ΣΕΙΡΑ ΕΡΓΑΣΙΩΝ 2

ΒΑΣΙΛΕΙΟΣ ΚΥΡΙΤΣΗΣ 2999

ΑΛΕΞΑΝΔΡΟΣ ΣΤΑΜΟΥΛΟΣ

2.1 Δυαδικοί σηματοφόροι

- typedef struct {
- int semid; //
- int init; // σήμα για την αρχικοποίηση του σηματοφόρου
- struct sembuf op;
- }
- Για την κλήση up ελέγχουμε την τιμή του σηματοφόρου, τον αυξάνουμε μόνο όταν είναι μηδέν.

2.2 Αναγνώριση πρώτων αριθμών

```
mysem_t *t_terminate,mutex,full,empty;
init(mutex, 1), init(full, 0), init(empty, 1)
Init(t_terminate[i] ,0)  για κάθε i.
```

```
Main thread {
    while(input != -1) {
       input = get_value();
       down(empty);
       down(mutex);
       send value
       up(mutex)
       up(full)
    }
    send signal to workers to terminate
}
```

```
Worker thread{
    while(!terminate) {
       down(full);
       down(mutex);
       get value
       up(mutex);
       up(empty);
       process the value
    }
}
```

2.3 Στενή γέφυρα

```
Car thread {
    if you can cross the bridge go ahead
    else sleep until someone wakes you up

    cross the bridge

    if you can wake up someone in your direction wake him
    else if you can wake up someone in your direction wake him
    else if nobody is waiting reset the bridge
}
```

2.4 Τρενάκι

```
int boarded = 0, unboarded = 0;
mysem_t board, depart, unboard, arrive, mutex1, mutex2;
init(mutex1, 1), init(mutex2, 1), init(board, 0), init(unboard, 0), init(depart, 0), init(arrive, 0)
```

```
Passenger thread {
    down(board);
    down(mutex1);
    boarded++;
    if last passenger
          up(depart);
          board = 0;
    else
          up(board);
    up(mutex1);
    down(arrive);
    down(mutex2);
    unboarded++;
    if last to unboard from the train
          up(unboard);
          unboarded = 0;
    else
          up(arrive);
   up(mutex2);
}
```

```
Roller coaster thread {
    load passengers
    up(board);
    down(depart);

    depart-run-arrive

    up(arrive);
    unload passengers
    down(unboard);

}
```