
Λειτουργικά Συστήματα Ι

Τομέας Λογικού των Υπολογιστών

Δούκας Ιωάννης
5509
douka@ceid.upatras.gr

Νικολάου Γλαύκος
5693
nikolaog@ceid.upatras.gr

1 Στόχος της Άσκησης

Στόχος της άσκησης είναι η εξοικείωση με θεμελιώδεις έννοιες και μηχανισμούς που προσφέρει το λειτουργικό σύστημα. Για το σκοπό αυτό κάνουμε δύο διαφορετικές υλοποιήσεις. Η μία βασίζεται στη δημιουργία και διαχείριση διεργασιών (processes) και την επικοινωνία /συγχρονισμό μεταξύ τους με IPC (Interprocess Communication), όπως shared memory, semaphores, signals κ.τ.λ. Η άλλη υλοποίηση γίνεται με τη χρήση νημάτων (threads), και το συγχρονισμό μεταξύ τους για την σωστή διαχείριση της εκ των πραγμάτων κοινής τους μνήμης με mutexes και condition variables. Και οι δύο υλοποιήσεις χρησιμοποιούν sockets για την επικοινωνία client-server

2 Γενική Περιγραφή

Στην παρούσα εργασία υλοποιήσαμε έναν server για ένα multiplayer online game με συγκεκριμένο αριθμό παιχτών. Κάθε καινούριος παίχτης που συνδέεται επιλέγει στοιχεία για το αρχικό του inventory. Μόλις συμπληρωθεί ο προκαθορισμένος αριθμός παιχτών ξεκινάει μία καινούρια παρτίδα η οποία θα εκτελεστεί παράλληλα με τις προηγούμενες είτε μέσω διεργασίας (fork) ή μέσω νημάτων (threads). Για κάθε παρτίδα υπάρχει ένα socket μέσω του οποίου μπορούν οι παίχτες να επικοινωνούν μεταξύ τους και να ανταλλάσσουν μηνύματα. Η παρτίδα τελειώνει μόλις και ο τελευταίος παίχτης (της παρτίδας) αποχωρίσει από το παιχνίδι.

3 Περιγραφή Server-Client μοντέλου

Και στις δύο υλοποιήσεις, με διεργασίες και με νήματα, έχουμε χρησιμοποιήσει το μοντέλο server-client.

3.1 Server

Ο server εκτελείται ως εξής: `./gameserver -p <num_of_players> -i <game_inventory> -q <quota_per_player>` Η πρώτη παράμετρος είναι ένας integer που δηλώνει τον αριθμό παικτών ανά παρτίδα και η δεύτερη παρέχει το filename με το συνολικό απόθεμα (inventory) ανά παρτίδα. Η τρίτη παράμετρος (quota) καθορίζει το μέγιστο μέγεθος inventory που αναλογεί σε κάθε παίχτη. Και οι τρεις παράμετροι παραμένουν ίδιες για όλες τις παρτίδες που θα ξεκινήσει αυτό το instance του server. Για κάθε παρτίδα το inventory είναι αναλώσιμο. Για να δοθούν κάποια αντικείμενα σε κάποιο παίχτη πρέπει να είναι διαθέσιμα. Στην περίπτωση που κάποιο από τα αντικείμενα τα οποία ζητάει ο παίκτης είναι άγνωστο στον server ή αν έχει ήδη εξαντληθεί, ή αν κάποιος παίκτης ζητάει περισσότερα αντικείμενα από το quota του ο server θα του επιστρέψει ένα μήνυμα λάθους και θα διακόψει άμεσα τη σύνδεση με τον συγκεκριμένο παίκτη. Μέχρι να συγκεντρωθεί ο απαιτούμενος αριθμός παικτών για να ξεκινήσει μία παρτίδα, οι ήδη συνδεδεμένοι παίκτες λαμβάνουν ένα μήνυμα αναμονής κάθε 5 sec. Για κάθε καινούρια παρτίδα το συνολικό inventory του server αρχικοποιείται στις αρχικές τιμές. Η έναρξη της παρτίδας στους παίκτες σηματοδοτείτε λαμβάνοντας από τον server το μήνυμα "START". Κατά την διάρκεια μίας παρτίδας ότι μήνυμα στέλνεται από κάποιον παίκτη στον server, προωθείται από τον server προς όλους τους υπόλοιπους παίκτες της ίδιας παρτίδας. Ο gameserver τερματίζεται με Ctrl-C.

