

# M2-W6D1 – Esercizio 2

*DATA*

**Cybersecurity Analyst**

*Studente:*

*Andrea Scarmagnani*

*Docente:*

*Giuseppe Placanica*

## Traccia:

Traccia: Ieri abbiamo visto come scrivere un piccolo gioco di domande e risposte in C. Oggi pensiamo all'ottimizzazione del codice, ed alla gestione delle situazioni non previste. Riprendete il codice del programma che avete scritto ieri 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.

Esecuzione dell'esercizio sono stati corrette eventuali errori e aggiunto un nuovo sistema di input utilizzando `fgets(nome, sizeof(nome), stdin);`

```
/*
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
*/

#include <stdio.h>

void inizializzazione (){
    printf("Benvenuto nella KAME HOUSE, ora metteremo alla prova il tuo grado di conoscenza
di questa casa.\nPer ogni domanda corretta riceverà 10 punti per ogni domanda errata ti
verranno tolti 5 punti.\n");
    printf("\n\n A) Nuova Partita \n B) Uscire dal gioco\n");
    printf("Inserisci la tua Scelta: \n");
}

void Gioca(){
    char nome[30] = {'\0'};          //da controllare/studiare meglio... il proffe ha
    inserito ---- char nome[20]= {'\0'};
    char risposta;
```

```

int max_count = 30;
int punteggio = 0;

printf("Nuova partita \n");
printf("Inserisci il tuo nome: \n");
getchar();

//Altra sistema per leggere l'input da tastiera dando però dei limiti,
ATTENZIONE questo prevede un \n alla fine della
fgets(nome, sizeof(nome), stdin);

//scanf("%s", &nome);

printf("Bene %s, è ora di iniziare... il tuo punteggio iniziale è di %d a punti
totalizzane 30 per vincere.\n\n\n\n", nome, punteggio);

//Domanda 1
printf("\n\n\n");
printf("Come si chiama il proprietario della casa?..\n");
printf("A) Johnny Depp\n");
printf("B) Maestro delle tartarughe\n");
printf("C) Goku\n");
scanf("%s", &risposta);

switch (risposta){
    case 'a':
    case 'A':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;

    case 'b':
    case 'B':
        printf("Risposta Esatta\n");
        punteggio = punteggio + 10;
        break;

    case 'c':
    case 'C':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;

    default:
        break;
}

//Domanda 2
printf("\n\n\n");
printf("Dov'è situata la casa del maestro delle tartarughe?..\n");
printf("A) Nello spazio\n");
printf("B) In montagna\n");

```

```

printf("C) In un'isola\n");
scanf("%s", &risposta);

switch (risposta){
    case 'a':
    case 'A':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;

    case 'b':
    case 'B':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;

    case 'c':
    case 'C':
        printf("Risposta Esatta\n");
        punteggio = punteggio + 10;
        break;

    default:
        break;
}

//Domanda 3
printf("\n\n\n");
printf("Come si chiamano gli allievi del maestro?..\n");
printf("A) Goku e Crili\n");
printf("B) Capitan Marvel e Capitan America\n");
printf("C) Luffy e Zorro\n");
scanf("%s", &risposta);

switch (risposta){
    case 'a':
    case 'A':
        printf("Risposta Esatta\n");
        punteggio = punteggio + 10;
        break;

    case 'b':
    case 'B':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;

    case 'c':
    case 'C':
        printf("Risposta errata\n");
        punteggio = punteggio - 5;
        break;
}

```

```

        default:
            break;
    }
//Ho inserito il controllo if al posto (anche se sarebbe stato più semplice) di switch, a
scopo di studio. inserendo due if per un doppio controllo.
    if (punteggio == 30)
    {
        printf("\n\nComplimenti %s hai finito il gioco, con il massimo punteggio. Hai
totalizzato ben %d punti.\n\nTorna a trovarci.\n\n", nome, punteggio);
        return;
    }
    if (punteggio <= 0)
    {
        printf("\n\nIl tuo punteggio è di: %d Hai perso", punteggio);
        return;
    }
    else
    {
        printf("\n\nComplimenti %s hai finito il gioco, il tuo punteggio è di: %d",
nome, punteggio);
        return;
    }
}

int main()
{
    char scelta;
    char nome;

    inizializzazione();

    scanf(" %c", &scelta);

    printf("\n");

    switch(scelta){

        case 'a':
        case 'A':
            Gioca();
            break;

        case 'b':
        case 'B':
            printf("Grazie per aver giocato... Torna a trovarci presto");
            return 0;
            break;
    }
}

```

```
    default:  
        printf("Ricomincia");  
        inizializzazione();  
        main();  
    break;  
}  
  
}
```

# **Cybersecurity Analyst 2023**