

「計算と論理」 GitHub でのレポート提出

五十嵐 淳

`cal18@fos.kuis.kyoto-u.ac.jp`

京都大学

October 2, 2018

前提

- git add, git commit を使って変更履歴の記録が取れる
- その0に従って GitHub アカウントを作り, 自分の教科書レポジトリを作った
 - ▶ 以下アカウント名を aigarashi とする
 - ▶ レポジトリの URL は `https://github.com/ComputationAndLogicAtKUEng/hw-aigarashi` になる

通知をメールで受け取る設定

- 1 `https://github.com/ComputationAndLogicAtKUEng/hw-aigarashi` にアクセス
- 2 画面右上の自分のアイコンのプルダウンメニュー



→ Setting → Notifications

- 3 Watching の欄の Email にチェックを入れる

Issues¹ などを通じた教員側からの連絡 (レポートの不備など) の通知をメールで受け取れます

¹知らない人は “github issues” でググって :-)

レポジトリの複製を手元にする

宿題をやるPCで以下を実行。(コマンドは一行で。)

https で接続する場合:

```
git clone
https://github.com/ComputationAndLogicAtKUEng
/hw-aigarashi.git
```

公開鍵を GitHub に登録し ssh 接続する場合:

```
git clone
git@github.com:ComputationAndLogicAtKUEng
/hw-aigarashi.git
```

hw-aigarashi というディレクトリができその中に全
ファイルがダウンロードされる

レポジトリの状態を見る

```
$ cd hw-aigarashi
$ git status
ブランチ master
Your branch is up to date with 'origin/master'.
nothing to commit, working tree clean
```

- origin/master とは GitHub 側のレポジトリについた名前
- up to date とあるので、同期している (同じ状態) であることがわかる

宿題をやる

- CoqIDE なり Emacs で .v ファイルを編集してください
- `make XXXX.vo` (もしくは CoqIDE なら Compile buffer) が成功することを確認すべし
- `git add`, `git commit` をして変更履歴を記録する

再びレポジトリの状態を見る

```
$ git status
```

```
ブランチ master
```

```
このブランチは 'origin/master' よりも 1 コミット  
進んでいます。
```

```
(use "git push" to publish your local commits)
```

```
nothing to commit, working tree clean
```

- 手元は、GitHub 側よりも変更が施されているので、「1 コミット進んでい」る。
 - ▶ 「1」はコミット数によって変わります

GitHub への提出

コミットをリモートに「push」する

```
$ git push
```

と再び同期される

```
$ git status
```

```
ブランチ master
```

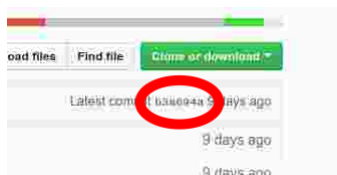
```
Your branch is up to date with 'origin/master'.
```

```
nothing to commit, working tree clean
```


提出確認

以下を比較して一致していればOK

- ブラウザでレポジトリ (の master ブランチ) を表示した時の Latest commit の後のハッシュ値



- `git rev-parse --short HEAD` を実行した時の先頭の commit のハッシュ値 (の上7桁)

```
$ git rev-parse --short HEAD  
abcdef0
```

トラブル: push できない!

GitHub 側に変更があると push できない

```
$ git push
...
error: failed to push some refs to ...
...
```

代表的な原因:

- ふたつ以上の PC/Mac で課題をやっている
- 教科書が教員によって書き換えられた

解決: リモート側の変更を取り込む

`git pull` を実行

- 「マージ」が発生し, リモート側と手元の変更が両方施された状態になる
 - ▶ コミットログを求められることがあるのでその場合はふつうの `commit` と同様にログを保存するとマージ完了
 - ▶ `git push` できる
- 編集箇所がぶつかると `conflict` と呼ばれる状態になる
 - ▶ 手でいい感じに編集して `add, commit`
 - ▶ `git push` して, この `commit` をリモートへ送信

git がわからない…

Understanding Git Conceptually:

<https://www.sbf5.com/~cduan/technical/git/> を
読むことをおすすめします。