Connect4

Puiu Bogdan-Marian 2B4

Facultatea de Informatica iasi

1. Introducere
   1. Functionalitate

Jocul Connect4 este tipul de joc ce necesita minim 2 jucatori, in care fiecare plaseaza o pisa (fie ea rosie fie albastra) iar piesele cad in casuta cea mai de jos posibila din tabela. Primul jucator care isi aliniaza 4 piese orizontal,vertical sau pe diagonala , va castiga.

* 1. Tehnologii utilizte

Tehnologia utilizata in aceasta aplicatie pentru comunicarea intre client si server este de TCP concurent. (Transmission Control Protocol). Desi aceasta metoda de comunicare nu este la fel de rapida precum cea UDP(User Datagram Protocol) ,este mult mai sigura(lucru necesar deoarece mutarea trebuie sa ajunga in siguranta la server).Astfel , nu se poate tolera niciun fel de pierdere(nici la transmiterea tabelei de joc/mutarea aleasa de client) deoarece ar putea avea un impact devastator asupra jocului, influentant ce client va castiga.Se vor utiliza doar un socket pt a comunica cu ambii client. Ilustrarea de mai jos justifica alegerea facuta in favoarea tehnologiei TCP:

(1)

2. Arhitectura aplicatiei

2.1 Acceptarea clientilor

Deoarece este un joc care trebuie jucat in minim 2 clienti , serverul va putea porni jocul doar cand sunt 2 clienti acceptati. Dupa ce au fost acceptati , ambii clienti trebuie sa fie pregatiti pentru a incepe.Pentru a confirma ca sunt pregatiti trebuie sa introduce comanda “play”.

2.2 **Inceperea rundei**

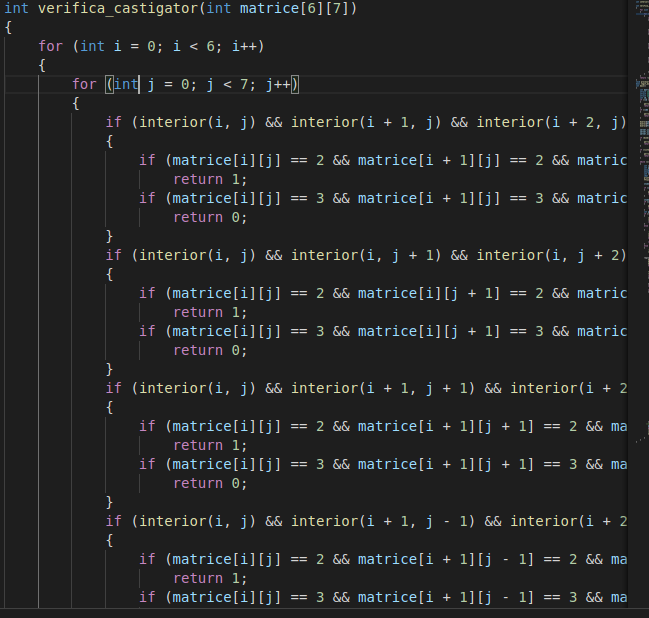
Odata ce jocul a pornit , fiecare client va avea tura sa pentru a face o mutare spre a castiga jocul. Daca acesta introduce un numar invalid(care nu se afla in multimea {1,2…,7}) tura acestuia nu se va modifica si va fi atentionat spre a face o mutare valida. Serverul va analiza daca mutarea este valida sau nu.De asemenea, le va asigna clientilor tura lor ( fie 1 sau 0) dupa care se va ghida tot jocul.

2.3 **Transmiterea tabelei**

Tabela jocului are la baza o matrice de 6x7 ce va fi transmisa fiecarui client dupa fiecare mutare. Clientul va avea fie culoarea rosie fie albastra cu care va completa tabela de joc.

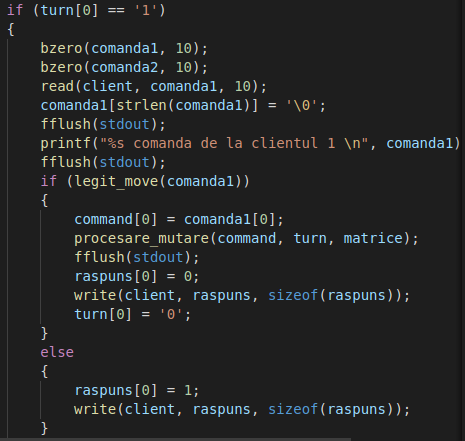
**2.4 Procesarea castigatorului**

Dupa fiecare mutare facuta serverul va verifica daca exista un jucator. Aceasta verificare o face cu ajutorul functiei verifica\_jucator(int matrice[6][7]) ce va returna 1 sau 0 daca exista un castigator.In caz contrar va returna -1. In matrice o mutare de a primului client va fi notate cu 2 iar a celuilat cu 3.



2.4 Procesarea turn-ului clientilor

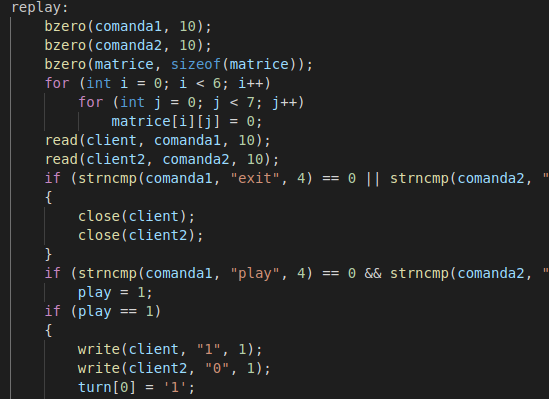
Deoarece serverul nu poate face distinctia usor intre cei doi clienti se va utiliza tura ce le a fost asignata la inceputul jocului. Astfel in functie de acea variabila vor muta piesele pe rand. Daca un client introduce o variabila cand nu este tura lui , se va realiza un “pre-move” iar atunci cand va fi randul sau se va procesa comanda introdusa precedent ,dand skip turei sale.

