

## C++ Project Step 2

Please read the project description and refer to it while reading this.

Our second step will implement the commands without handling triggers or checking conditions. To do that you must also build the objects that represent the dungeon. We will also implement a few commands without worrying about triggers. In particular:

*n, s, e, w*; movement commands to put the player in a different room. If there is not another room (i.e., *border*) in the indicated direction, print something like “cannot move that way”.

*i*: short for inventory. This command print *Inventor:* and then lists all of the items in the player’s inventory, with items separated by commas if there is more than one. If there are no items in the inventory, display *Inventory: empty*.

*take <item>*: Changes the ownership of the item named *<item>* from the room or container (e.g., a chest) it was contained in to the inventory of the player. If successful, print *Item <item> added to the inventory*.

*open exit*: If a room is of type exit this command prints *Game over* and gracefully (i.e., no seg fault, etc.) ends the program.

*drop <item>*: Removes an item from the inventory and makes the room the player is in the owner of the item. The command should print *<item> dropped*. If the item is not in the inventory, print *<item> not in inventory*.

### Behind the Scenes Commands – implement these as needed.

*Add <object> to <room/container>*: creates an instance of *<object>* with the specified room or container being the owner. This does not work with the inventory.

*Delete <object>* Removes *<object>* from the game, but it can be returned to the game by using an *Add* command. Rooms cannot be added back in if deleted.

*Update <object> to <status>*: creates a new status for *<object>*. Object status can be checked by triggers.

*Game Over*: ends the game and prints *Victory!*

### What you should turn in:

Turn in your code to the code part of Step2.

A video showing you downloading your code from Brightspace and performing the commands described below on *sample.xml*, *itemsample.xml* and *rooms.xml*. This video is turned into the video part of Step2. Try to keep videos brief as the graders will have approximately 250 projects to grade. This will be easier to do if you practice downloading and building your project and videoing a run before you do the actual video.

In sample.xml, enter the dungeon, navigate to another room, and return back.

In itemsample.xml, do an inventory, pick up an item, do an inventory, drop the item, do another inventory. The first inventory should show an empty pack, the **second a pack containing the picked up item, and the third an empty pack.**