**CSD311 Assignment-4**

**Members of the Team:**

Aditya Srivastava (1910110034) (35%)

Arinjay Srivastava (1910110087) (30%)

Prabhat Kumar Singh Gaur (1910110272) (35%)

**Programming Language Used: Java**

**Operators:**

1. **S (X, Y):** Put block X on top of block Y. For this operation, the top of Y must be clear and X should be in the robot’s arm.
2. **US (X, Y):** Pickup block X from block Y. For this operation, the arm must be empty and the top of X must be clear.
3. **PU (X):** Pickup X from the table and hold it. For this operation, the arm must be empty and the top of X must be clear.
4. **PD (X):** Put down X on the table. For this operation, the arm must be holding X before putting it down.

**Predicates:**

1. **ON (X, Y):** Block X is on block Y.
2. **ONT (X):** Block X is on the table.
3. **CL (X):** Top of block X is clear.
4. **HOLD (X):** The robot arm is holding block X.
5. **AE:** Robot arm is empty.

**Classes**

1. **Operator:** This class stores the type of operations that need to be performed on the stack to make it progressively reach the goal state.

**Member Functions:**

(a) ArrayList<Predicate> getPre\_C ()

(b) ArrayList<Predicate> getInsList ()

(c) ArrayList<Predicate> getRemoveList ()

1. **Block:** This class is being used to signify a block. It stores the notation of a particular block.
2. **Predicate:** This class stores and sets the type of predicate on a single block or on two blocks.
3. **Goal:** This class is being used to store the final goal desired by the user. It stores all the elements using an arraylist and keeps the operation to be done.

\*In the above classes Operator, Predicate and Goal we have overridden constructors depending on the requirement.

1. **BlockWorldProgram:** This is the main class of the application. It contains the main function and a static function to create stacks by providing appropriate predicates. It also contains two array lists, which are being used to store the initial and the goal state.

The main function keeps in check all the intermediate stages, it checks what is the state of the current goal and set predicates for it accordingly. Then operator class is called from which we get the operations that need to be performed in order to make our current stage reach the current goal state. This process of simultaneous insertion, removal of predicates, and operations is performed until we reach the goal state.

**Member Functions:**

* + - * 1. public static ArrayList<Predicate> getStackFormation (ArrayList<ArrayList<Block>> initialState, int sizeOfState)
        2. public static void main (String args []) throws IOException

\*For properly understanding the role of each variable, function or class being used in the project, we have used comments explaining the functions of the variables we have used. For test cases, we have attached text files containing the output of each test case given in the question.