

Progetto Reti di calcolatori

Progetto 2 - Food Delivery



De Rosa Salvatore 0124001981
De Leo Mario 0124001945

DESCRIZIONE DEL PROGETTO :

Il cliente appena registrato, collegandosi al server, riceve la lista dei ristoranti (questi ultimi collegandosi al server), una volta scelto il ristorante, il server, inoltrerà la richiesta di menu al ristorante, il ristorante, in risposta inoltrerà il menu lo inoltrerà al cliente, dopo che il cliente avrà scelto il piatto e le bevande, subentrerà il rider, che completerà l'ordine il server infine, manderà una notifica positiva che avrà completato l'ordine.

TECNOLOGIE UTILIZZATE:

Si è deciso di utilizzare il linguaggio C, implementando la comunicazione socket, e utilizzando come gestore di IO uno schema IO/Multiplexing (tramite la select)

DESCRIZIONE SCHEMI DELL'ARCHITETTURA :

Lo scopo del progetto è di implementare un servizio di food delivering, dove prevede le seguenti entità :

- Server :

Fondamentale, perché gestisce tutte le comunicazioni e rimarrà sempre attivo, egli farà uso dello schema I/O multiplexing.

- Cliente :

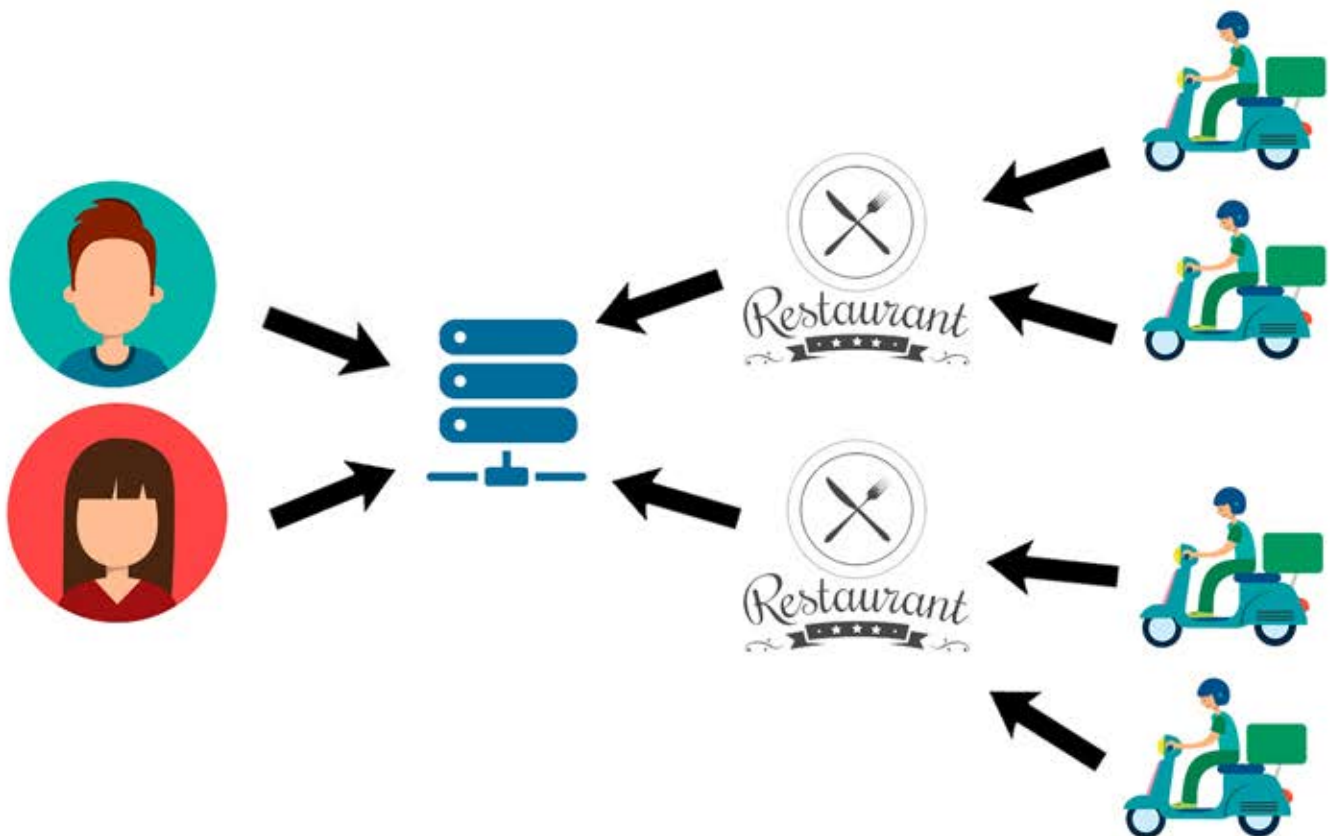
Colui che, una volta effettuato l'accesso e collegatosi al server, richiederà un servizio (o più precisamente un ordine) al ristorante.