3.2 Client

Στη συγκεκριμένη περίπτωση οι clients οι οποίοι θα συνδεθούν στον server είναι οι παίκτες, οι οποίοι εκτελούνται ως εξής: `./player -n <name> -i <inventory> <server_host>` Η πρώτη παράμετρος δηλώνει το όνομα του παίχτη, η δεύτερη το inventory που επιλέγει ο παίκτης για τον εαυτό του, και η τρίτη το hostname όπου τρέχει ο gameserver. Κάθε παίκτης δικαιούται να ζητήσει μέχρι quota αντικείμενα για το δικό του inventory.

3.3 Επικοινωνία Server-Client

Η επικοινωνία μεταξύ παίχτη και server γίνεται με TPC, και το πρωτόκολλο επικοινωνίας είναι σε ASCII. Συγκεκριμένα ο, ο παίχτης ανοίγει μία σύνδεση TCP με τον server, και στέλνει μερικές γραμμές με ASCII δεδομένα, μέσω ενός socket. Η πρώτη γραμμή περιέχει απλά το όνομα του, και οι επόμενες τα αντικείμενα που επιλέγει. Κατόπιν, στέλνει μία κενή γραμμή, και αυτό σηματοδοτεί το τέλος της αίτησης συμμετοχής του. Ο server απαντά με "OK" ή με σχετικό μήνυμα λάθους, ακολουθούμενο από \n.

4 Αναλυτική Υλοποίηση με βάση νήματα

4.1 Περιγραφή Υλοποίησης

Για την υλοποίηση με βάση τα νήματα έχουμε χρησιμοποιήσει τις εντολές `pthread_create()` για την δημιουργία νημάτων, `pthread_mutex_lock()` και `pthread_mutex_unlock()` για τον συγχρονισμό τους. Αρχικά ο server ελέγχει τις παραμέτρους που δόθηκαν από τη γραμμή εντολών, ανοίγει το socket μέσω του οποίου θα συνδεθούν οι clients και περιμένει για αιτήματα. Η αποδοχή των clients γίνεται μέσα σε έναν ατέρμων βρόχο.

Κάθε καινούριος παίχτης ο οποίος συνδέεται είναι και ένα καινούριο νήμα για τον server. Για κάθε παίχτη κρατάμε σε ένα πίνακα τα sockets descriptors για την μελλοντική τους επικοινωνία στο τραπέζι. Το νήμα εκτελεί μία συνάρτηση η οποία αναλαμβάνει την επικοινωνία με τον client ο οποίος το ενεργοποίησε και τερματίζεται κατά την αποσύνδεση του. Η συνάρτηση αυτή λαμβάνει το αίτημα από τον client και ελέγχει να δει αν μπορεί να του παρέχει το inventory το οποίο ζητάει. Εάν δεν μπορεί του επιστρέφει αντίστοιχο μήνυμα και τερματίζεται η σύνδεση και το αντίστοιχο thread. Στην περίπτωση όπου γίνει αποδεκτό το αίτημα του, για το τρέχον τραπέζι το οποίο περιμένει να γεμίσει, αυξάνονται οι αντίστοιχοι μετρητές (πόσα τραπέζια, πόσοι παίχτες) και περιμένει ο παίχτης μέχρις ότου να γεμίσει το τραπέζι του λαμβάνοντας ανά πέντε δευτερόλεπτα ένα μήνυμα αναμονής.

Το κάθε τραπέζι για τον server σηματοδοτείται από έναν αριθμό. Για κάθε παίχτη έχουμε μία δομή στην οποία κρατάμε το όνομα του παίχτη και το τραπέζι στο οποίο ανήκει. Κατά την διάρκεια του τραπέζιου χρησιμοποιούμε την δομή αυτή για να ελέγχουμε όταν λαμβάνουμε κάποιο μήνυμα από κάποιο παίχτη σε ποιους θα πρέπει να σταλεί.

