

## **Testing plan (only for new changes in model):**

### **Both wrapping and non-wrapping dungeon:**

Player action: shootArrow(Direction dir, int distance):boolean

- should return true only if the monster is hit based on the provided parameters.

## **Testing the controller:**

### **With actual model and mock view:**

Method: playGame(DungeonView v):void

- Should begin the game by placing the player at the start location; verify by getting the player location and game status from the model.
- Should call view's player information refresh method and dungeon refresh method.

Method: move(Direction dirToMove):void

- Should move the user in the right direction; verify by getting the player location and comparing the location.
- If there is no move in the direction, should continue the game.
- If the user's move is valid, should call view's player information refresh method and dungeon refresh method.

Method: collectTreasure():void

- In a location with no treasure, the game should continue without issues.
- In a location with treasure, the treasure must be picked correctly; verify by getting player's description and checking for treasure quantity, Also, location's treasure should be reduced.
- In case of picking up the treasure, should call view's player information refresh method and dungeon refresh method.

Method: pickWeapon():void

- In a location with no weapon, the game should continue without issues.
- In a location with weapon, the weapon must be picked correctly; verify by getting player's description and checking for weapon quantity, Also, location's weapon quantity should be reduced.
- In case of picking up the weapon, should call view's player information refresh method and dungeon refresh method.

Method: shootArrow(Direction, int):void

- Should shoot arrow in the right direction and distance specified; If model returns, stench at the location must change; else, the stench should remain the same.
- The number of arrows user has should reduce by 1; verify by getting player description and checking for arrow quantities.

- After the user's shoots, should call view's player information refresh method and dungeon refresh method.

Method: setUpGame(String name, int row, int column, int interConn, int treasureP, int difficulty):void

- If the dungeon can't be created with the provided parameters, should throw exception that the model throws.
- If dungeon creation is successful, should begin the game by placing the player at the start location; verify by getting the player location and game status from the model.
- Should call view's assignReadOnlyModel method by passing a read only model, player information refresh method and dungeon refresh method.

**With mock model and mock view:**

- For methods in the model that take parameters from controller, the parameter passed should match the parameters passed to the controller methods.
- Should also call the right methods on the view.