- Ristorante :

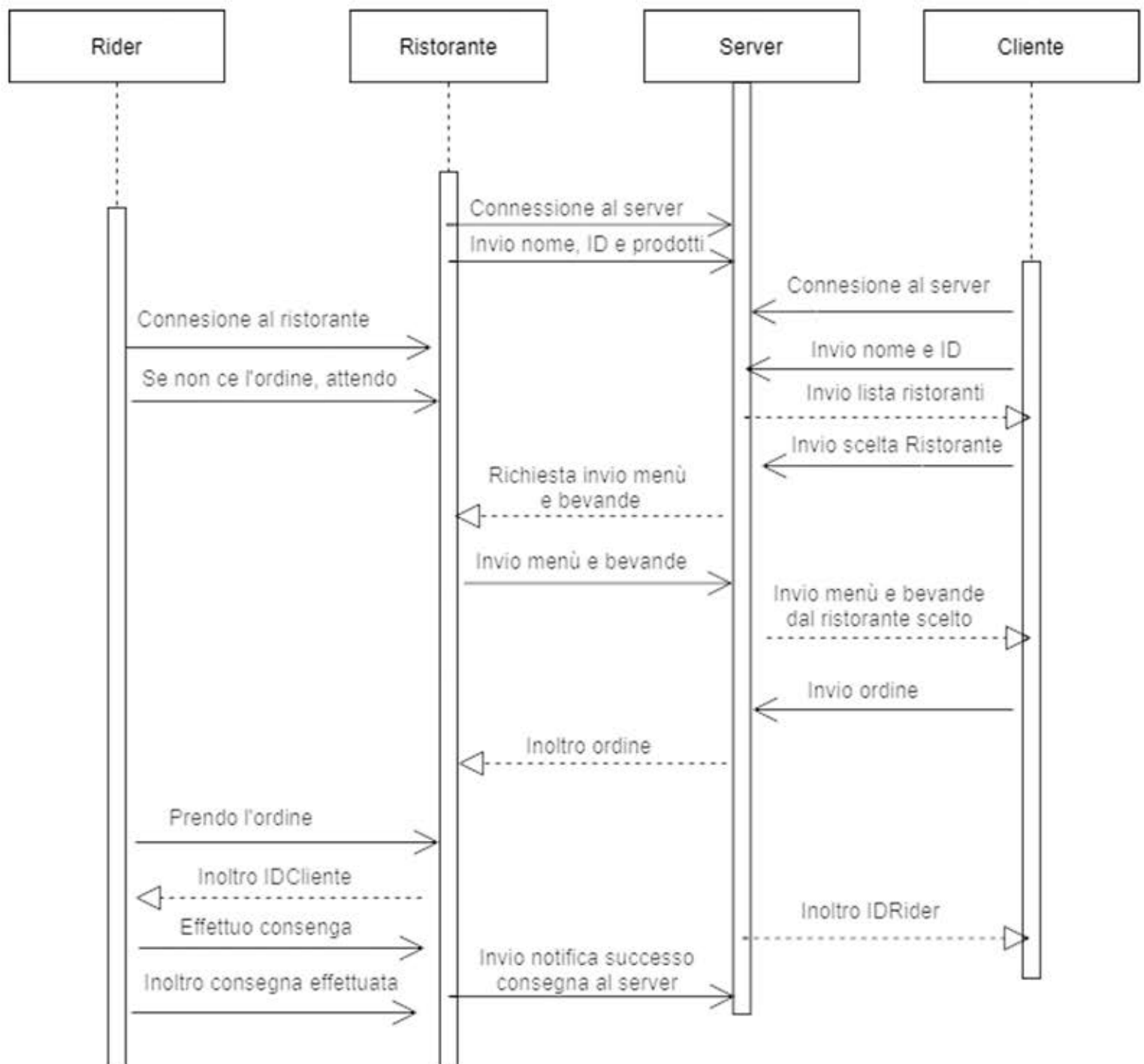
Colui che fornisce i servizi e le richieste dei clienti, e inoltre gestirà i più rider, anche lui farà uso dello schema I/O multiplexing.