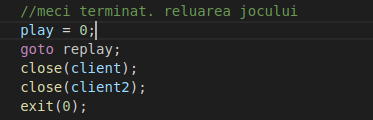
Daca comanda introdusa nu este valida , tura nu se va

schimba.

2.5 Reinceperea unei runde noi

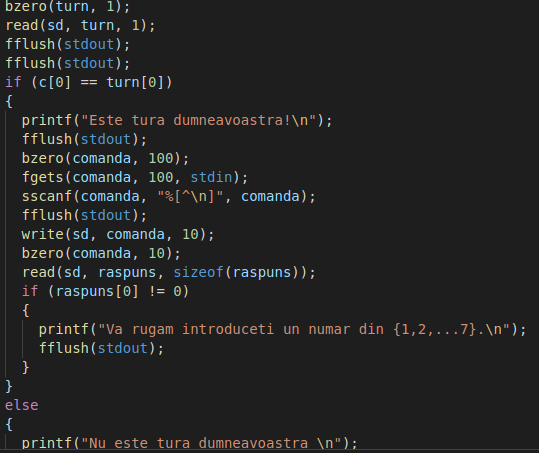
Inainte de a incepe jocul avem punctual de ‘replay’. Dupa ce jocul s-a terminat si a fost ales un castigator , se va face un “goto replay;” pt ca juca din nou(bineinteles daca ambii client doresc asta). Daca un client doreste sa iasa folosing comanda “exit” si celalalt client va fi fortat sa faca la fel.





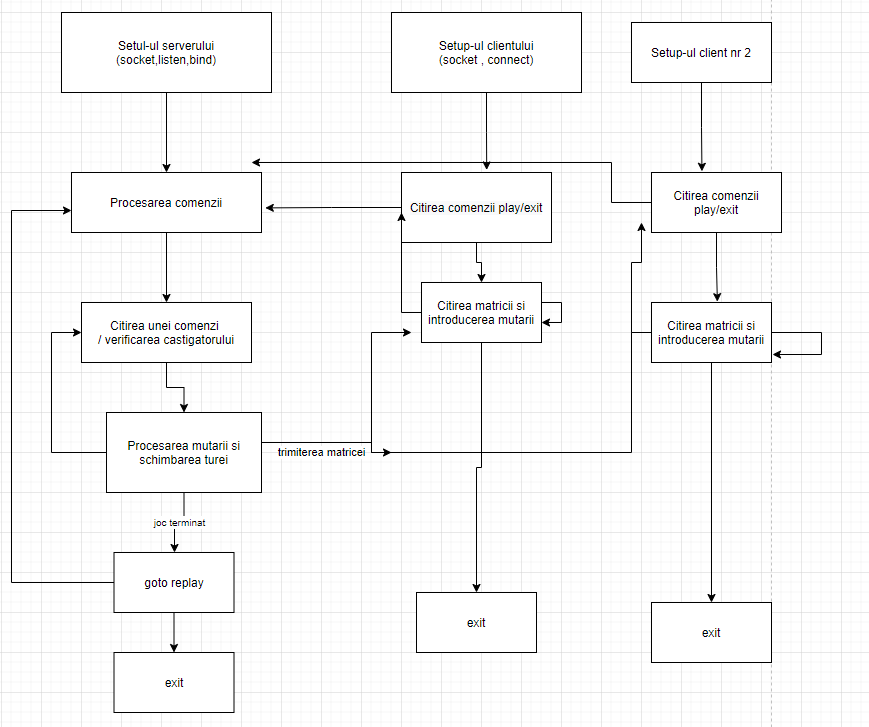
2.6 Task-urile clientului

Asa cum era de asteptat , clientul nu trebuie sa faca decat un minim de efort. Acesta doar va citi tabela , va analiza mesajul serverului daca este castigator si comanda pe care o primeste de la jucator. In rest se va baza pe server pentru a analiza chestiuniile logice ale jocului.



1. Implementarea

Deoarece este un server TCP concurent , se pot juca mai multi client in paralel, singura conditie fiind ca numarul lor sa fie multiplu de 2. Astfel serverul dupa ce primeste doi client , jocul se va derula intr-un proces copil , cu ajutorul primitivei fork(). Chiar daca se joaca simultan mai multi jucatori , tabela sau mutarile lor nu vor fi afectate. Serverul este mult mai eficient putand sa proceseze mai multi clienti , fapt ce il face un server TCP concurent.



1. Concluzii

Aplicatia poate fi imbunatatita prin a face un gamemode cu mai mult de 2 clienti , eventual 2vs2 pe o tabla mai mare. De asemenea , ar putea avea un mod de singleplayer prin care ar putea juca doar un singur client doar cu server-ul.

References

1. https://www.reddit.com/r/ProgrammerHumor/comments/9gcwgw/tcp\_vs\_udp/
2. https://profs.info.uaic.ro/~gcalancea/laboratories.html
3. https://profs.info.uaic.ro/~gcalancea/lab7/servTcpConc.c
4. https://profs.info.uaic.ro/~gcalancea/lab7/servTcpConc