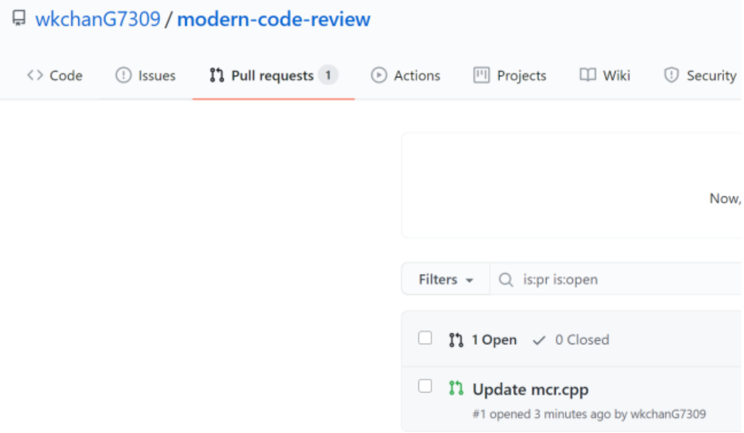
Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

8. [**10-25 minutes**] **R reviews P and comments in the e-channel and posts comments**

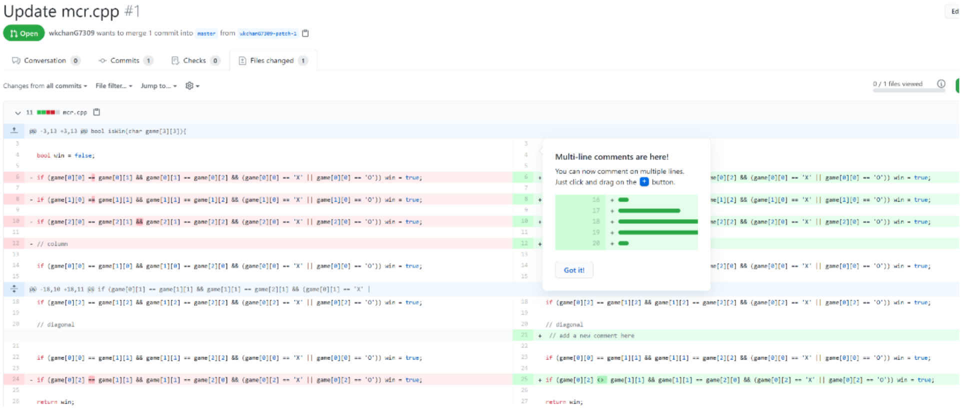
- Navigate to a webpage to review the "patch" and project members should comment on the changes.

8.1 **[All students]** Press [**Pull request**] menu item in this project, which will see the following screen, and click the “Update mcr.py”.



Update mcr.cpp
#1 opened 3 minutes ago by wkchanG7309

8.2 [All students] Press [File changed]



Update mcr.cpp #1

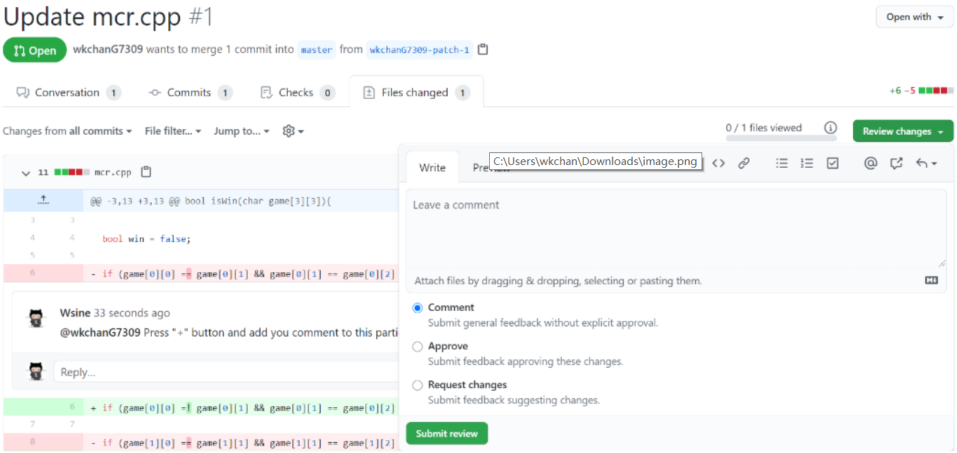
wkchanG7309 wants to merge 1 commit into master from wkchanG7309-patch-1