- Rider :

Il rider, connettendosi nell'apposito ristorante, si occuperà di consegnare i prodotti ordinati dal cliente.



DESCRIZIONE E SCHEMI DEL PROTOCOLLO APPLICAZIONE



SERVER :

Il server attende nuove connessioni, sia da clienti che da ristoranti, come già detto in precedenza il server utilizza l'i/o multiplexing per la gestione dell'I/O e ogni volta che si sblocca dalla select, effettuerà un controllo per vedere se ci sono altri clienti in attesa di una richiesta di ricezioni dei piatti, e quindi poter comunicare.

Il server, inoltre comunica efficientemente attraverso degli identificativi

[0]->I ristoranti si connettono

[1]->I clienti si connettono

[2]->Il server aggiungera una richiesta

[3]->Il server richiede i prodotti ai ristoranti

[6]->Il server richiede le bibite ai ristoranti

[4]->Il server inoltrerà le scelte effettuate al cliente

[5]->Il server riceve gli ID dei clienti e rider, li inoltra a quest'ultimi e completa la consegna

[8]->Il cliente dopo che ha terminato l'ordine si disconnette, utilizza FD_CLR per rimuovere dall'insieme di descrittori FD_SET ponendo a 0 il bit corrispondente

[9]->Nell'eventualità che un ristorante decida di disconnettersi, utilizza FD_CLR per rimuovere dall'insieme di descrittori FD_SET ponendo a 0 il bit corrispondente

RISTORANTI :

Il ristorante anch'egli userà il multiplexing per poter comunicare in modo efficiente con i rider, il ristorante si crea attraverso una fgets, stesso procedimento per il menu e bibite, dopodichè si connette al server, inviando 0, appena il cliente effettuerà l'ordine vorrà dire che dovrà inoltrare la richiesta al primo rider disponibile.

IDENTIFICATORI :

- [0]-> Mando i miei dati
- [8]-> Mando i prodotti
- [6]-> Mando le bevande
- [9]-> Ricevo gli id dei prodotti scelti dal cliente
- [0]-> Effettuo consegna

CLIENTI :

I clienti attraverso un sistema di registrazione e login, si conatteranno al server, inviando 1, dove, il server, invierà la lista dei ristoranti disponibili, dopodichè sceglierà i piatti e le bibite che il ristorante offrirà, effettuato l'ordine attenderà l'ID del rider che effettuerà la consegna

- [2]-> Invia 2 perche vuole vedere i ristoranti attivi, dopodichè effettuerà gli ordini
- [4]-> Invia i suoi prodotti scelti al server
- [5]-> Riceve l'ID del rider e si disconnette

RIDER :

I Rider si conatteranno al ristorante mandando 0, una volta connessi, controlleranno se ci sono ordini, se ci sono, digitando 1 prenderanno il compito di consegnare l'ordine, altrimenti, premendo 5 si aggiorna, dopo aver preso l'ordine se nessun altro rider lo ha preso, lo completa e si disconnette altrimenti se un altro rider ha già preso l'ordine aspetterà

DESCRIZIONE STRUCT E LISTE

Ristorante :

```
struct ristorante {
    char name[max];
};
typedef struct ristorante rist;
struct prodotto {
    char nome[max];
};
typedef struct prodotto Prodotto;
struct bibite {
    char nomeb[max];
};
typedef struct bibite Bibite;
```

Queste strutture servono per la creazione dei ristoranti, menù e delle bibite.

