

Tipi di dati derivati: strutture

Salvatore Filippone
salvatore.filippone@uniroma2.it

Strutture

Una struttura è un tipo di dato derivato costituito da una collezione di dati potenzialmente disomogenei tra loro; i singoli elementi vengono acceduti per *nome*

```
1  struct book {
2      char title[10];
3      char author[10];
4      int publication_year;
5      float price;
6  };
7
8  struct book  mybook;
9  printf("Autore:  %s\n Titolo: %s\n",mybook.author,mybook.title);
```

```
#include <stdio.h>
#include <stdlib.h>
#include <assert.h>
#include <string.h>
```

```
struct person {
    char *name;
    int age;
    int height;
    int weight;
};
```

```
struct person *person_create(struct person *who,
                             char *name, int age, int height, int weight)
{
    assert(who != NULL);
    who->name = strdup(name);
    who->age = age;
    who->height = height;
    who->weight = weight;
    return who;
}
```

```
void person_print(struct person *who)
{
    printf("Name: %s\n", who->name);
    printf("\tAge: %d\n", who->age);
    printf("\tHeight: %d\n", who->height);
    printf("\tWeight: %d\n", who->weight);
}

int main(int argc, char *argv[])
{
    struct person joe;
    (void)person_create(&joe, "Joe Alex", 32, 64, 140);
    printf("Joe is at memory location %p:\n", &joe);
    person_print(&joe);
    joe.age += 20;
    person_print(&joe);
    return 0;
}
```

Una struttura può contenere un puntatore (o anche diversi puntatori) ad una variabile del suo stesso tipo

```
#include "list.h"
typedef struct list {
    int data;
    struct list *next;
} List;
```

```
void foo()
{
    List *head;

    head = (List *) malloc(sizeof(List));
}
```

Le strutture sono fondamentali:

- Nella realizzazione di qualunque programma complesso
- Come livello di astrazione necessario a ragionare sui dati
- Per incapsulare funzionalità, e rendere possibili miglioramenti futuri