CMPUT 201 LAB SUBMISSION NOTES

SUBMISSION PROCESS

Compilation: Before submitting, always ensure that your code compiles without any errors. For C programs, you can use the following command:

Packaging with tar: Once you've ensured that your code compiles and runs correctly, you need to package your submission files using `tar`. For instance:

Checking the tar file: Before submitting, you should always check your tar file to ensure it contains all the necessary files. You can use the provided `check` program for this:

If everything is correct, you should see a 'Successful!' message.

Transferring files with scp (Secure Copy Protocol): If you're working remotely and need to transfer your submission file to or from a lab machine, you can use `scp`. Here's how you can copy the `submit.tar` file from a lab machine to your local machine:

This command copies the `submit.tar` file from the lab machine to the current directory on your local machine.

Submitting on eClass: Once you have the `submit.tar` file on your local machine, you can then upload it to eClass for submission.

ADDITIONAL NOTES FOR GIT BASH USERS

If you're using Git Bash on Windows, the process is largely the same. However, there are a few things to keep in mind:

Directory Paths: In Git Bash, directory paths use forward slashes (`/`) instead of backslashes (`\`). For example, `C:/Users/Ayub/Desktop` instead of `C:\Users\Ayub\Desktop`.

scp on Git Bash: When using `scp` in Git Bash, ensure you're in the directory where you want to save or retrieve the file. For instance, if you want to save the `submit.tar` file in a specific directory on your local machine, navigate to that directory in Git Bash first, then run the `scp` command.

Remember, the key to a smooth submission process is to always double-check your work and ensure you're following the submission instructions carefully.

```
scp ayub2@ug30.cs.ualberta.ca:~/c201/labs/lab01/
submit.tar .
```

```
chmod 700 check
./check submit.tar
```

```
tar -cvf submit.tar ls_com.txt ls_out.txt hello.c
readme.txt check
```

```
gcc -Wall -std=c99 hello.c -o hello
```