

Programmazione C S2/L4

Manuel Di Gangi

1 febbraio 2024

TRACCIA

Lo scopo di oggi è realizzare un piccolo gioco di domanda/risposta in C, il numero e le domande sono a vostra scelta. Il gioco dovrà funzionare in modo tale da:

- Presentare una rapida introduzione all'utente con lo scopo del programma.
- Mostrare all'utente un menu di scelta iniziale tra: A) Iniziare una nuova partita; B) Uscire dal gioco.
- Ricevere in input la scelta dell'utente.
- Creare o meno una nuova partita in base all'input utente.
- Ricevere in input nome dell'utente in caso di nuova partita.
- Presentare un set di domande all'utente a risposta multipla (almeno 3 risposte a domanda).
- Valutare la risposta utente per ogni domanda ed aggiornare una variabile «punteggio in caso di risposta esatta».
- Scrivere a schermo a fine partita il punteggio totalizzato dal giocatore corrente.
- Presentare nuovamente il testo per la scelta tra: A) Iniziare una nuova partita; B) Uscire dal gioco.

Il codice

S2L4_Gioco_v1.c

La prima versione funzionante del programma:

```
1 #include <stdio.h>
2
3 //----- Prototipi funzione da implementare-----//
4 int partita(void);
5 int stampa_domanda(int, char[], char[], char[]);
6
7 void inserimento_classifica(int);
8 void mostra_classifica();
9
10 //----- Variabili globali-----//
11 int pnt[3]={0, 0, 0};
12
13
14 //----- Inizio programma -----//
15 int main()
16 {
17     char scelta;
18     char nome[10];
19
20     //Presentare una rapida introduzione all'utente con lo scopo del programma.
21     printf("-----\n");
22     printf("\tBenvenuto!\n");
23     printf("\tQuesto gioco ti chiederà di rispondere a delle \n\t domande riguardanti i comandi di Kali\n");
24     printf("\tOgni risposta esatta ti farà guadagnare 100 punti\n");
25     printf("\tIniziamo!!!\n\n");
26     printf("-----\n");
27
28     //Mostrare all'utente un menu di scelta iniziale tra: A) Iniziare una nuova partita; B) Uscire dal gioco.
29     //Di mia iniziativa ho inserito anche la voce visualizza classifica
30     printf("\nDigita -A- per iniziare una nuova partita\nDigita -B- per uscire dal gioco\nDigita -C- per visualizzare la
31 classifica\n");
32     //Ricevere in input la scelta dell'utente.
33     scanf("%c", &scelta);
```

```

33
34     while(scelta != 'b' && scelta != 'B')
35     {
36         switch(scelta){
37             case 'a':
38             case 'A':
39                 //Ricevere in input nome dell'utente in caso di nuova partita.
40                 printf("Inserisci il tuo Username: \n");
41                 scanf("%19s", nome);
42                 //Creare o meno una nuova partita in base all'input utente.
43                 //Richiamo la funzione partita
44                 int n = partita();
45                 inserimento_classifica(n);
46
47                 //Scrivere a schermo a fine partita il punteggio totalizzato dal giocatore corrente.
48                 printf("*****\n\n");
49                 printf("\t %s hai terminato la partita\n\tIl tuo punteggio è: %d punti\n\n", nome, n);
50                 printf("*****\n");
51                 break;
52             case 'c':
53             case 'C':
54                 mostra_classifica();
55                 break;
56             default:
57                 printf("Scelta non valida\n");
58                 break;
59         }
60
61         //Presentare nuovamente il testo per la scelta tra: A) Iniziare una nuova partita; B) Uscire dal gioco.
62         scanf("%c", &scelta);
63         printf("\nDigita -A- per iniziare una nuova partita\nDigita -B- per uscire dal gioco\nDigita -C- per
visualizzare la classifica\n");
64         scanf("%c", &scelta);
65

