

```

// *****
// REALIZZAZIONE DEI MONITOR
// *****
typedef struct
{
    semaphore sem; // Inizializzato a 0
    int cont;
} condition;

semaphore mutex = 1;

// *****
// SIGNAL AND CONTINUE
// *****
void wait(condition cond, semaphore mutex)
{
    cond.cont++;
    V(mutex);
    P(cond.sem);
    P(mutex);
}

void signal(condition cond)
{
    if (cond.cont > 0)
    {
        cond.cont--;
        V(cond.sem);
    }
}

void op()
{
    P(mutex);
    // ...
    V(mutex);
}

// *****
// SIGNAL AND WAIT
// *****
void wait(condition cond, semaphore mutex)
{
    cond.cont++;
    V(mutex);
    P(cond.sem);
}

void signal(condition cond)
{
    if (cond.cont > 0)
    {
        cond.cont--;
        V(cond.sem);
        P(mutex);
    }
}

```

```

void op()
{
    P(mutex);
    // ...
    V(mutex);
}

// *****
// SIGNAL AND URGENT WAIT
// *****

semaphore urgent = 0;
int urgentCont = 0;

void wait(condition cond, semaphore mutex)
{
    cond.cont++;
    if (urgentCont > 0)
        V(urgent);
    else
        V(mutex);
    P(cond.sem);
    cond.cont--;
}

void signal(condition cond)
{
    if (cond.cont > 0)
    {
        urgentCont++;
        V(cond.sem);
        P(urgent);
        urgentCont--;
    }
}

void op()
{
    P(mutex);
    // ...
    if (urgentCont > 0)
        V(urgent);
    else
        V(mutex);
}

// *****
// SIGNAL AND RETURN
// *****

semaphore urgent = 0;
int urgentCont = 0;

void wait(condition cond, semaphore mutex)
{
    cond.cont++;
    V(mutex);
    P(cond.sem);
    cond.cont--;
}

```

```
void signal(condition cond)
{
    if (cond.cont > 0)
    {
        V(cond.sem);
    }
    else
        V(mutex);
}

void op()
{
    P(mutex);
    // ...
    << se c'è stata una signal >> signal(cond);
    << altrimenti >> V(mutex);
}
```