

```
File Actions Edit View Help
(kali@kali)-[~]
$ nano gioco.c
(kali@kali)-[~]
$ ./gioco
start menu:
A>> Iniziare una nuova partita
B>> Uscire dal gioco
Inserisci la lettera corrispondente alla tua scelta
A
inserisci il tuo nome:
ALESSIO
benvenuto ALESSIO, iniziamo la partita
domanda numero 1:
quanti colpi conta il jeb diretto
A>>1
B>>2
C>>3
D>>4
inserire la risposta

```

```
GNU nano 6.4      gioco.c
#include <stdio.h>

void print_menu();
int gioca_partita();

int main () {
    char scelta= {'\0'};

    print_menu();
    scanf ("%c", &scelta);
    if (scelta== 'B')
    {
        printf ("grazie per aver giocato, alla prossima\n");
        return 0;
    }
    while (scelta == 'A')
    {
        gioca_partita();
        print_menu();
        scanf ("%c", &scelta);
    }
}

[ Bad lock file is ignored: ../gioco.c.swp ]
G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

```
GNU nano 6.4                                gioco.c
char risposta;
printf ("inserisci il tuo nome:\n");
scanf ( "%s", &nome);

printf ("benvenuto %s, iniziamo la partita\n", nome);
printf ("domanda numero 1:\n");
printf ("quanti colpi conta il jeb diretto\n");
printf ("A>>1\nB>>2\nC>>3\nD>>4\n");
printf ("inserire la risposta\n");
scanf ("%s", &risposta);

if (risposta == 'B')
{
printf ("risposta esatta\n");
punteggio++;
}
else {
printf ("risposta sbagliata\n");
}

domanda2

Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify

GNU nano 6.4                                gioco.c

printf ("start menu:\n");
printf ("A>> Iniziare una nuova partita\nB>> Uscire dal gioco\n");
printf ("Inserisci la lettera corrispondente alla tua scelta\n");

int gioca_partita()
{

int punteggio = 0;
char nome [20]= {'\0'};
char risposta;
printf ("inserisci il tuo nome:\n");
scanf ( "%s", &nome);

printf ("benvenuto %s, iniziamo la partita\n", nome);
printf ("domanda numero 1:\n");
printf ("quanti colpi conta il jeb diretto\n");
printf ("A>>1\nB>>2\nC>>3\nD>>4\n");

G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

Questo e il quiz creato da me.

IL modulo if/else lo uso per assegnare i punti, quindi if la risposta sarà corretta, do punto, else (invece) se la risposta sarà un'altra, darò come schermata (printf) risposta sbagliata, senza aggiungere un punto (++1) come nel caso la risposta sia giusta (if).

All'inizio ho fatto immettere il nome salvandolo con scanf nella variabile nome, come ho assegnato nella variabile punteggio l'incrementare dei punti in caso la risposta fosse giusta ed in risposta ho salvato le risposte di tutte e tre le domande, in modo tale da poterla usare nel modulo if/else.

Infine all'inizio ho usato il modulo if/while, in modo che prendesse blocchi di dati solo all'avverarsi di certe condizioni, quindi nel caso la risposta sarebbe stata B esci dalla partita, sarei tornato a main (return 0),

mentre se la risposta fosse stata A iniziare una partita, il processore del pc avrebbe continuato a leggere il programma arrivando a main giocare_partita iniziando il gioco.