```

```

66     }
67
68     return 0;
69 }
70
71 //----- Definizione delle fnzioni -----//
72 int partita(void)
73 {
74     //Presentare un set di domande all'utente a risposta multipla (almeno 3 risposte a domanda).
75     int punti = 0;
76     printf("\nQual'è il comando per elencare il contenuto di una directory? \n");
77     punti += stampa_domanda(2, "dir", "ls", "mkdir");
78     printf("\nQuale comando permette di muoversi fra le directory? \n");
79     punti += stampa_domanda(1, "cd", "pwd", "rm");
80     printf("\nQuale comando possiamo utilizzare per spostare un file o una directory? \n");
81     punti += stampa_domanda(3, "nano", "rm", "mv");
82     printf("\nQuale comando permette di eseguire operazioni con i permessi di amministratore? \n");
83     punti += stampa_domanda(2, "admin", "sudo", "super");
84     return punti;
85 }
86
87 //Valutare la risposta utente per ogni domanda ed aggiornare una variabile «punteggio in caso di risposta esatta.
88 int stampa_domanda(int ris_cor, char risa[10] , char risb[10] , char risc[10] )
89 {
90     int ris;
91     printf("1- %s\n", risa);
92     printf("2- %s\n", risb);
93     printf("3- %s\n", risc);
94     scanf("%d", &ris);
95     if (ris == ris_cor)
96     {
97         return 100;
98     }else{
99         return 0;
100     }
101 }
102

```


Rendendo le operazioni iterabili

```
82 int partita(void)
83 {
84     //Presentare un set di domande all'utente a risposta multipla (almeno 3 risposte a domanda).
85     int punti = 0;
86     for(int i=0; i<4; i++) //itero n volte per mostrare tutte le domande
87     {
88         punti+=stampa_domanda(ris_cor[i], *ris_matrix[i], *dmnd[i]);
89     }
90     return punti;
91 }
92 }
93
94 //Valutare la risposta utente per ogni domanda ed aggiornare una variabile «punteggio in caso di risposta esatta.
95 int stampa_domanda(int ris_cor, char *ris, char dmnd[50])
96 {
97     int risposta_utente;
98     printf("\n%s\n ", dmnd); //stampo la domanda che ho ricevuto come parametro "dmnd"
99     for(int i=0; i<3; i++)
100     {
101         printf("%d-%s\n", i, ris[i]); //stampo le tre possibili risposte
102     }
103     scanf("%d", &risposta_utente);
104     if (risposta_utente == ris_cor) //confronto i risultati
105     {
106         return 100;
107     }else{
108         return 0;
109     }
110 }
111 }
```

L'esecuzione

- Avvio del programma

```
(kali㉿kali)-[~/Documents/C]
$ ./g
Benvenuto!
Questo gioco ti chiederà di rispondere a delle
domande riguardanti i comandi di Kali
Ogni risposta esatta ti farà guadagnare 100 punti
Iniziamo!!!

Digita -A- per iniziare una nuova partita
Digita -B- per uscire dal gioco
Digita -C- per visualizzare la classifica
a
Inserisci il tuo Username:
Manuel
```

- Inizio nuova partita

```

Inserisci il tuo Username:
Manuel
Qual'è il comando per elencare il contenuto di una directory?
1- dir
2- ls
3- mkdir
2
Qual'è il comando che permette di muoversi fra le directory?
1- cd
2- pwd
3- rm
1
Qual'è il comando che possiamo utilizzare per spostare un file o una directory?
1- nano
2- rm
3- mv
3
Qual'è il comando che permette di eseguire operazioni con i permessi di amministratore?
1- admin
2- sudo
3- super
2
|*****|
|
|      Manuel hai terminato la partita
|      Il tuo punteggio è: 400 punti
|
|*****|
|*****|
|*****|
Digita -A- per iniziare una nuova partita
Digita -B- per uscire dal gioco
Digita -C- per visualizzare la classifica

```

- Visualizzo la classifica delle varie partite

```

Digita -C- per visualizzare la classifica
C
|*****|
|      1* Player - Pnt. 400
|      2* Player - Pnt. 0
|      3* Player - Pnt. 0
|*****|

```

Esempio di classifica in un'altra partita

```

C
|*****|
|      1* Player - Pnt. 200
|      2* Player - Pnt. 100
|      3* Player - Pnt. 100
|*****|

```

- Chiusura del gioco

```

Digita -A- per iniziare una nuova partita
Digita -B- per uscire dal gioco
Digita -C- per visualizzare la classifica
B
(kali@kali)-[~/Documents/C]
$

```