Queste strutture si trovano sia nei ristoranti sia nel server e sia nel cliente.

```
struct ordine {
    int stato_Ord;
    struct ordine * next;

}* p_punt, * head, * fine;
typedef struct ordine Ordine;
```

Questa lista è esclusiva del ristorante, il suo compito è quello di tenere traccia degli ordini

Cliente :

```
struct cliente {  
    char uname[max];  
};  
typedef struct cliente Cliente;
```

Questa struttura serve per creare il cliente, si trova nel cliente e nel server

Server :

```
struct Server_Ristoranti {      //Lista dei ristoranti  
    rist ristorante;  
    int descriptor;  
    struct Server_Ristoranti * next;  
};  
typedef struct Server_Ristoranti SERVER_RISTORANTI;  
struct listaclienti      //Lista dei clienti  
{  
    Cliente clienti;  
    int descriptor;  
    struct listaclienti *next;  
};  
typedef struct listaclienti LISTACLIENTI;  
  
struct lista //Questa struct serve a gestire le richieste effettuate  
{      //rispettivamente non solo dai clienti ma anche dai ristoranti  
//tramite i loro FD  
    int FDC;  
    rist ristorante;  
    Prodotto prodotti;  
    int FDR;  
    int Stato_Operazione;  
    struct lista *next;  
    struct lista *prec;  
};  
typedef struct lista LISTA;
```

Queste liste sono esclusive del server.

FUNZIONI

Funzioni Cliente:

```
void signup();  
void account_check();  
void login();  
int cerca_ristoranti();  
void stampaMenu(int clientsocket, int numProdotti);  
void stampaBevande(int clientsocket, int numBibite);
```

Funzioni Ristorante:

```
void insl_testa(Ordine ** head, Ordine ordine);  
void eliminazione_ordine(Ordine **p_testa);
```

Funzioni Server:

```
SERVER_RISTORANTI trova_ristoranti(SERVER_RISTORANTI * testa, int posizione);  
LISTA * trovarichiasta(LISTA ** head, int fd, int identificativo, int stato);  
void ins_testa(rist info_ristoranti, struct lista **p_testa, int fd);  
void aggiungi_Ristoranti(SERVER_RISTORANTI ** testa, rist ris, int fd);  
void aggiungiC(LISTACLIENTI ** testa2, Cliente cli, int fd);  
void aggiungere_Richiasta(int client, int ristorante);  
void eliminazione_ristorante(SERVER_RISTORANTI **p_punt, int id);  
void eliminazione_cliente(LISTACLIENTI **p_punt, int id);  
void eliminazione_richiasta(LISTA ** punt);
```

GUIDA ALLE SORGENTI:

Server.c:

File sorgente del server.

Ristorante.c e Ristorante2.c

File sorgenti dei ristoranti.

Rider.c,Rider2.c e Rider3.c,Rider4.c

File sorgenti dei rider

Cliente.c e Cliente2.c

File sorgenti dei clienti

Wrapper.h

Header File contenente molte funzioni "wrapper"

ISTRUZIONI PER L'ESECUZIONE

Per poter comunicare in modo efficace si consiglia di effettuare le connessioni nel seguente ordine:

