

---

# Grading Process

Ralph W. Crosby, PhD.

01-Feb-2023 12:01:59

---

## 1 Grading Process

Assumes Grading directory is pathed (via `.envrc`)

### 1.1 Pre-processing

1. Make Directory Structures, copy file templates  
From the assignment root directory:  
`MakeStructure.sh`
2. Tailor assignment specific grading configuration and template files  
In the assignment `Grading` directory:
  - `GradingTemplate.md`
  - `GradingConfig.sh`
3. Build student directories from downloaded zip file
  - From the assignment `Grading` Directory  
`StudentDirectoriesFromZip.sh ~/Downloads/<zipname>`
  - A individually downloaded file can be processed from the `Grading` directory with  
`StudentDirectoryFromDownload.sh <LastName> <FirstName> ~/Downloads/<zipname>`
4. Make the grading markdown files  
From the assignment `Grading` Directory
  - On-Time Submissions  
`MakeGradingFiles.sh`
  - Late Submissions  
`MakeGradingFiles.sh -late`

### 1.2 Grade Assignments

### 1.3 Post-processing

1. Make feedback files  
From the `Working` directory:  
`MakeFeedbackFiles.sh`
2. Proof feedback files and remake as required  
From the assignment `Grading/Working` directory  
`find . -type f -name '* - Feedback.pdf' -exec open {} \;`
3. Create links to the feedback files  
From the assignment `Grading/Feedback` directory  
`find ../Working -type f -name '* - Feedback.pdf' -exec ln -s {} . \;`

## 2 Environment Variables

Note: all variables must be exported so Python can access them

### 2.1 Course Level `GradingConfig.sh`

- `CourseNumber`

- CourseName
- AssignName

### 2.2 Student Assignment Level Info.sh

- StudentCode
- StudentFirstName
- StudentLastName
- StudentTimestamp
- StudentSubmissionFile

## 3 Scripts

### 3.1 MakeStructure.sh

Build the grading directory structure in an assignment directory

See `-h` output for details.

Files Used

- ../../GradingTemplate.md
- ../../GradingConfig.md

### 3.2 StudentDirectoriesFromZip.sh

Process a downloaded zip file of student submissions building one directory per student. Expand the student compressed file into the directory

See `-h` output for details.

### 3.3 MakeGradingFiles.sh

Loop through the student directories building tailored grading files

From the grading directory

`MakeGradingFiles.sh`

See `-h` output for details.

Files Used

- ./GradingTemplate.md
- ./GradingConfig.sh

### 3.4 MakeGradingFile.py

Tailor the grading template file into the student specific file

Assumes Assign, Course, and Student environment variables are set

`MakeGradingFile.py <template> <output>`

### 3.5 MakeFeedbackFiles.sh

Build pdf files from the grading files

See `-h` output for details.

### 3.6 Markdown2Pdf.py

Build a pdf file from a markdown file using a standard pandoc configuration file

Markdown2Pdf.py <markdown input file> <pdf output file>

See -h output for details.

Files Used

- scriptdir/Markdown2Pdf\_PandocConfig.yaml
- ~/SharedEnvironment/Latex/CofCGrading.sty

## 4 LaTeX Processing

### 4.1 Heading block:

```
-----  
CSCI-xxx - Course Title  
Assignment Title  
  
Student Name  
  
File Submitted: filename  
  
Submission Time: timestamp  
-----
```

### 4.2 Related Files

- Markdown2Pdf\_PandocConfig.yaml  
Pandoc configuration file
- Markdown2Pdf.py  
Python script to build a single pd
- SampleMarkdown.md  
Sample file for testing
- SamplePdf.pdf  
Sample output file from testing
- CofCGrading.sty  
Part of SharedEnvironment/Latex