4.2 Παράδειγμα Εκτέλεσης

Παρακάτω παρουσιάζεται παράδειγμα εκτέλεσης. Στο συγκεκριμένο παράδειγμα έχουμε ανοίξει τον server: `./server -p 2 -i inv_server.txt -q 10` και 4 με ονόματα giannis, glafkos, spyros και andreas.

```
./server -p 2 -i testing/inv_server.txt -q 10
```

For exit press Ctr-C

Starting inventory:

gold	10
armor	50
ammo	40
lumber	20
magic	15
rock	50

Quota per player: 10

Players per table: 2

Message from client:

giannis

gold	1
------	---

rock	1
------	---

+----- NEW TABLE -----+

1 players in table 1

New inventory for 1 table

gold	9
armor	50
ammo	40
lumber	20
magic	15
rock	49

Message from client:

glafkos

gold	1
------	---

rock	1
------	---

2 players in table 1

New inventory for table 1

gold	8
armor	50

ammo 40
lumber 20
magic 15
rock 48

Message from client:

spyros

gold 1

rock 1

+----- NEW TABLE -----+

3 players in table 2

New inventory for 2 table

gold 9

armor 50

ammo 40

lumber 20

magic 15

rock 49

Message from client:

andreas

gold 1

rock 1

4 players in table 2

New inventory for table 2

gold 8

armor 50

ammo 40

lumber 20

magic 15

rock 48

- giannis from table 1 said: hello

- glafkos from table 1 said: hi

- glafkos from table 1 said: my name is glafkos
- giannis from table 1 said: my name is giannis
- giannis from table 1 said: bye
- glafkos from table 1 said: bey

Client giannis has left from the table (1)

Client glafkos has left from the table (1)

- spyros from table 2 said: hello
- spyros from table 2 said: my name is spyros
- andreas from table 2 said: hi my name is andreas
- andreas from table 2 said: bye
- spyros from table 2 said: bye

Client spyros has left from the table (2)

Client andreas has left from the table (2)

^C

```
./client -n giannis 2 -i testing/inv_client.txt localhost
```

giannis

```
gold      1
rock      1

OK
Server 's answer  : Please wait
Server 's answer  : Please wait
Server 's answer  : START
hello
- giannis: hello

- glafkos: hi

- glafkos: my name is glafkos

my name is giannis
- giannis: my name is giannis

bye
- giannis: bye

- glafkos: bey

^C
```

```
./client -n glafkos 2 -i testing/inv_client.txt localhost
```

```
glafkos
gold      1
rock      1

OK
Server 's answer  : START
- giannis: hello

hi
- glafkos: hi
```

```
my name is glafkos
- glafkos: my name is glafkos

- giannis: my name is giannis

- giannis: bye

bey
- glafkos: bey

- Message from server: giannis has left from the table
^C
```

```
./client -n andreas 2 -i testing/inv_client.txt localhost

andreas
gold      1
rock      1

OK
Server 's answer : START
- spyros: hello

- spyros: my name is spyros

hi my name is andreas
- andreas: hi my name is andreas

bye
- andreas: bye

- spyros: bye

- Message from server: spyros has left from the table
^C
```



```
./client -n spyros 2 -i testing/inv_client.txt localhost
```

```
spyros
```

```
gold      1
```

```
rock      1
```

```
OK
```

```
Server 's answer : Please wait
```

```
Server 's answer : Please wait
```

```
Server 's answer : Please wait
```

```
Server 's answer : Please wait
```

```
Server 's answer : Please wait
```

```
Server 's answer : START
```

```
hello
```

```
- spyros: hello
```

```
my name is spyros
```

```
- spyros: my name is spyros
```

```
- andreas: hi my name is andreas
```

```
- andreas: bye
```

```
bye
```

```
- spyros: bye
```

```
^C
```