

# **User Guide for Sokoban in RISC-V**

By: Jensen Sung

## Welcome to Sokoban!

You are a hardworking construction worker who is tasked with pushing a box to a location you are assigned to. You must do this in the fewest amount of moves to become the #1 worker of all time. You are competing with all of your fellow co-workers and whoever lands the top spot on the leaderboard gets to be framed as employee of the day!

## How to Run the Game

The game file for sokoban is meant to be run on <https://cpulculator.01xz.net/?sys=rv32-spim>.

Double check that the language selected is “RV32”. Once you have confirmed CPULATOR is in “RV32”, head to the top menu bar [Figure 1.1] and hover on **file** -> select **open** -> select **Sokoban.s** [Figure 1.2]. Once the file is loaded, you must click **compile and load** -> click **continue** at the top [Figure 1.3]. Then you are able to start the game. Once you are finished your game, select **restart** -> **continue** to start up again [Figure 1.1].

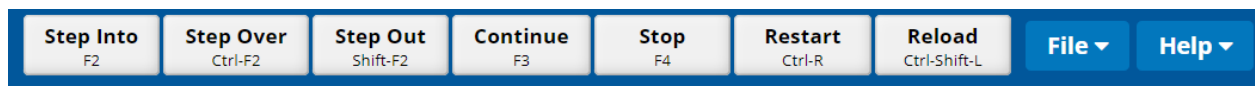


Figure 1.1

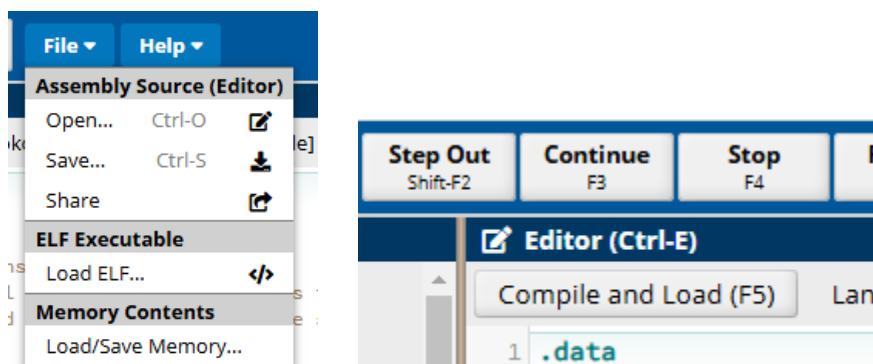


Figure 1.3

Figure 1.2

## How to Play

Once you have compiled and started (continue) the game, you are then able to choose the size of the game board using your input into the terminal, starting with # of **columns (x)** then # of **rows (y)** [Figure 2.1]. The values you choose must be **positive natural numbers** and **be larger than 1** otherwise the game will **NOT RUN AS INTENDED**.

You will then be prompted in the terminal with the number of players that will be playing. Enter only a **natural positive number ( $\geq 1$ )** otherwise the game will **NOT RUN AS INTENDED** [Figure 2.1]. Once entered, the game board will show on the terminal.

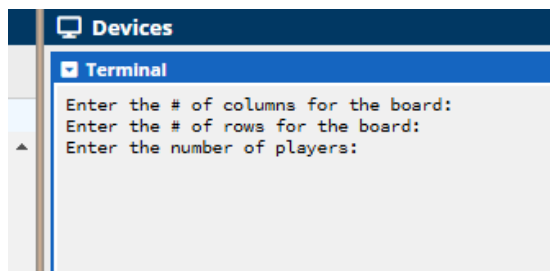


Figure 2.1

On the game board, you will see **5 different characters** which represent the following [Figure 2.2]:

- P: your **player** that you are controlling
- B: the **box** you are able to push
- X: the **target** where you must push the box to
- @: the **outer walls** of the board; you are not able to pass through the wall
- \_ : an **empty space** on the board where you are able to move your player (and/or box)

You may input one of w, a, s, d, r **in lowercase** as an input [Figure 2.2]. The inputs do the following:

- w: moves your player **up** one space
- a: moves your player **left** one space
- s: moves your player **down** one space
- d: moves your player **right** one space
- r: **restarts the game** board for the current player
- If any other character is inputting, the user will be notified that the board is unchanged and an invalid input was entered [Figure 2.3]

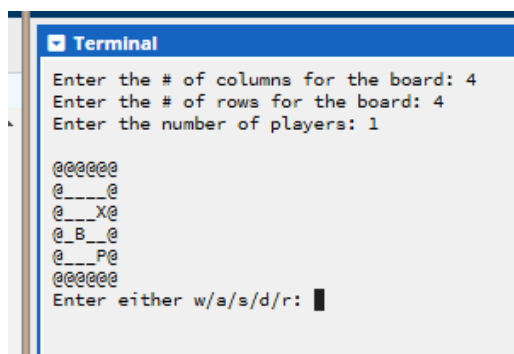


Figure 2.2

```

@@@@@
@___@
@_P__@
@_B__@
@_XB__@
@@@@@
Enter either w/a/s/d/r: g
The move you inputted is not valid. Please try again.
Enter either w/a/s/d/r: g
The move you inputted is not valid. Please try again.
Enter either w/a/s/d/r:

```

Figure 2.3

If a box is **on** of the space you are moving your player to, the box will be **pushed too** ONLY if there is **another empty space in front** of the box. Otherwise, it will be an invalid move which will notify the user in the terminal. For example, looking at [Figure 2.4], the user wanted to push the box right, but since the box is against the wall, the move is invalid. It will then prompt the user to enter another move.

```

@@@@@
@___@
@_X_@
@_PB_@
@_B__@
@@@@@
Enter either w/a/s/d/r: d
The move you inputted does not change the board.

