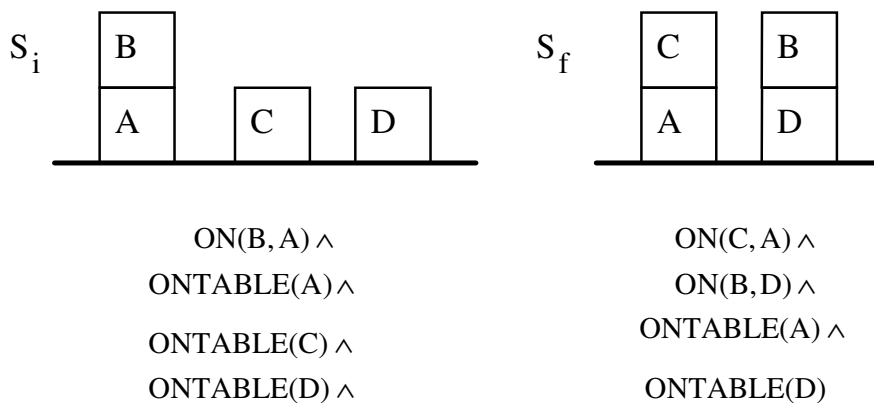


BDI Agent for the Blocks World

Consider an initial configuration in the **blocks world**. The blocks are identified by capital letters. Blocks can be placed one on top of the other, or on table, which is considered infinite. Blocks can be moved by a robot arm. Suppose that the **BDI agent** is the **robot arm**, which can pick blocks and can put them down, one block at a time. The **task** of the robot is to create a **plan** for moving the blocks from the initial position into the final position. The possible actions of the robot are: **STACK**, **UNSTACK**, **PICKUP**, **PUTDOWN**. The state of the robot arm can be checked using the predicates **ARMEMPTY** and **HOLD**. If more blocks are stacked one on top of the other, only the top block can be unstacked. A block can only be on the table, stacked on another block, or held by the robot arm.

A configuration from the blocks world is given below, where on the left is the initial state and on the right is the final state:



Implement in Java the BDI agent for solving different configurations from the blocks world, using the **single minded commitment** strategy.

Tasks:

- a) Define a **model for the representation of the blocks world**. This will be the model for the beliefs.
- b) Define a **model for the representation of goals/desires and intentions**.
- c) Implement the **options** and **filter** methods for the agent.
- d) Define a **planning method** to solve the problem.
- e) Implement the **BDI behavior** of the agent, using the components developed so far. Represent in a graphical interface the movement of the blocks.

Laboratory 4: Implement the data structures and the basic methods.

Laboratory 5: Implement the BDI agent working in the blocks world.