Multi-line comments are here!
You can now comment on multiple lines.
Just click and drag on the + button.

```
@@ -3,13 +3,13 @@ bool isWin(char game[3][3]){
3
4     bool win = false;
5
6   - if (game[0][0] == game[0][1] && game[0][1] == game[0][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;
7
8   - if (game[1][0] == game[1][1] && game[1][1] == game[1][2] && (game[1][0] == 'X' || game[1][0] == 'O')) win = true;
9
10  - if (game[2][0] == game[2][1] && game[2][1] == game[2][2] && (game[2][0] == 'X' || game[2][0] == 'O')) win = true;
11
12  - // column
13
14    if (game[0][0] == game[1][0] && game[1][0] == game[2][0] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;
15
@@ -18,10 +18,11 @@ if (game[0][1] == game[1][1] && game[1][1] == game[2][1] && (game[0][1] == 'X' |
18    if (game[0][2] == game[1][2] && game[1][2] == game[2][2] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;
19
20    // diagonal
21
22    if (game[0][0] == game[1][1] && game[1][1] == game[2][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;
23
24  - if (game[0][2] == game[1][1] && game[1][1] == game[2][0] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;
25
26    return win;
```

```
18    if (game[0][2] == game[1][2] && game[1][2] == game[2][2] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;
19
20    // diagonal
21  +  // add a new comment here
22
23    if (game[0][0] == game[1][1] && game[1][1] == game[2][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;
24
25  + if (game[0][2] <> game[1][1] && game[1][1] == game[2][0] && (game[0][2] == 'X' || game[0][2] == 'O')) win = true;
26
27    return win;
```

8.3 **[All students]** Add your comments and then press [**Review changes**] followed by [**Submit review**] as comments. One example on line 6 is as below.



Update mcr.cpp #1

wkchanG7309 wants to merge 1 commit into master from wkchanG7309-patch-1

```
@@ -3,13 +3,13 @@ bool isWin(char game[3][3]){
3  3
4  4    bool win = false;
5  5
6     - if (game[0][0] == game[0][1] && game[0][1] == game[0][2]
```

Wsine 33 seconds ago
@wkchanG7309 Press "+" button and add you comment to this parti

```
   6  + if (game[0][0] =! game[0][1] && game[0][1] == game[0][2]
7  7
8     - if (game[1][0] == game[1][1] && game[1][1] == game[1][2]
```

Attach files by dragging & dropping, selecting or pasting them.

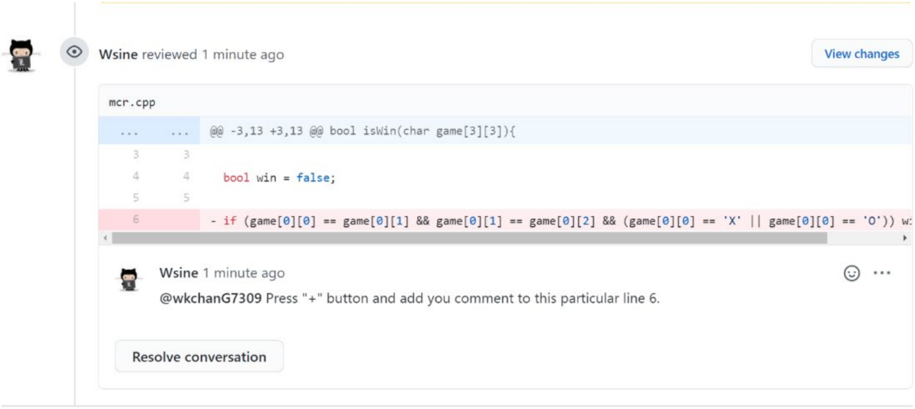
● Comment
Submit general feedback without explicit approval.

○ Approve
Submit feedback approving these changes.

○ Request changes
Submit feedback suggesting changes.

9. [**10-25 minutes**] **Author A answers issues and provide clarifications made by R and sends out revised patch P’ iteratively.**

9.1 **[All students]** Start a conversation by pressing [**Conservation**] or you can start reviewing other part of the code