0)Server

1)Ristorante

2)Cliente

3)I due rider dell'apposito ristorante scelto

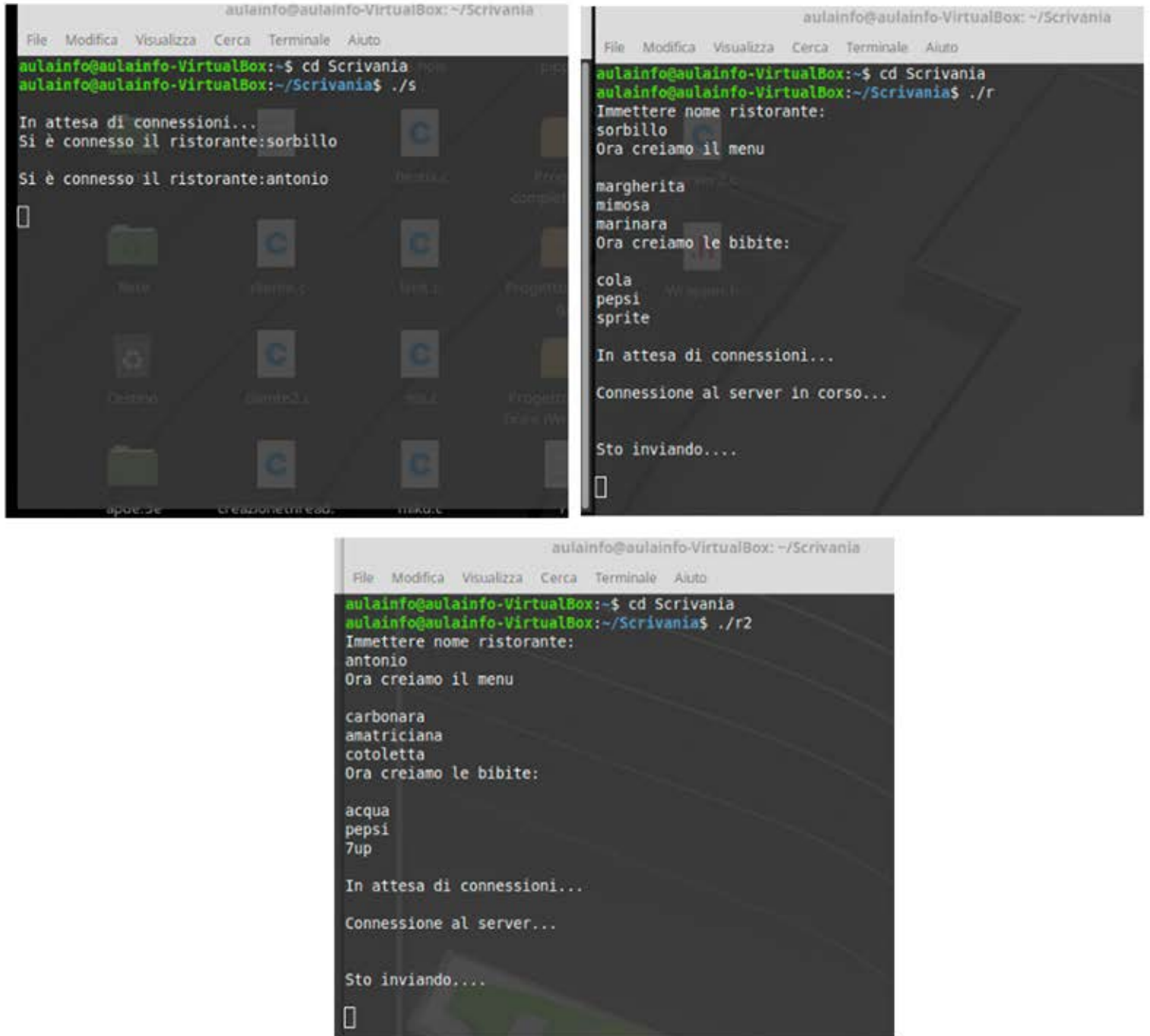
Per far partire i sorgenti non dobbiam far altro che digitare dal terminale
cd Scrivania

gcc -o "nome eseguibile" file.c

e per eseguirli

./"nome eseguibile"

