

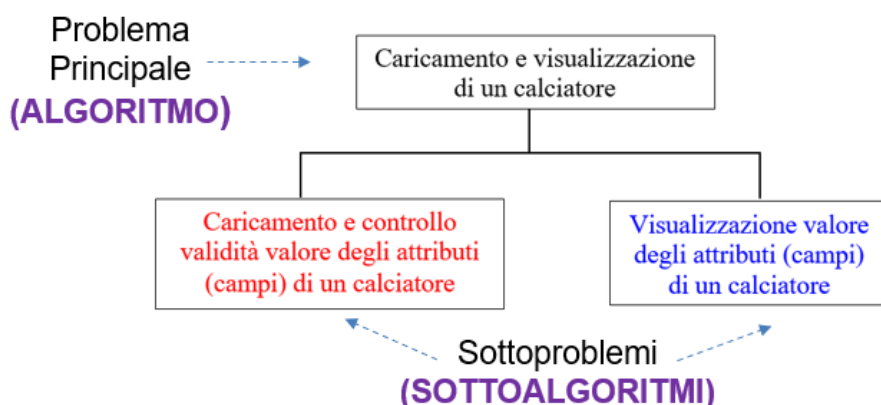
PROBLEMA: Caricamento e visualizzazione di un record di tipo "Calciatore" il cui tracciato è assegnato secondo la seguente rappresentazione tabellare (SOTTOPROGRAMMI):

Numero	Nome Campo	Tipo Campo	Lunghezza ¹	Descrizione	FlagSubRec ²
1	Cognome	ARRAY DI CHAR	30	Cognome del calciatore	
2	Nome	ARRAY DI CHAR	30	Nome del calciatore	
3	Maglia	INT	2	Num. maglia del calciatore	

1. In caso di valori numerici decimali scrivere **5,2** vuol dire **5** cifre intere in totale di cui **2** decimali (max 999.99)
 In caso di valori numerici interi scrivere **4** vuol dire prevedere un massimo di **4 cifre significative** (max 9999)

2. Se nel campo FlagSubRec viene posta una X vuol dire che quel campo deve essere considerato parte di un altro record (vedi SOTTORECORD)

N.B. Utilizzando la la METODOLOGIA di PROGETTAZIONE TOP-DOWN il problema assegnato potrebbe essere scomposto nei seguenti sottoproblemi:



Grazie alla metodologia di progettazione top-down, abbiamo dunque individuato i seguenti **SOTTOPROGRAMMI** (procedura o funzione) da implementare, che si devono occupare delle seguenti azioni:

- 1) il **caricamento** (con eventuale controllo del valore di tutti gli attributi (i campi))
- 2) la **visualizzazione** di tutti gli attributi (campi) di un RECORD con il seguente **tracciato (PSEUDOCODIFICA)**:

TIPO **Calciatore** = RECORD

Cognome : **ARRAY[30] DI CHAR**

Nome : **ARRAY[30] DI CHAR**

Maglia : **INT**

FINE RECORD

Prototipo o segnatura se **PROCEDURA**

1.a **PROCEDURA** Carica_Record_P (**REF** c : **Calciatore**)

Prototipo o segnatura se **FUNZIONE**

(N.B. IN ALTERNATIVA alla procedura Carica_Record_P())

1.b **FUNZIONE** Carica_Record_F () : **Calciatore**

Prototipo o segnatura se **PROCEDURA**

2 **PROCEDURA** Visualizza_Record_P (**VAL** c : **Calciatore**)

TABELLE DEI DATI

DATI DI INPUT DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

DATI DI OUTPUT DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
C	Calciatore	STATICA	Vedi TRACCIATO record	Istanza generica del tipo Calciatore
MAXNUMCHAR	INT	STATICA	30	Massimo numero di caratteri per Nome e Cognome
MINNUMMAGLIA	INT	STATICA	1	Minimo numero maglia
MAXNUMMAGLIA	INT	STATICA	99	Massimo numero maglia

