

1)

```
int mivalor = 1;
while(mivalor == 1 ){
    while(lock ==1 );
    Exchange(lock, mivalor);
}
```

lock = 0

2)

a)

```
<turn[i] = number; number = (number%n )+ 1;>
while(turn[i] != next);
next = next%N + 1;
```

b)

```
int number = 1;
int next = 1;
int Fetch_Add(int var, int incr){
    int tmp = var;
    var = var + incr;
    return tmp;
}
turn[i] = Fetch_Add(number, 1);
if(turn[i] == N + 1)
    turn[i] = Fetch_Add(number, -N);
if(turn[i] > N){turn[i]-=N;}

while(turn[i] != next);

next = next%N + 1;
```

3)

```
int arrive[n] = ([n] 0);
```

```
int continue[n]=([n] 0);
```

```
process Worker[i=1 to n] {
```

```
  while (true)
```

```
  { // iteracion proceso i
```

```
    arrive[i] = 1;
```

```
    while (continue[i]!=1); <await (continue[i] == 1)
```

```
    continue[i]=0;
```

```
  }
```

```
}
```

```
process coord
```

```
while(true){
```

```
  // 1º ESPERO QUE LLEGUEN TODOS
```

```
  for[i = 1 to n]{
```

```
    while(arrive[i] == 0)
```

```
      ;
```

```
    arrive[i] = 0;
```

```
  }
```

```
  // 2º INDICO QUE ARRANQUEN
```

```
  for[i = 1 to n]
```

```
    continue[i] = 1;
```

```
}
```

4)

```
void barrera(int id){

    int d = 1 ;

    int waitId          //id con el q tengo q esperar

    while(d < N){

        arrive[id] = arrive[id]+1;

        waitId = (id + d -1)%N+1;

        d = d+d;

        <await (arrive[waitId] >= arrive[id])>

    }

}
```