Wsine reviewed 1 minute ago

mcr.cpp

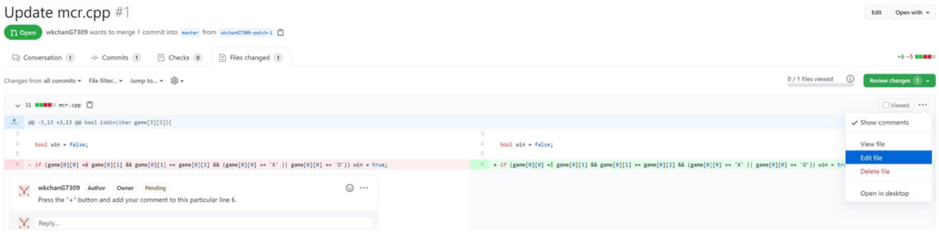
```
...   ...   @@ -3,13 +3,13 @@ bool isWin(char game[3][3]){
3     3
4     4       bool win = false;
5     5
6           - if (game[0][0] == game[0][1] && game[0][1] == game[0][2] && (game[0][0] == 'X' || game[0][0] == 'O')) wi
```

Wsine 1 minute ago

@wkchanG7309 Press "+" button and add you comment to this particular line 6.

9.2 **[Author A]** Change the code according to the consensus.

- Press [file changes]
- Press […] in mcr.py, and then press [edit file] in the pop up menu.
- Edit the file
- Select “Commit directly to the <<**YOUR-PATCH**>> branch.”
- Press [Commit changes]



Update mcr.cpp #1

Open wkchanG7309 wants to merge 1 commit into `master` from `wkchanG7309-patch-1`

11 mcr.cpp

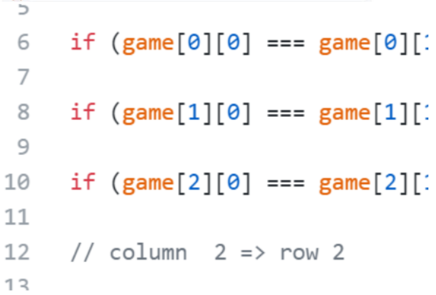
```
@@ -3,13 +3,13 @@ bool isWin(char game[3][3]){
3
4     bool win = false;
5
6   - if (game[0][0] == game[0][1] && game[0][1] == game[0][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = true;
```

```
3
4     bool win = false;
5
6   + if (game[0][0] == game[0][1] && game[0][1] == game[0][2] && (game[0][0] == 'X' || game[0][0] == 'O')) win = tru
```

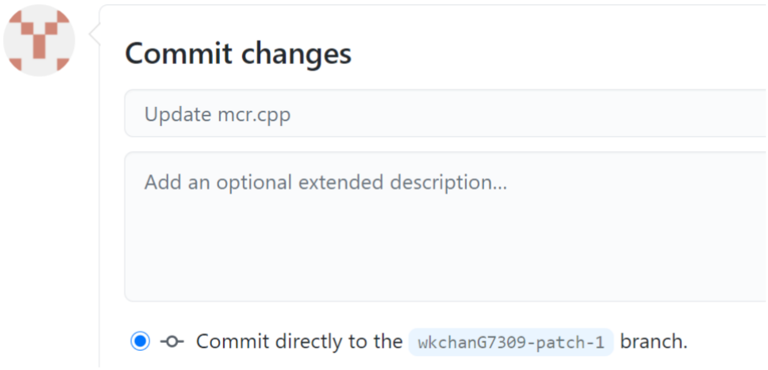
wkchanG7309 Author Owner Pending

Press the "+" button and add your comment to this particular line 6.

Reply...



```
5
6    if (game[0][0] === game[0][
7
8    if (game[1][0] === game[1][
9
10   if (game[2][0] === game[2][
11
12   // column  2 => row 2
13
```



Commit changes

Update mcr.cpp

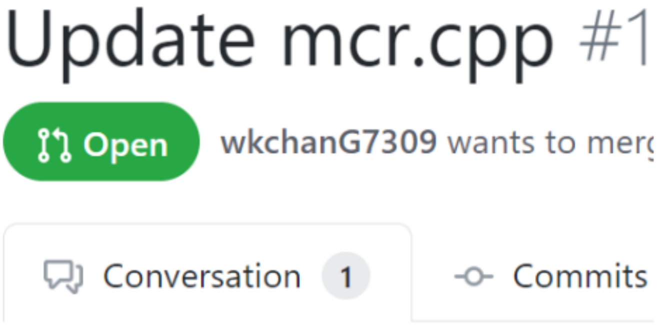
Add an optional extended description...

● Commit directly to the `wkchanG7309-patch-1` branch.

9.3 **[All students]** Repeat steps 8-9 until the review session is ended.

10. [**25-30 minutes**] **R approves P and Author A applies the final patch P' get the final corrected code C2**

10.1 **[Author A]** After all reviews done, author goes to [**Conversation**] followed by pressing [**Merge pull request**].



Update mcr.cpp #1

11. [**the 30th minute**] Author A **submits the corrected C to Canvas** as a commit.