

Course: Operating System	Class Activity	NU Id:
Course Code: CS-2006		Date: 20/03/2025

P0

```
while (true) {
    flag[i] = true;
    turn = j;
    while (flag [j] && turn == j);

    /* critical section */

    flag[i] = false;

    /* remainder section */
}
```

P1

```
while (true) {
    flag[j] = true;
    turn = i;
    while (flag [i] && turn == i);

    /* critical section */

    flag[j] = false;

    /* remainder section */
}
```

Dry-Run Table:

Fill in the values of **turn**, **flag [i]**, and **flag [j]** at each step. Track when each process enters and exits the critical section.

Step	Process	Action	turn	flag[i]	flag[j]	Remarks
1	P0	flag[i] = true				
2	P0	turn = j				
3	P0	Check flag[j] && turn==j				
4	P1	flag[j] = true				
5	P1	turn = i				
6	P1	Check flag[i] && turn==i				
7		Enters critical section				
8		Exits critical section				
9		Enters critical section				
10		Exits critical section				

DRYRUN FOR PETERSON:- Code same as previous page. P_0 ; P_1

Step	Process	Action	turn	flag[i]	flag[j]	Remarks
1	P_0	flag[i] = true	$P_0 (i)$	true	false	P_0 sets its flag to true.
2	P_0	turn = j	j	true	false	Turn assigned to P_1 .
3	P_0	check flag[j] & turn = j	j	true	false	true & false \rightarrow false $\rightarrow P_0$ will enter CS.
4	P_1	flag[j] = true	j	true	true	P_1 sets its flag to true
5	P_1	turn = i	i	true	true	Turn assigned to P_0 .
6	P_1	check flag[i] & turn = i	i	true	true	true & true \rightarrow True $\rightarrow P_1$ will do busy waiting.
7	P_0	Enters critical section	i	true	true	P_0 exists, resets flag[i] = false
8	P_0	Exits critical section	i	false	true	P_0 enters critical section
9	P_1	Enters critical section	i	false	true	P_1 enters critical section
10	P_1	Exits critical section	i	false	false	P_1 exists, resets flag[j] = false