Select the right page in Gradescope or we will not grade the question!

# Homework 3
# Shells, Environment, and Scripting

**Due: Saturday, January 28, 10:00PM (Hard Deadline)**

## Submission Instructions

Submit this assignment on Gradescope. You may find the free online tool PDFescape helpful to edit and fill out this PDF. You may also print, handwrite, and scan this assignment.

## 1 Understanding your `PATH`

In a terminal, type `PATH=` (just hit enter after the equal sign, no space characters anywhere). Try to use the terminal like normal (try running `ls`). What happened?

**Give an example of a command that used to work but now doesn't:**

**Can you still run this command with an empty `PATH`? How?**

**Give an example of a command that works the same even with an empty `PATH`. Why does this command still work?**

## 2 Playing with the shell a bit: Special Variables

Bash has quite a few special variables that can be very useful when writing scripts or while working at the terminal.

**What does the variable `$?` do? Give an example where this value is useful**

**What does the variable `$1` do? Give an example where this value is useful**

# 3 Basic Scripting

Recall from lecture that scripting is really just programming, only in a very high-level language. Interestingly, `sh` is probably one of the oldest languages in regular use today.

`make` is a good tool for build systems, but we can actually use some basic scripting to accomplish a lot of the same things. First, write a simple C program that prints "Hello World!". Write a shell script named `build.sh` that performs the following actions:

1. Compile your program
2. Runs your program
3. Verifies that your program outputs exactly the string "Hello World!"
   - There are good utilities that check the `diff`erence of two files. They could be helpful.
4. Prints the string "All tests passed." If the output is correct, or prints "Test failed. Expected output >>Hello World<<, got output >>{the program output}<<".

**Copy the output of `cat build.sh` here:**

# 4 Controlling your environment

In lecture, we added a directory to our `PATH` so that we could just type `hello` and the Hello World program would run. It would be annoying to update the `PATH` variable every time we open a new terminal. Fortuntately, we can do better.

**Describe how you would set up your system to modify your `PATH` automatically every time you open a new terminal (what file would you change and what would you put in it?)**

**Roughly how long did you spend on this assignment?** _____