Lecture 9 – COMP 304 Handout by Didem Unat

1) Attempt 1: Only 2 processes, P_0 and P_1

```
//Pi:
//turn =>shared variable, initially 0
//turn = i => Pi can enter its critical

do {
        while ( turn != i );
        /* critical section */
        turn = j;
        /* remainder section */

} while(TRUE);
```

```
//Pj:
//turn =>shared variable, initially 0
//turn = j => Pj can enter its critical

do {
        while ( turn != j );
        /* critical section */
        turn = i;
        /* remainder section */

} while(TRUE);
```

2) Attempt 2: Shared variables

- boolean flag[2];
  //initially flag[0] = flag[1] = false.
- flag[i] = true $\Rightarrow P_i$ ready to enter its critical section

```
do {
        flag[i] = true;
        while (flag[j]) ;
        //critical section

        flag [i] = false;

       //remainder section
} while (true);
```

3) Attempt 3: Shared variables

```
bool flag[0] = false;
bool flag[1] = false;
int turn;
```

```
P0: flag[0] = true;
P0: turn = 1;

while (flag[1] && turn == 1) { // busy wait }
// critical section

flag[0] = false;

//remainder section
```

```
P1: flag[1] = true;
P1: turn = 0;

while (flag[0] && turn == 0) { // busy wait }
// critical section

flag[1] = false;

//remainder section
```