CS 111 Discussion Notes—Lab 1.a
Updated 3 Apr 2013

What you need to do

The main.c file is the main file—*main* already exists and you shouldn't need to modify it for lab 1.a. It has a line

while ((command = read_command_stream (command_stream))) { … }

The idea is that *read_command_stream* should read the next top-level and-or from the input stream, returning NULL if there are no more left. *command_stream* is of type **command_stream_t** = **struct command_stream ***; you will need to define this struct. The intention is for it to be something that, when an instance is passed to read_command_stream, it returns the next top-level and-or. command.h is a good place for this definition.

You will also need to implement *make_command_stream*, which should create and initialize a command_stream_t instance. Both read_command_stream and make_command_stream are in read-command.c .

Types

```
enum command_type
  {
    AND_COMMAND,          // A && B
    SEQUENCE_COMMAND,     // A ; B
    OR_COMMAND,           // A || B
    PIPE_COMMAND,         // A | B
    SIMPLE_COMMAND,       // a simple
                          //   command
    SUBSHELL_COMMAND,     // ( A )
  };
```

```
struct command
{
  enum command_type type;
  int status;
  char *input;
  char *output;
  union
  {
    struct command *command[2];
    char **word;
    struct command *subshell_command;
  } u;
};
```

The **command_type** remembers what type of command is under consideration. AND and OR correspond to and-ors in the assignment guidelines; SEQUENCE to complete commands, PIPE to pipelines, SIMPLE to commands, and SUBSHELL to subshells.

A **command** has five fields. The first is the type. The second, which is not needed for 1.a, is a nonnegative integer containing the exit status of the command, or -1 if unknown/still running. The third and fourth are used for I/O redirection, and should be NULL if they use stdin or stdout.

The fifth field is a union. The member of the union used should depend on the command type. AND, SEQUENCE, OR, and PIPE commands should use *command*. SIMPLE commands should use *word*. SUBSHELL commands should use *subshell_command*.

Other existing functions

There are three functions, prefixed with "checked_", that used for allocating memory, which should be used instead of the non-checked versions: *checked_malloc*, *checked_realloc*, and *checked_grow_alloc*. checked_grow_alloc doubles the allocated size if possible, else raises it to the maximum possible, else fails.

print_command is the function that is called to print a command; it calls *command_indented_print*, which prints a command recursively. This latter function is useful to understand with what parsed data the command fields should be filled. You should **not** modify either of these functions.