ESECUZIONE:



```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./s

In attesa di connessioni...
Si è connesso il ristorante:sorbillo
Si è connesso il ristorante:antonio

In attesa di connessioni...
Immettere nome ristorante:
sorbillo
Ora creiamo il menu

margherita
mimosa
marinara
Ora creiamo le bibite:

cola
pepsi
sprite

In attesa di connessioni...
Connessione al server in corso...

Sto inviando....

aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./r2
Immettere nome ristorante:
antonio
Ora creiamo il menu

carbonara
amatriciana
cotoletta
Ora creiamo le bibite:

acqua
pepsi
7up

In attesa di connessioni...
Connessione al server...

Sto inviando....
```

Avvio del server, e creazione ed invio dei dati dei ristoranti

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./c
BENVENUTO AL FOOD DELIVERY DI SALVATORE DE ROSA E MARIO DE LEO!
*****
* MATRICOLE: *
* 0124001981->SALVATORE DE ROSA *
* 0124001945->MARIO DE LEO *
*****
IMMETTERE UNA DELLE SEGUENTI OPERAZIONI

1)REGISTRAZIONE
2)ACCESSO
3)USCITA

Fai la tua scelta! 1
Digita il tuo nome: mario
```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
* 0124001981->SALVATORE DE ROSA *
* 0124001945->MARIO DE LEO *
*****
IMMETTERE UNA DELLE SEGUENTI OPERAZIONI

1)REGISTRAZIONE
2)ACCESSO
3)USCITA

Fai la tua scelta! 2
LOGIN

Digita il tuo nome: mario
```

Registrazione e login del cliente

```
vm [in esecuzione] - Oracle VM VirtualBox
File Macchine Visualizza Inserimento Dispositivi Aiuto

aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./s

In attesa di connessioni...
Si è connesso il ristorante:sorbillo
Si è connesso il ristorante:antonio
Un cliente si e' connesso!
Si è connesso il cliente mario
Sono stati inviati i ristoranti al cliente!

aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
Benvenuto mario

Ti fornisco due scelte
1)Cerca ristorante
2)Esci
QUESTO E IL TUO ID: 925
Digita la tua scelta:mario
1
1)Il ristorante connesso e' il seguente:antonio
2)Il ristorante connesso e' il seguente:sorbillo

Puoi scegliere tra i ristoranti sopra citati utilizzando il loro ID
Fai la tua scelta
```

Invio dei dati al server, ricezione dei ristoranti da parte del cliente

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./s
In attesa di connessioni...
Si è connesso il ristorante:sorbillo
Si è connesso il ristorante:antonio
Un cliente si e' connesso!
Si è connesso il cliente mario
Sono stati inviati i ristoranti al cliente!
Aggiungo richiesta
Prodotti del ristorante mostrati al client.
Lettura ID Rider in corso...
Il client ha effettuato l'ordinazione ed è stata inviata al ristorante.
Prima di consegnare, prendo l'ID del rider e l'ID del cliente!
```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
Connessione al server in corso...
Sto inviando...
Il server ha bisogno dei miei prodotti!
Aspetto la scelta del client...
Il server ha bisogno delle mie bevande!
Aspetto scelta cliente...
Sto inviando...
Il cliente ha effettuato la scelta, vuole il prodotto con ID:1 come bever
prodotto con ID:3 per 1 persone!
Inoltro la richiesta al rider...
```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
*****
BIBITE DISPONIBILI
*****
Nome
IDProdotto:1 cola
IDProdotto:2 pepsi
IDProdotto:3 sprite
Digita l'ID corretto!
3
Il tuo ordine e' andato a buon fine!
Resta in linea finche non ricevi l'ID del Rider!
```

Operazione effettuate dal cliente,
in alto si evince come operano
nel mentre ristorante e server

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./ri
Digita il tuo nome: luigi
BENVENUTO:luigi
Connessione con il ristorante riuscita!
RIDER CON ID: 239
Vedo se ci sono ordini disponibili
Nessun ordine disponibile, premere[5] per aggiornare
```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./ri
Digita il tuo nome: luigi
BENVENUTO:luigi
Connessione con il ristorante riuscita!
RIDER CON ID: 239
Vedo se ci sono ordini disponibili
Nessun ordine disponibile, premere[5] per aggiornare
5
Vedo se ci sono ordini disponibili
-Ce un ordine in carico, vuoi provare a prenderlo?
[1]51
```

Il rider completerà l'ordine, se alla connessione
del rider c'è l'attesa di un ordine si verificherà
il caso a sinistra, mentre invece se non ci sono ordini disponibili
il rider aggiornerà fino a quando non ce ne sarà uno


