Group Project

Group members: Burnie, Nate

Class Tasks:

1. Deck Class
   1. Inheritance:
      1. Publicly inherits from the Collection class.
   2. Special Member Functions:
      1. Defines special member functions to ensure the deck is initially empty.
      2. All fields are private, with no limitations on the number of fields.
   3. Overridden Methods from Collection:
      1. Insert()
         1. Adds a Token into the deck based on the flag (string) parameter.
      2. Remove()
         1. Removes a Token from the deck based on the flag (string) parameter.
   4. View()
      1. Returns a Token from the deck based on the flag (string) parameter if the deck is not empty and the flag is valid; otherwise, throws an error.
   5. Extend()
      1. Sets whether the display of the deck is extended or compressed.
   6. Size()
      1. Returns the size of the deck.
   7. Empty()
      1. Returns true if the deck is empty; otherwise, returns false.
   8. Clear()
      1. Empties the deck.
   9. Info()
      1. Returns a string listing the flags for the Insert(), Remove(), and View() methods.
   10. ToString()
       1. Returns a string in the format:
          1. [0] if the deck is empty.
          2. [t]... [b] if the deck view is extended, where t is the top token and b is the bottom token, each token on a separate line.
          3. [t] if the deck view is compressed.
2. The Deck Class
   1. Inheritance:
      1. Publicly inherits from the Program class.
      2. Special Member Functions:
   2. Defines special member functions but deletes its copy constructor and assignment operator.
      1. All fields are private, with no limitations on the number of fields.
   3. Overridden Methods from Program:
      1. Initialize()
         1. Configures the game into its initial state.
      2. Move()
      3. Performs a task based on the flag (string) parameter.
   4. Undo()
      1. Undoes the previous move performed.
   5. Completed()
      1. Returns true only if the game has ended.
   6. State()
      1. Returns a string indicating the status of the game.
   7. Info()
      1. Returns a string listing the flags for the Move() method.
   8. ToString()
      1. Returns a string representing the game board.

Our game choice is Ball Sort!

Game instructions: the game consists of three rules:

1. Game will start with ‘n’ number of ball colors and n+2 number of tubes to sort them in
2. You can place any color object into an empty container
3. If the container is not empty, only objects matching the topmost object can be placed inside the container
4. Tubes can contain up to four objects and the goal is to sort all the objects of the same color into their own individual containers.