TEST

Test reset\_game: build or execution failed (reset\_game from a completely random memory state), returncode=(1)

Test reset\_game: 0 out of 1 passed -> Score: 0 / 5

MESSAGE

*Current step is 1 and displayed as such on the 7-SEG display.*

Therefore, the definition implies that reset\_game should overwrite 7-SEGs.

<https://moodle.epfl.ch/mod/forum/discuss.php?d=49274#p101106>

Hello,

Same remark as below, the game should be entirely playable (and displayed as such) right after the reset\_game call, make sure you draw the leds as well.

Best,

Alex.

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48902#p100678>

Hello,

Yes you should also make sure you display the first GSA. In general, a reset function should set the program to a valid ready-to-be-used state. In this case, we should be able to call reset\_state and then immediately play the game (e.g. check for inputs and compute the next state accordingly).

Best,

Alex.

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48902#p100677>

Hello,

I also failed reset without any information, yet my procedure does exactly what you explained. Perhaps, is it because I do not clear the LEDS in reset (or draw the GSA of seed 0) ?

Bests,

Alexis

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48902#p100622>

Hello,

For the reset\_game procedure, you should first check that you follow exactly the description, so you should make sure that with a single reset\_game call, your game should be back on INIT state with the first seed selected, number of steps is 1 (and displayed as such). You should also make sure that GSA\_ID is set to 0, the first GSA is set to the first seed, the pause is selected and the current speed is set to 1. You could use the simulator to make sure the internal values are correctly set.

The "X" undefined value on the 7-segs indicates that no assumptions are made on that value, you could have anything here and its value will not be checked. The video shows a 0 because it does not matter and is nicer.

Best,

Alex.

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48902#p100458>

Hello,

The reset\_game unique test at the grader fails without giving much details, i checked again the code + the visual result on the FPGA and i still dont get what's wrong with the method. Is there some procedure i have to follow to identify the problem ?

Another question: the GoL demo video indicates  that the leftmost 7-segmet shows 0 but the checklist says "segment displays show X001, where Xstands for an undefined value".  Which assumption should we consider true ?

Thank you.

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48902#p100383>

Hi Leo!

I would tell you that, in general, update\_state will handle the changes of state (together with actions that are followed by this change of state) and select\_action will handle the events that trigger actions in the same state.

Having said this, in the assignment we tell you to use reset\_game to handle the change of state to INIT, because this change of state implies several actions that you already handle in reset\_game.

Finally, random\_gsa (when transitioning from INIT to RAND) should already be called implicitly by another function that you will use in select action (I let you figure out which one it is ;) ).

I hope this clarifies things, otherwise, let us know,

Marcel

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48360#p99488>

Hello!

my partner and I were wondering whether the random\_gsa (when we transition from INIT to RAND) should be called within update\_state function or select\_action. furthermore, when button3 is pressed in the RUN state, and the game is reset, do we need to transition to the INIT state, or remain in the RUN, and should the reset\_game function be called within update\_state and select\_action.

Globally, we are confused about the relation between select\_action and update\_state, since they share some functionalities.

thank you, LL

<https://moodle.epfl.ch/mod/forum/discuss.php?d=48360#p99370>

TEST

Test increment\_seed: build or execution failed (Increments the seed from a random number), returncode=(1)

Test increment\_seed: build or execution failed (Increments the seed from a random number), returncode=(1)

Test increment\_seed: failed (Increments the seed from N\_SEEDS)

Test increment\_seed: 0 out of 3 passed -> Score: 0 / 5

MESSAGE

TEST

Test mask: build or execution failed (Mask a random GSA), returncode=(1)

Test mask: build or execution failed (Mask a random GSA), returncode=(1)

Test mask: 0 out of 2 passed -> Score: 0 / 5

MESSAGE

TEST

Test update\_gsa: build or execution failed (update gsa from a random state), returncode=(-9)

Test update\_gsa: passed (update gsa from a random state)

Test update\_gsa: 1 out of 2 passed -> Score: 5 / 10

MESSAGE