



# Corso di Informatica Medica

## Esercitazione VI

29 maggio 2014

Alessandro A. Nacci  
[nacci@elet.polimi.it](mailto:nacci@elet.polimi.it) - [alessandronacci.com](http://alessandronacci.com)





# Automobili

- Rappresentare in  $C$  una automobile. Nel nostro caso, una automobile è descritta da un nome, un costo, un colore, da un insieme di componenti e da un libretto di circolazione.
- Un componente ha un nome, un costo ed una categoria. Le categorie possibili sono TRAZIONE, MULTIMEDIA, SICUREZZA
- Il libretto di circolazione riporta invece l'anno e la provincia di immatricolazione e in che classe Euro rientra.
- Il programma deve poter permettere la creazione di auto e la stampa a schermo di tutti i dati relativi ad un'auto
- Deve poter permettere inoltre di modificare il nome dell'auto
- Deve poter calcolare il costo totale per la produzione dell'auto



# Automobili: Le strutture dati - Codice C

```
typedef enum {TRAZIONE, MULTIMEDIA, SICUREZZA} tipi_categoria;  
  
typedef struct {  
    int anno_immatricolazione;  
    char provincia[STR_LEN];  
    int classe_euro;  
} libretto_circolazione;  
  
typedef struct {  
    char nome[STR_LEN];  
    double costo;  
    tipi_categoria categoria;  
} componente;  
  
typedef struct {  
    char nome[STR_LEN];  
    double costo;  
    char colore[STR_LEN];  
    int numero_componenti;  
    componente* componenti;  
    libretto_circolazione libretto;  
} automobile;
```



- Scriviamo una funzione per la creazione di un generico libretto di circolazione



- Scriviamo una funzione per la creazione di un generico libretto di circolazione

```
libretto_circolazione crea_libretto_circolazione(int anno_immatricolazione,  
                                                char provincia[STR_LEN], int classe_euro)  
{  
  
    libretto_circolazione libretto;  
    libretto.anno_immatricolazione = anno_immatricolazione;  
    strcpy(libretto.provincia,provincia);  
    libretto.classe_euro = classe_euro;  
  
    return libretto;  
}
```



- Scriviamo una funzione per la creazione di un generico componente di un'auto



- Scriviamo una funzione per la creazione di un generico componente di un'auto

```
componente crea_componente(char nome[STR_LEN],
                          double costo, int categoria)
{
    componente c;

    strcpy(c.nome, nome);
    c.costo = costo;
    c.categoria = categoria;

    return c;
}
```



- Scriviamo una funzione per la creazione di una generica automobile



- Scriviamo una funzione per la creazione di una generica automobile

```
automobile crea_auto(char nome[STR_LEN], double costo, char colore[STR_LEN],
                    int numero_componenti, componente* componenti,
                    libretto_circolazione libretto)
{
    printf("Creo una nuova autovettura di nome: %s\n", nome);

    automobile autovettura;

    strcpy(autovettura.nome, nome);
    autovettura.costo = costo;
    strcpy(autovettura.colore, colore);
    autovettura.numero_componenti = numero_componenti;
    autovettura.componenti = componenti;
    autovettura.libretto = libretto;

    return autovettura;
}
```



# Stampa a schermo dei dati di un'auto

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

**COMPONENTI**

**NOME**

**COSTO**

**CATEGORIA**

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

**COMPONENTI**

*stringa*

**NOME**

**COSTO**

**CATEGORIA**

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

**COMPONENTI**

**NOME**

**COSTO**

**CATEGORIA**

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO

*stringa*

*float*





# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO

*stringa*

*float*

*enum*



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO

stringa

float

enum

```
char* stringa_categoria(tipi_categoria categoria)
{
    if (categoria == TRAZIONE) return "TRAZIONE";
    if (categoria == MULTIMEDIA) return "MULTIMEDIA";
    if (categoria == SICUREZZA) return "SICUREZZA";

    return "SCONOSCIUTO";
}
```



# Stampa di un componente

```

void stampa_componenti(componente* componenti, int numero_componenti)
{
  int i;

  for (i = 0; i < numero_componenti; i++)
  {
    printf("Nome componente: %s |\t Costo: eur. %f |\t Categoria: %s \n",
           componenti[i].nome, componenti[i].costo,
           stringa_categoria(componenti[i].categoria));
  }
}

```

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO

stringa

float

enum

```

char* stringa_categoria(tipi_categoria categoria)
{
  if (categoria == TRAZIONE) return "TRAZIONE";
  if (categoria == MULTIMEDIA) return "MULTIMEDIA";
  if (categoria == SICUREZZA) return "SICUREZZA";

  return "SCONOSCIUTO";
}

```



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

**COMPONENTI**

**NOME**

**COSTO**

**CATEGORIA**

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO

```
float calcola_costo_componenti(automobile autovettura)
{
    int i;
    float tot = 0;

    for (i = 0; i < autovettura.numero_componenti; i++)
    {
        tot += autovettura.componenti[i].costo;
    }

    return tot;
}
```