```

Figure 2.4

Once you have successfully pushed the box onto the target, you are congratulated with a message **corresponding** to your **player** [Figure 2.5]. If you chose **more** than 1 player, the next player will go as the board will reset. The players will go in **chronological** order, starting from 1.

```

@@@@@
@___@
@___X@
@___B@
@___P@
@@@@@
Enter either w/a/s/d/r: w
Player #1 you have beat the game! Congratulations!

```

Figure 2.5

Once all players have gone, a **leaderboard** will be visible in the terminal, sorting all the players from **least to most moves** displaying the number of moves and player number [Figure 2.6].

```

Leaderboard:
Player #2 with total moves: 6
Player #1 with total moves: 8

```

Figure 2.6

You will then be prompted with whether you want to view a **replay** or not. A replay can be called on any of the players. Enter only a valid player number from **1 - max # of players** otherwise the game will **NOT RUN AS INTENDED**. When a player is chosen for a replay, the terminal will output the **initial game board** and the **sequence of moves** that specific player chose [Figure 2.7]. Enter **0** if you want the game to **exit**.

```

Enter a player number to replay their moves or 0 to exit: 1
Game is being restarted.

@@@@@
@X___@
@_P_@
@B___@
@___@
@@@@@
->d->s->s->a->a->w-
>w

Enter a player number to replay their moves or 0 to exit: 0

```

Figure 2.7

## Common Issues

There is no possible way to move the box anymore!

- In the case where the box is stuck (in a corner for example) and cannot be moved, the player must restart the game by entering 'r' into the console [Figure 3.1].

```

@@@@@
@___@
@___@
@___P@
@_X_B@
@@@@@
Enter either w/a/s/d/r: r
Game is being restarted.

@@@@@
@___@
@_P_@
@___@
@_XB_@
@@@@@
Enter either w/a/s/d/r:

```

Figure 3.1

The game is not compiling or the compiler failed!

- Some common fixes to common issues would be to
  - Redownload the file and try opening it again
  - Ensure that no new characters were mistakenly input into the editor
  - Ensure the language chosen at the top is RV32, not C [Figure 3.2]

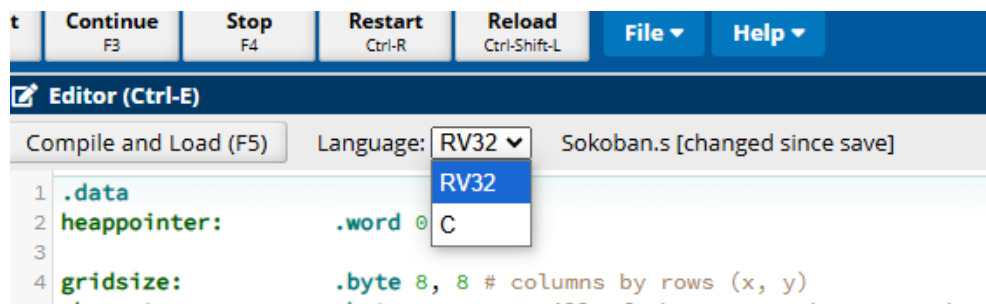


Figure 3.2

The terminal is cluttered with misaligned text or too messy!

- If the terminal gets too messy, you can always clear the terminal whenever desired by hovering over the tiny arrow to the left of the terminal [Figure 3.3]. It is recommended that you clear the terminal every time you restart the whole game.

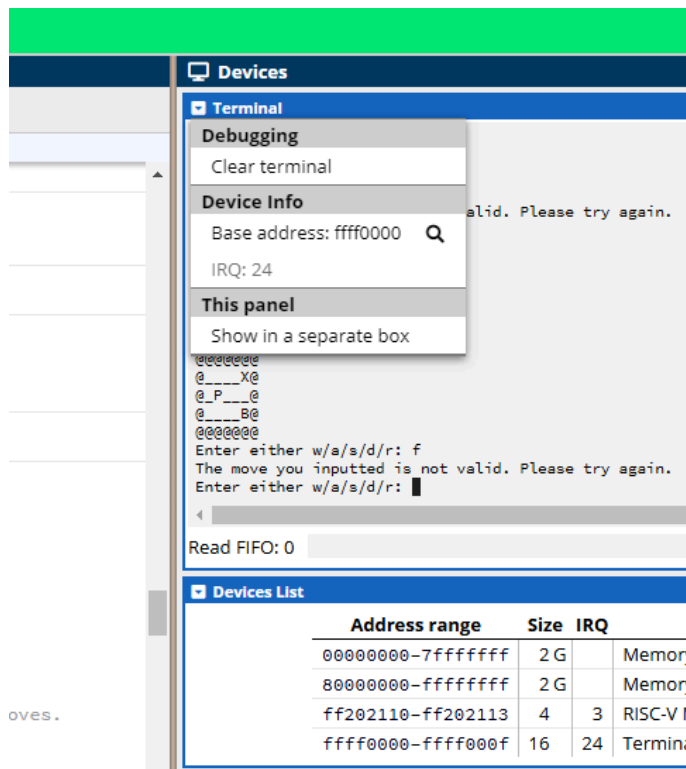


Figure 3.3