INSTRUCTIONS

The preferred environment is Linux. Ubuntu would be good.
The preferred version of g++ for compilation would be gcc version 4.6.3 (Ubuntu/Linaro 4.6.3-1ubuntu5).

The program written is a C++ file (blocks_world.cpp). Once compiled, please provide the first command line argument as the number of blocks and second command line argument as the number of stacks during execution.

Steps:

1. $ g++ blocks_world.cpp
2. $ ./a.out 10 5

Note that here the number of blocks is 10 and the number of stacks is 5.

The blocks in a state will be represented in the form of alphabets like this:

1 | A
2 | B D
3 | C E

Note that if the number of blocks exceeds 26, which is the max limit of English alphabets, it will include symbols from ASCII characters that are available after Z sequentially. It will look something like this after reaching final state:

1 | A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\
2 |
3 |

There is a macro named MAX_SCRAMBLE in the blocks_world.cpp file. This can be used to control the amount of scrambling of the goal node to create the start node from the goal node.

The number of stacks must be greater than or equal to three. This is to allow for the minimum number of stacks required to sort the elements into one stack. The number of blocks and stacks must be positive.

There is no limit on the number of iterations. Some of the solutions may take more time so feel free to stop the program if and whenever your patience wears out. :)

To manually input a state, please note that aside from the command line arguments for that particular state, changes are required in the blocks_world.cpp file. Note that the instructions for that are present in the blocks_world.cpp file itself alongside the now commented sample code for two states.

For Queries:
Email: aryarockstar@tamu.edu
Ph: 979-985-7204

Regards,
Abhi