ALGORITMO Record_E_Sottoprogrammi

MAXNUMCHAR 30

MINNUMMAGLIA 1

MAXNUMMAGLIA 99

TIPO Calciatore = RECORD

Cognome : **ARRAY**[MAXNUMCHAR] **DI CHAR**

Nome : **ARRAY**[MAXNUMCHAR] **DI CHAR**

Maglia : **INT**

FINE RECORD

PROCEDURA main()

c : Calciatore

INIZIO

//Utilizziamo la procedura 1.a

Carica_Record_P (c)

//Oppure potremmo utilizzare IN ALTERNATIVA la funzione 1.b

//**c** ← **Carica_Record_F**()

//Utilizziamo la procedura 2

Visualizza_Record_P (c)

RITORNA

FINE

TABELLE DEI DATI

DATI DI INPUT DEL SOTTOPROBLEMA: PROCEDURA Carica_Record_P()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
C	Calciatore	STATICA	Vedi TRACCIATO record	Parametro passato per RIFERIMENTO

DATI DI OUTPUT DEL SOTTOPROBLEMA: PROCEDURA Carica_Record_P()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
C	Calciatore	STATICA	Vedi TRACCIATO record	Parametro passato per RIFERIMENTO

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: PROCEDURA Carica_Record_P()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

PROCEDURA Carica_Record_P (**REF** c : **Calciatore**)

INIZIO

RIPETI

Scrivi("Cognome = ")

Leggi (c.Cognome)

FINCHE' (Lunghezza(c.Cognome) \neq 0) **AND**
(Lunghezza(c.Cognome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Nome = ")

Leggi (c.Nome)

FINCHE' (Lunghezza(c.Nome) \neq 0)
AND (Lunghezza(c.Nome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Maglia = ")

Leggi (c.Maglia)

FINCHE' (c.Maglia \geq MINNUMMAGLIA) **AND** (c.Maglia \leq MAXNUMMAGLIA)

RITORNA

FINE

ALTERNATIVA: TABELLE DEI DATI

DATI DI INPUT DEL SOTTOPROBLEMA: FUNZIONE Carica_Record_F()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
c	Calciatore	STATICA	Vedi TRACCIATO record	Calciatore generico

DATI DI OUTPUT DEL SOTTOPROBLEMA: FUNZIONE Carica_Record_F()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
c	Calciatore	STATICA	Vedi TRACCIATO record	Calciatore generico restituito nel nome della funzione

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: FUNZIONE Carica_Record_F()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

FUNZIONE Carica_Record_F () : **Calciatore**

c : Calciatore

INIZIO

RIPETI

Scrivi("Cognome = ")

Leggi (c.Cognome)

FINCHE' (Lunghezza(c.Cognome) \neq 0) **AND**
(Lunghezza(c.Cognome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Nome= ")

Leggi (c.Nome)

FINCHE' (Lunghezza(c.Nome) \neq 0)
AND (Lunghezza(c.Nome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Maglia = ")

Leggi (c.Maglia)

FINCHE' (c.Maglia \geq MINNUMMAGLIA) **AND** (c.Maglia \leq MAXNUMMAGLIA)

RITORNA (c)

FINE

TABELLE DEI DATI

DATI DI INPUT DEL SOTTOPROBLEMA: PROCEDURA Visualizza_Record_P()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
C	Calciatore	STATICA	Vedi TRACCIATO record	Parametro passato per VALORE

DATI DI OUTPUT DEL SOTTOPROBLEMA: PROCEDURA Visualizza_Record_P()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
C	Calciatore	STATICA	Vedi TRACCIATO record	Parametro passato per VALORE

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: PROCEDURA Visualizza_Record_P()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

PROCEDURA Visualizza_Record_P (VAL c : Calciatore)

INIZIO

Scrivi("Cognome immesso: ")

Scrivi (c.Cognome)

Scrivi("Nome immesso: ")

Scrivi (c.Nome)

Scrivi("Maglia immessa: ")

Scrivi (c.Maglia)

RITORNA

FINE

ALGORITMO Record_E_Sottoprogrammi

MAXNUMCHAR 30

MINNUMMAGLIA 1

MAXNUMMAGLIA 99

TIPO Calciatore = **RECORD**Cognome : **ARRAY**[MAXNUMCHAR] **DI CHAR**Nome : **ARRAY**[MAXNUMCHAR] **DI CHAR**Maglia : **INT****FINE RECORD****PROCEDURA** main()

c : Calciatore

INIZIO

//Utilizziamo la procedura 1.a

Carica_Record_P (c)//Oppure potremmo utilizzare **IN ALTERNATIVA** la funzione 1.b//c ← **Carica_Record_F**()

