CS240 - Operating Systems
Assignment #4
Spring 2024

For this exercise, you are to implement a simple "shell" program, similar to a simple bash shell. Call your shell `uish` (for "University of Idaho shell").

- When invoked it should produce a prompt: `$`
- You should be able to enter any Unix/Linux command name, including options, and the program should execute properly. Your shell should parse the input, then execute the command using the fork/exec mechanism.
- Your shell should support shell variables. You should be able to define variables and use them in shell commands:

```
ABC=xyz
echo $ABC
```

- One shell variable that should be used is `PATH`. This should work just like the "real" shell. That is, once defined, it should prepend each file prefix in the PATH variable until either it finds an executable, or it goes through the full list and doesn't find a match, in which case your shell should print an error message, similar to the real shell. NOTE that your shell should use `execv()` (rather than `execp()` or `execvp()`), since your program should do the PATH processing.
- Your shell should wait until the command completes, then it should issue another prompt and be able to accept another command. EXCEPT:
- Your shell should support "background" execution. That is, if the command contains the '&' (ampersand) at the end, your shell should launch the command, but return immediately, displaying the prompt and be ready to accept another command.

As before, your program should not use the real shell for any processing, other than to start your program.