


Esercizio S2_L4 Bug Hunting

Simone Mininni

Traccia

 Completato per oggi PDF **Esercizio**
Ottimizzazione del codice

Traccia:

Riprendete il codice del programma che avete scritto e pensiamo all'ottimizzazione del codice alla gestione delle situazioni non previste e facciamo le seguenti considerazioni:

- Cosa succede se l'utente inserisce una lettera diversa da A o B in fase di scelta iniziale? Il programma termina, ma non è una casistica che abbiamo gestito.
- Cosa succede se l'utente inserisce un nome che ha più caratteri della dimensione dell'array «nome» che abbiamo dichiarato inizialmente nella fase di avvio nuova partita? Riceveremo un errore (provate ad inserire una sequenza molto lunga di caratteri)
- Cosa succede se l'utente inserisce la lettera D per la risposta alle domande durante una partita? O un carattere numerico?

Tutte queste situazioni vanno considerate in fase di programmazione in quanto errori logici o errori di mancata gestione di situazioni non standard potrebbero portare a bug nel codice che potrebbero essere sfruttati da un attaccante per prendere controllo dell'esecuzione del programma ed eseguire codice malevolo.

Codice dopo bug hunting:

```
#include <stdio.h>
```

```
//Dichiarazione Funzioni
```

```
void print_menu();
```

```
void gioca_partita();
```

```
int controlloRisp(risposta);
```

```
void clear();
```

```
//Main
```

```
int main(void)
```

```
{
```

```
    char scelta = {'\0'};
```

```
    /**Controllo di input, do while per gestire eventuali input errati da parte dell' utente,  
    in quanto la condizione per uscire dal ciclo e terminare il programma è premere B,  
    A ti fa entrare nel gioco, quindi qualsiasi altro input non ti permette di uscire dal  
    ciclo.**/
```

```

do{
    print_menu();
    scanf(" %c", &scelta);

    if(scelta == 'A'){
        gioca_partita();
    }
}while(scelta!='B');

printf("Grazie per aver giocato, alla prossima\n");

return 0;
}

```

//Funzioni

void print_menu(){

```

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

```

void gioca_partita(){

```

    int punteggio = 0;
    char nome[20] = {"\0"};
    char risposta1, risposta2;
    int flag;

```

*/**possibile errore di stack overflow se la stringa inserita è più grande dello spazio allocato per l'array.**/*

//Soluzione

```

    printf ("Inserisci il tuo nome:\n");
    scanf(" %19s", &nome);
    clear(); //Ripulisce il buffer per scartare gli input in eccesso

```

```

    printf("Domanda n 1\n");
    printf("? \n");
    printf("A,B,C\n");

```

//Controllo input utente

```

do{
    scanf(" %c", &risposta1);
    flag = controlloRisp(risposta1); //Vedi funzione
}while(flag==0);

```

```
if(risposta1 == 'B'){  
    punteggio++;  
}
```

```
printf("Domanda n 2\n");  
printf("? \n");  
printf("A,B,C\n");
```

```
//Controllo input utente
```

```
do{  
    scanf(" %c", &risposta2);  
    flag = controlloRisp(risposta2);  
}while(flag==0);
```

```
if(risposta1 == 'A'){  
    punteggio++;  
}
```

```
printf("Punteggio concluso, punteggio totalizzato da %s:%d\n", nome, punteggio);  
  
}
```

```
int controlloRisp(risposta){
```

```
if(risposta=='A' || risposta == 'B' || risposta == 'C'){  
    return 1;  
}
```

```
printf("Premi A, B o C!\n");  
return 0;
```

```
}
```

```
void clear(){
```

```
while(getchar() != '\n');
```

```
}
```