//Utilizziamo la procedura 2

Visualizza_Record_P (c)**RITORNA****FINE****PROCEDURA** Carica_Record_P (**REF** c : **Calciatore**)**INIZIO****RIPETI**

Scrivi("Cognome = ")

Leggi (c.Cognome)

FINCHE' (Lunghezza(c.Cognome) ≠ 0) AND
(Lunghezza(c.Cognome) ≤ MAXNUMCHAR)**RIPETI**

Scrivi("Nome = ")

Leggi (c.Nome)

FINCHE' (Lunghezza(c.Nome) ≠ 0)
AND (Lunghezza(c.Nome) ≤ MAXNUMCHAR)**RIPETI**

Scrivi("Maglia = ")

Leggi (c.Maglia)

FINCHE' (c.Maglia ≥ MINNUMMAGLIA) AND (c.Maglia ≤ MAXNUMMAGLIA)**RITORNA****FINE**

FUNZIONE Carica_Record_F () : **Calciatore**

POSSIBILE ALTERNATIVA

c : Calciatore

INIZIO

RIPETI

Scrivi("Cognome = ")

Leggi (c.Cognome)

FINCHE' (Lunghezza(c.Cognome) \neq 0) **AND**
(Lunghezza(c.Cognome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Nome= ")

Leggi (c.Nome)

FINCHE' (Lunghezza(c.Nome) \neq 0)
AND (Lunghezza(c.Nome) \leq MAXNUMCHAR)

RIPETI

Scrivi("Maglia = ")

Leggi (c.Maglia)

FINCHE' (c.Maglia \geq MINNUMMAGLIA) **AND** (c.Maglia \leq MAXNUMMAGLIA)

RITORNA (c)

FINE

PROCEDURA Visualizza_Record_P (**VAL** c : **Calciatore**)

INIZIO

Scrivi("Cognome immesso: ")

Scrivi (c.Cognome)

Scrivi("Nome immesso: ")

Scrivi (c.Nome)

Scrivi("Maglia immessa: ")

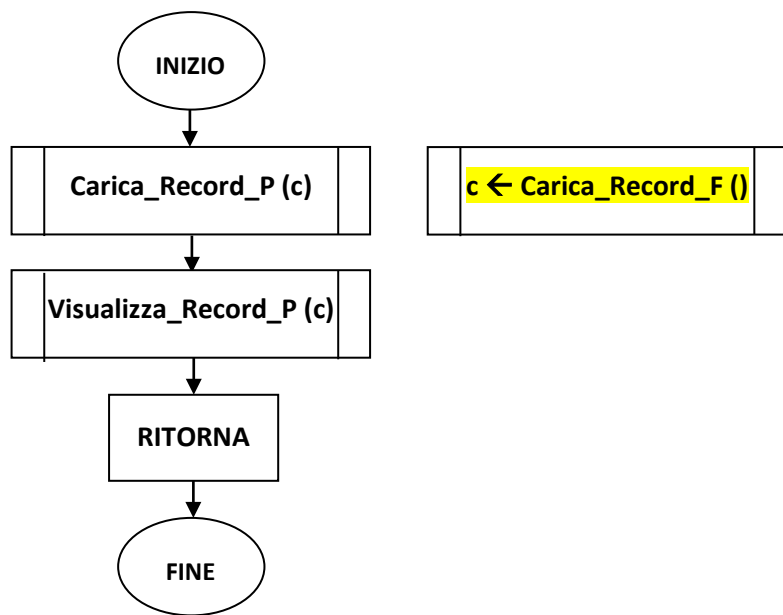
Scrivi (c.Maglia)

