

/ Scrivere un programma che richieda all'utente i dati relativi a 20 piloti di formula uno e li memorizzi in un vettore. Le informazioni per ogni pilota sono: nome e cognome del pilota, scuderia del pilota e punteggio in classifica da inizializzare a zero.*

Il programma deve inoltre richiedere all'utente l'esito di un gran premio, richiedendo il nome e il cognome dei classificati sul podio e deve aggiornare opportunamente la classifica assegnando 10 punti al vincitore, 6 punti al secondo classificato e 4 punti al terzo.

Modificare il programma in modo che l'aggiornamento della classifica venga ripetuto richiedendo l'esito di 10 gran premi consecutivi.

Il programma deve anche stampare il nome e il cognome del campione finale, riportandone il punteggio in classifica (si supponga per semplicità che non esistano piloti a parimerito).

**/*

```
#include<stdio.h>
#define NUMPILOTI 20
#define GRANPREMI 10

typedef struct{
    char nome[20];
    char cognome[20];
    char scuderia[15];
    int punteggio;
}pilota;

typedef enum{false,true} Boolean;

main ()
{
    pilota PILOTI[NUMPILOTI];
    char nome[20], cognome[20];
    int i,j,posizione,punti,maxpunteggio;
    Boolean trovato;

    for(i=0;i<NUMPILOTI;i++){
        printf("Dati del pilota n. %d\n", i+1);
        printf("Inserire il nome: ");
        gets(PILOTI[i].nome);
        printf("Inserire il cognome: ");
        gets(PILOTI[i].cognome);
        printf("Inserire la scuderia: ");
        gets(PILOTI[i].scuderia);
        PILOTI[i].punteggio=0;
    }

    for(j=0;j<GRANPREMI;j++){
        printf("Inserisci il podio del gran premio %d:\n", j+1);
        for(posizione=1;posizione<=3;posizione++){
            printf("Posizione numero %d:\n",posizione);
            printf("Nome: ");
            gets(nome);
            printf("Cognome: ");
            gets(cognome);
            switch(posizione){
                case(1): punti=10; break;
                case(2): punti=6; break;
                case(3): punti=4; break;
            }
            i=0;
        }
    }
}
```

```
        trovato=false;
        while(i<NUMPILOTI && !trovato){
            if((!strcmp(PILOTI[i].nome,nome)) && (!strcmp(PILOTI[i].cognome,cogn
ome))) {
                trovato=true;
                PILOTI[i].punteggio += punti;
            }
            i++;
        }
    }
}

strcpy(nome,PILOTI[0].nome);
strcpy(cognome,PILOTI[0].cognome);
maxpunteggio=PILOTI[0].punteggio;
for(i=1;i<NUMPILOTI;i++){
    if(PILOTI[i].punteggio > maxpunteggio){
        maxpunteggio=PILOTI[i].punteggio;
        strcpy(nome,PILOTI[i].nome);
        strcpy(cognome,PILOTI[i].cognome);
    }
}
printf("Il campione finale e': %s %s\n",nome,cognome);
printf("con punteggio: %d\n",maxpunteggio);
}
```