Environment Variables

What are they

- Holds data specific to the running process
  - Stored per process
- Newly created processes inherit environment of parent process
  - Can be used to pass information down

Useful Examples

- echo $PATH
  - ~/bin/git/:/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin
- echo $HOME
  - /home/faculty/cop4610t
- echo $USER
  - cop4610t
- echo $SHELL
  - /bin/bash
- echo $PWD
  - /home/faculty/cop4610t/public_html

- export UNDEFINED=now_defined; echo $UNDEFINED
  - now_defined

How To Access in C

- #include <stdlib.h>
  - char *getenv(const char* name)
    - *char *path = getenv(“USER”);*
    - If defined, returns value of $name
    - If undefined, returns NULL
- You do not need to free the values when you are done with them
  - These are already defined values
  - Changing them changes the actual stored value

How to Change in C

- #include <stdlib.h>
  - int setenv(const char *name, const char *new_value, int override)
  - Only creates new values if override != 0
  - returns 0 on success, -1 on failure
- Assigns new_value to the value of name