RITORNA

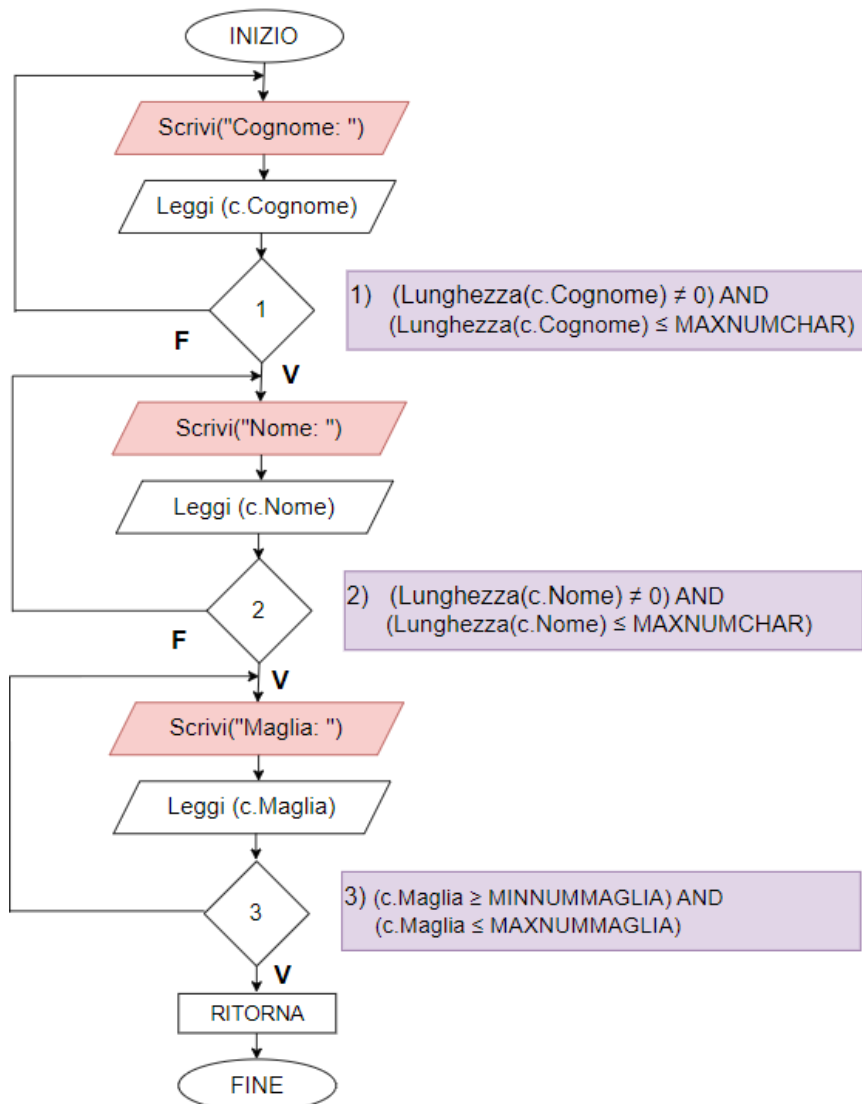
FINE

FLOWCHART

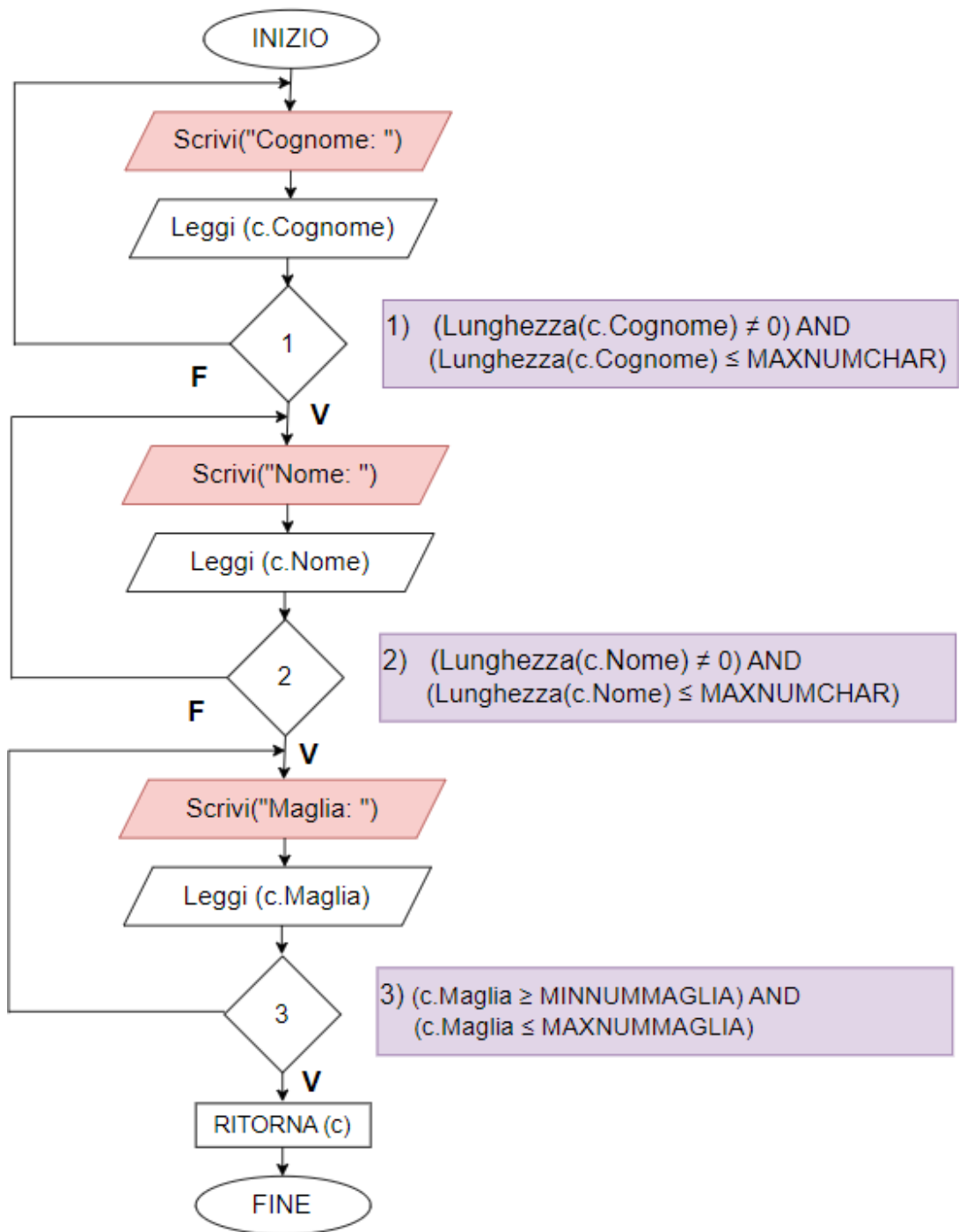