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

**LIBRETTO**

**ANNO IMM.**

**PROVINCIA**

**CLASSE EURO**



# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

**LIBRETTO**

**ANNO IMM.**

**PROVINCIA**

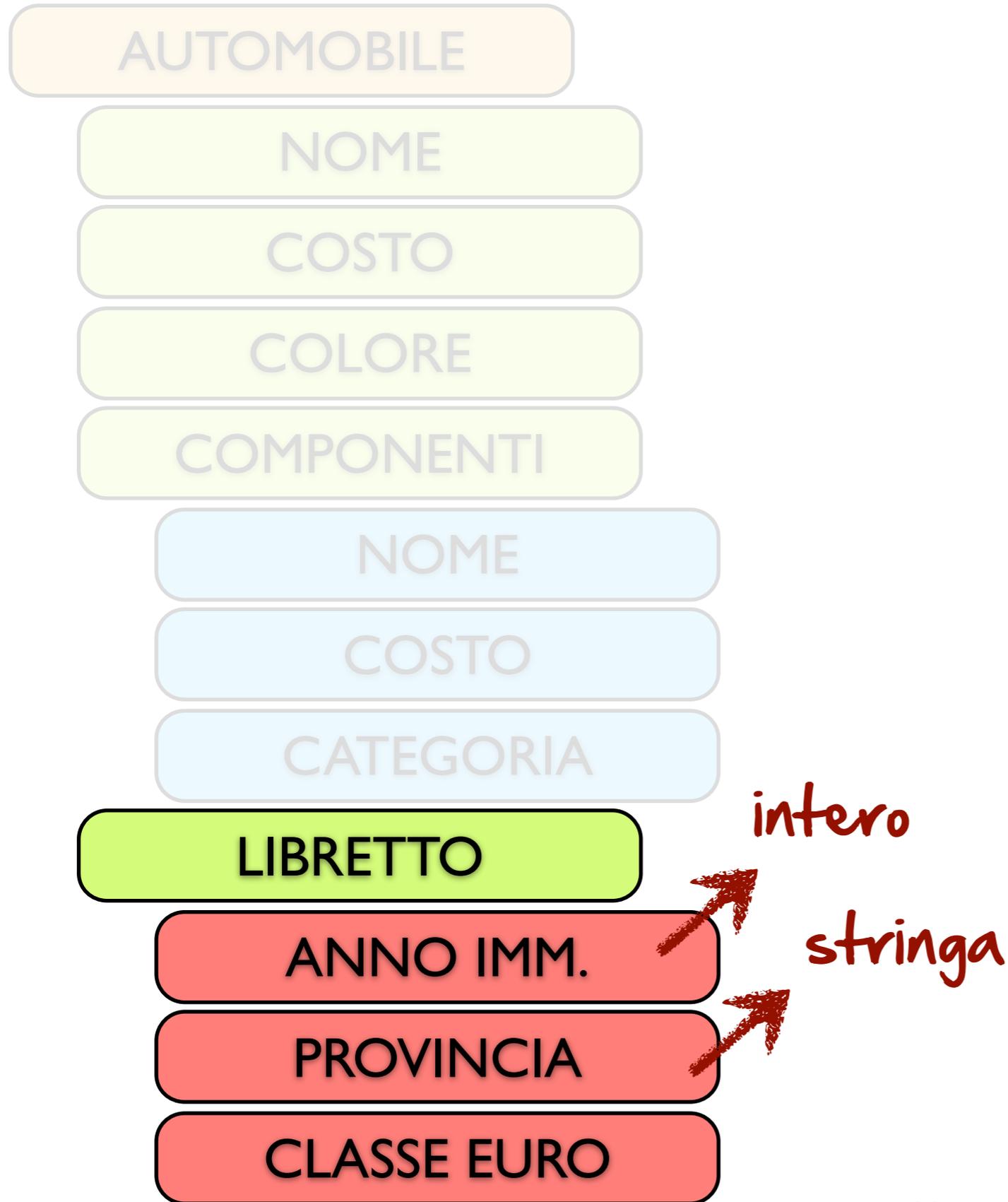
**CLASSE EURO**

*intero*





# Stampa di un componente





# Stampa di un componente

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

**LIBRETTO**

**ANNO IMM.**

**PROVINCIA**

**CLASSE EURO**

*intero*

*stringa*

*intero*





# Stampa di un componente

AUTOMOBILE

NOME

```
void stampa_libretto_circolazione(libretto_circolazione libretto)
{
    printf("Anno: %d |\t Prov.:%s |\t Euro:%d\n", libretto.anno_immatricolazione,
        libretto.provincia, libretto.classe_euro);
}
```

NOME

COSTO

CATEGORIA

**LIBRETTO**

ANNO IMM.