```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ gcc -o r rider.c
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./r1
Digita il tuo nome: luigi
BENVENUTO:luigi

Connessione con il ristorante riuscita!

RIDER CON ID: 867
Vedo se ci sono ordini disponibili
-Ce un ordine in carico, vuoi provare a prenderlo?
[1]51

```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ gcc -o r12 rider2.c
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./r12
Digita il tuo nome: carlo
BENVENUTO:carlo

Connessione con il ristorante riuscita!

RIDER CON ID: 883
Vedo se ci sono ordini disponibili

```

Nel caso due rider, connessi alla stessa porta del ristorante, non potranno prendere lo stesso ordine (sinistra ha preso l'ordine, destra non ha fatto in tempo)

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
...
si è connesso il cliente salvatore
Sono stati inviati i ristoranti al cliente!
Aggiungo richiesta
Prodotti del ristorante mostrati al client.
Lettura ID Rider in corso...
Il client ha effettuato l'ordinazione ed è stata inviata al ristorante.
Prima di consegnare, prendo l'ID del rider e l'ID del cliente!
Il Rider:239 sta consegnando al cliente:714.....
Consegna effettuata!!!!
Lettura ID Rider in corso...
Un cliente si è disconnesso
Un ristorante si è disconnesso

```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
...
Il server ha bisogno delle mie bevande!
Aspetto scelta cliente....
Sto inviando....
Il cliente ha effettuato la scelta, vuole il prodotto con ID:2
prodotto con ID:3 per 1 persone!
Inoltre la richiesta al rider....
Il rider:239 si è connesso e vuole prendere l'incarico di consegna
Ho inviato l'ID del cliente al Rider!
Simulo consegna....
Consegna effettuata dal rider: 239 al cliente 714.
Ordine effettuato, vuoi continuare la giornata o chiudere baracca?
[0]Per chiudere, valore diverso da 0, continui la giornata
0
aulainfo@aulainfo-VirtualBox:~/Scrivania$

```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
aulainfo@aulainfo-VirtualBox:~$ cd Scrivania
aulainfo@aulainfo-VirtualBox:~/Scrivania$ ./r1
Digita il tuo nome: luigi
BENVENUTO:luigi

Connessione con il ristorante riuscita!

RIDER CON ID: 239
Vedo se ci sono ordini disponibili
nessun ordine disponibile, premere[5] per aggiornare
5
Vedo se ci sono ordini disponibili
-Ce un ordine in carico, vuoi provare a prenderlo?
[1]51
1
Consegna in corso.....
Ordine consegnato al cliente:714
aulainfo@aulainfo-VirtualBox:~/Scrivania$

```

```
aulainfo@aulainfo-VirtualBox: ~/Scrivania
File Modifica Visualizza Cerca Terminale Aiuto
...
BIBITE DISPONIBILI
*****
Nome
IDProdotto:1 acqua
IDProdotto:2 7up
IDProdotto:3 cola
Digita l'ID corretto!
3
Il tuo ordine e' andato a buon fine!
Resta in linea finche non ricevi l'ID del Rider!
Il rider con ID:239 presto inviera il prodotto richiesto
Adesso inviero il tuo id al server!
Arrivederci, buona giornata e torni a trovarci presto!
aulainfo@aulainfo-VirtualBox:~/Scrivania$

```

Risultato finale