PROCEDURA **main** ()



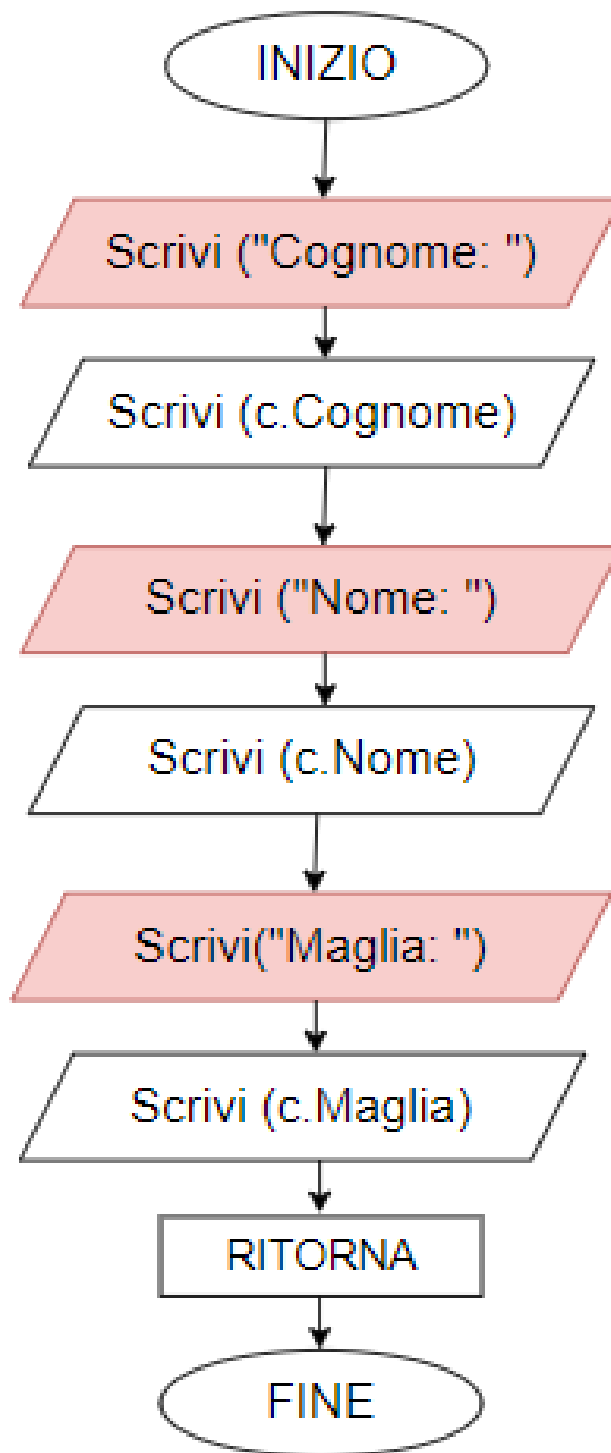
PROCEDURA **Carica_Record_P** ()