PROVINCIA

CLASSE EURO

*intero*

*stringa*

*intero*



# Stampa a schermo dei dati di un'auto

AUTOMOBILE

NOME

COSTO

COLORE

COMPONENTI

NOME

COSTO

CATEGORIA

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO



# Stampa a schermo dei dati di un'auto

AUTOMOBILE

NOME

COSTO

```
void stampa_auto(automobile autovettura)
{
    printf("Nome: %s\n", autovettura.nome);
    printf("Colore: %s\n", autovettura.colore);
    printf("Costo: eur. %f\n", autovettura.costo);
    printf("Costo componenti: eur. %f \n", calcola_costo_componenti(autovettura) );
    printf("\nCOMPONENTI:\n");
    printf("-----\n");
    stampa_componenti(autovettura.componenti, autovettura.numero_componenti);
    printf("\nLIBRETTO CIRCOLAZIONE:\n");
    printf("-----\n");
    stampa_libretto_circolazione(autovettura.libretto);
}
```

LIBRETTO

ANNO IMM.

PROVINCIA

CLASSE EURO



# Una prima parte di main() ...

```
int main () {

    automobile autovettura;
    componente componenti[MAX_COMP];
    libretto_circolazione libretto;

    automobile* ptr0;
    automobile* ptr1;
    automobile* ptr2;
    automobile* ptr3;

    // Creiamo il componente "FRENO"
    componenti[0] = crea_componente("FRENO", 420.20, TRAZIONE);
    componenti[1] = crea_componente("RUOTA", 656.40, TRAZIONE);

    // Creiamo le informazioni del libretto
    libretto = crea_libretto_circolazione(2010, "COMO", 5);

    // Creiamo una autovettura
    autovettura = crea_auto("FIAT BRAVO", 2000.00, "BLU", 2, componenti, libretto);
    ptr0 = &autovettura;

    // Stampiamo quello che abbiamo creato
    printf("\nBenvenuto!\n\n\n");
    stampa_auto(autovettura);

    return 0;
}
```



# Occupiamoci della modifica

```
automobile modifica_nome_auto(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```

```
void modifica_nome_auto2(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```

```
void modifica_nome_auto3(automobile* autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```



# Occupiamoci della modifica

```
automobile modifica_nome_auto(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura.nome, nuovo_nome);  
    return autovettura;  
}
```

```
void modifica_nome_auto2(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```

```
void modifica_nome_auto3(automobile* autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```



# Occupiamoci della modifica

```
automobile modifica_nome_auto(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura.nome, nuovo_nome);  
    return autovettura;  
}
```

```
void modifica_nome_auto2(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura.nome, nuovo_nome);  
}
```

```
void modifica_nome_auto3(automobile* autovettura, char nuovo_nome[STR_LEN])  
{  
  
}
```



# Occupiamoci della modifica

```
automobile modifica_nome_auto(automobile autovettura, char nuovo_nome[STR_LEN])
{
    strcpy(autovettura.nome, nuovo_nome);
    return autovettura;
}
```

```
void modifica_nome_auto2(automobile autovettura, char nuovo_nome[STR_LEN])
{
    strcpy(autovettura.nome, nuovo_nome);
}
```

```
void modifica_nome_auto3(automobile* autovettura, char nuovo_nome[STR_LEN])
{
    strcpy(autovettura->nome, nuovo_nome);
}
```



# Occupiamoci della modifica

```
automobile modifica_nome_auto(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura.nome, nuovo_nome);  
    return autovettura;  
}
```

```
void modifica_nome_auto2(automobile autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura.nome, nuovo_nome);  
}
```

```
void modifica_nome_auto3(automobile* autovettura, char nuovo_nome[STR_LEN])  
{  
    strcpy(autovettura->nome, nuovo_nome);  
}
```



# Finiamo il main() ...

```
int main () {

    automobile autovettura;
    componente componenti[MAX_COMP];
    libretto_circolazione libretto;

    automobile* ptr0;
    automobile* ptr1;
    automobile* ptr2;
    automobile* ptr3;

    // Creiamo il componente "FRENO"
    componenti[0] = crea_componente("FRENO", 420.20, TRAZIONE);
    componenti[1] = crea_componente("RUOTA", 656.40, TRAZIONE);

    // Creiamo le informazioni del libretto
    libretto = crea_libretto_circolazione(2010, "COMO", 5);

    // Creiamo una autovettura
    autovettura = crea_auto("FIAT BRAVO", 2000.00, "BLU", 2, componenti, libretto);
    ptr0 = &autovettura;

    // Stampiamo quello che abbiamo creato
    printf("\nBenvenuto!\n\n\n");
    stampa_auto(autovettura);

    printf("\nModifico nome auto...\n\n\n");
    autovettura = modifica_nome_auto(autovettura, "FIAT PUNTO");
    stampa_auto(autovettura);

    printf("\nModifico nome auto...\n\n\n");
    modifica_nome_auto2(autovettura, "FIAT ULISSE");
    stampa_auto(autovettura);

    printf("\nModifico nome auto...\n\n\n");
    modifica_nome_auto3(&autovettura, "FIAT PANDA");
    stampa_auto(autovettura);

    return 0;
}
```

**Potete lasciare il vostro giudizio qui:**

**<http://tinyurl.com/IEIMExe2014>**

**Tutte il materiale sarà disponibile sul mio  
sito internet:**

**[alessandronacci.com](http://alessandronacci.com)**

**See You Next Time!**

