**Lock variable:**

* Lock variable is used to handle more than one processes to go into critical section at the same time. It have also disadvantage like mutual exclusion is not there in this algorithm.
* There are some scenarios like more than 1 processes foes into the critical section and it will be resulting into some major issues and does not hold mutual-exclusion principle.
* Algorithm of the lock variable is given below.

Entry Section →

While (lock! = 0);

Lock = 1;

//Critical Section

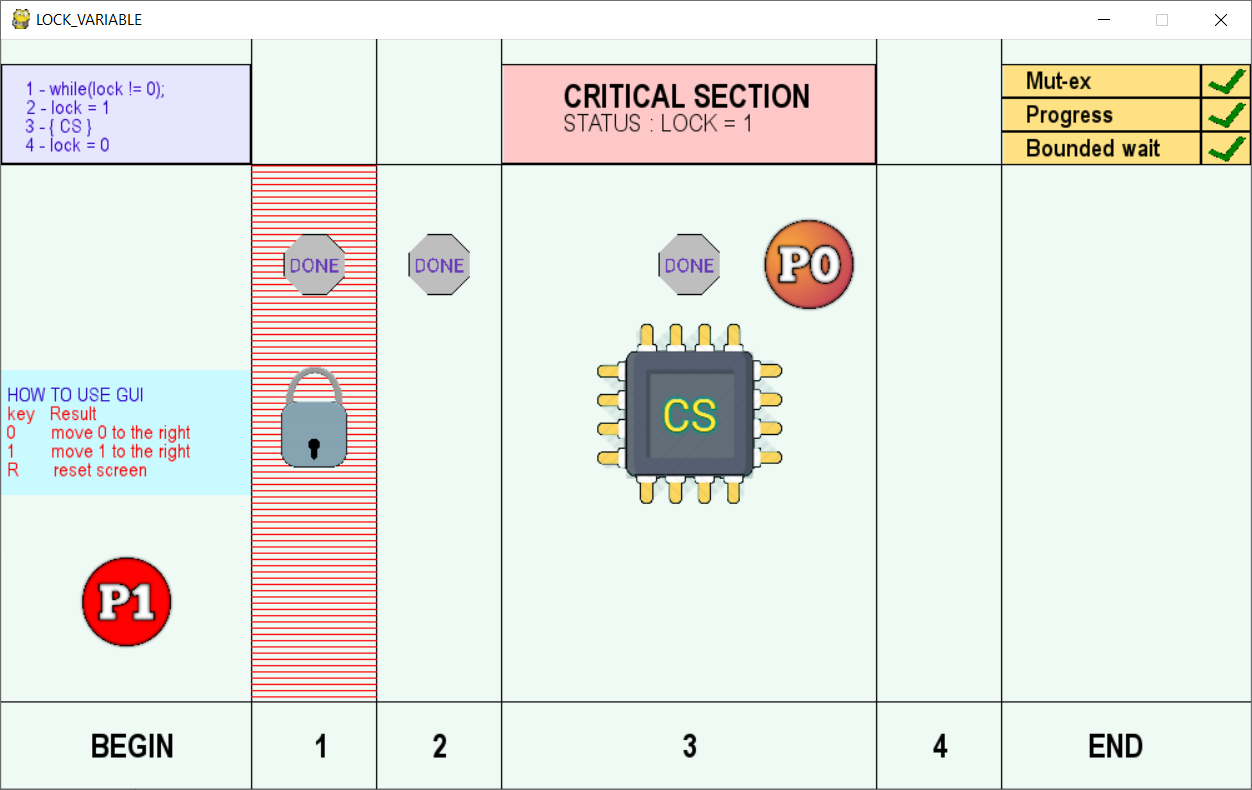
Exit Section →

Lock =0;

**Case-1:**

* Process can go simultaneously into the critical section and do their assigned tasks.
* There will be no problem and no interference of the two or more than two processes in this case.
* No other process come into the critical section if there is any process available into the critical section.
* Mutual exclusion will be there.

GUI FOR ALGORITHM



**Case-2:**

* In this case two process can go into the critical section at a same time but there is one condition which has to fulfil by two or more than two processes.
* In this scenario any process complete its first step of the whole algorithm and in position to start the second one. If swapping comes in between two process and second process also completes its first step. Then there is chance to intersection of first and second process into the critical section.
* Mutual exclusion is not there in this case.

GUI FOR ALGORITHM