ALTERNATIVA: FUNZIONE `Carica_Record_F ()`



PROCEDURA **Visualizza_Record_P ()**



CODIFICA C

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAXNUMCHAR 30
#define MINNUMMAGLIA 1
#define MAXNUMMAGLIA 99

typedef struct
{
    char Cognome [MAXNUMCHAR + 1];
    char Nome [MAXNUMCHAR + 1];
    int Maglia;
} Calciatore;

// prototipi funzioni C necessarie
void Carica_Record_P (Calciatore *c);
Calciatore Carica_Record_F ();
void Visualizza_Record_P (Calciatore c);

// PROCEDURA main()
int main(int argc, char*argv[])
{
    /* dati di input/output */
    Calciatore c;

    /* CALL alla PROCEDURA Carica_Record_P() */
    Carica_Record_P (&c);
    /* CALL alla FUNZIONE Carica_Record_F() */
    //c = Carica_Record_F();
    /* CALL alla PROCEDURA Visualizza_Record_P() */
    Visualizza_Record_P (c);
    return 0;
}

// PROCEDURA Carica_Record_P()
void Carica_Record_P (Calciatore *c)
{
    // caricamento e controllo campi del record
    do
    {
        printf ("Cognome: ");
        gets ((*c).Cognome);
    }
    while ( (strlen((*c).Cognome) == 0) || (strlen((*c).Cognome) > MAXNUMCHAR) );
    do
    {
        printf ("Nome: ");
        gets ((*c).Nome);
    }
    while ( (strlen((*c).Nome) == 0) || (strlen((*c).Nome) > MAXNUMCHAR) );
    do
    {
        printf ("Maglia: ");
        scanf("%d", &(c->Maglia));    //alternativa scanf("%d", &((*c).Maglia));
    }
    while ( ((*c).Maglia < MINNUMMAGLIA) || ((*c).Maglia > MAXNUMMAGLIA) );
    return;
}
```

```

// FUNZIONE Carica_Record_F()
Calciatore Carica_Record_F ()
{
Calciatore c;

// caricamento e controllo campi del record
do
{
printf ("Cognome: ");
gets (c.Cognome);
}
while ( (strlen(c.Cognome) == 0) || (strlen(c.Cognome) > MAXNUMCHAR) );
do
{
printf ("Nome: ");
gets (c.Nome);
}
while ( (strlen(c.Nome) == 0) || (strlen(c.Nome) > MAXNUMCHAR) );
do
{
printf ("Maglia: ");
scanf ("%d", &(c.Maglia));
}
while ( (c.Maglia < MINNUMMAGLIA) || (c.Maglia > MAXNUMMAGLIA) );
return (c);
}

// PROCEDURA Visualizza_Record_P()
void Visualizza_Record_P (Calciatore c)
{
// Visualizzazione campi del record
// Cognome
printf ("\nCognome immesso: ");
puts (c.Cognome);
// Nome
printf ("\nNome immesso: ");
puts (c.Nome);
// Maglia
printf ("\nMaglia immessa: ");
printf ("%d",c.Maglia);
return;
}

```