



Typedef

Tipo enumeration
Libreria ctype.h

Ing. Nadia Ranaldo
Dipartimento di Ingegneria
Università degli Studi del Sannio

Typedef



- Il C permette al programmatore di definire nuovi tipi a partire da quelli definiti dal linguaggio
- La parola chiave da utilizzare è **typedef**

Sintassi

```
typedef <tipo esistente> <nuovotipo>
```

- Esempio

```
typedef int gradi;  
/* gradi è il nuovo tipo, definito come int*/  
gradi a,b,c,d;  
/* a,b,c e d sono variabili di tipo gradi, cioè int */
```

2

Typedef con array



- Definire un nuovo tipo sugli array

```
typedef int voti[100];  
/* voti è un tipo array di 100 elementi di  
tipo intero in cui l'indice può variare da 0  
a 99 */  
  
voti v1, v2;  
/* v1 e v2 sono due array di 100 interi */
```

3

Tipo enumeration (1)



- Esempio di tipo definito dall'utente in base al tipo di applicazione
- Un **tipo enumeration** è un tipo di dato caratterizzato da una lista di possibili valori
- Ad esempio
`{LUN, MAR, MERC, GIOV, VEN, SAB, DOM}`
- In C il costrutto **enum** permette di creare una variabile di tipo enumeration
- Come vengono memorizzate le variabili di tipo enumeration?
- Il C associa ad ogni valore un numero intero ed utilizza un ordinamento nella lista dei valori
- Internamente quindi utilizza un tipo intero per rappresentare variabili di questo tipo

4

Tipo enumeration (2)



- Esempio:

```
enum {LUN, MAR, MERC, GIOV, VEN, SAB, DOM } giorno;
```

giorno è una variabile che vale uno dei valori listati

- Spesso conviene utilizzare un tipo per poter dichiarare variabili

```
typedef  
enum {LUN, MAR, MERC, GIOV, VEN, SAB, DOM }  
giornoSett;
```

```
giornoSett domani, oggi;
```

- `giornoSett` è il tipo delle variabili che possono assumere i valori listati, e `domani` e `oggi` sono due variabili di tale tipo

5

Tipo enumeration (3)



semplicemente delle nuove costanti, che valgono 0, 1, 2, 3, 4, 5, 6 (equivalente ad usare tante `#define LUN 0`, etc.)

```
typedef  
enum {LUN, MAR, MERC, GIOV, VEN, SAB, DOM }  
giornoSett;  
giornoSett domani, oggi;  
int ieri; /* minore informazione!*/
```

- Sono implementati come interi, a tutti gli effetti, quindi è possibile effettuare delle operazioni aritmetiche e di confronto

```
if ( (oggi > MERC) && (oggi <= VEN) ) ...
```

```
if ( oggi == DOM ) domani = LUN; else domani =  
oggi+1;
```

- Come si fanno trascorrere 4 giorni?
- In C non c'è il controllo sui valori assegnati!

```
oggi = (oggi + 4) % 7; /* somma modulo 7!!! */
```

6

Esempio (1)



```
typedef enum {OAK, MAPLE, ELM} treeT;  
/* quercia, acero, olmo */  
main() {  
    treeT myTree;  
    myTree = MAPLE;  
    ShowTreePicture(myTree);  
}  
/* funzione che visualizza un albero sulla  
base del tipo */  
void showTreePicture(treeT tree) {  
    ...  
}
```

7

Esempio (2)



- E' possibile specificare il valore di alcuni elementi del tipo enumeration

```
typedef enum {OAK=1, MAPLE, ELM} treeT;
```

- In questo caso che valore è assegnato a MAPLE?
2

```
typedef enum {OAK=1, MAPLE, ELM=1} treeT;
```

- Il valore assegnato a OAK e ELM è 1

8

Esempio (3)



```
typedef enum {
    Sunday, Monday, Tuesday, Wednesday,
    Thursday, Friday, Saturday
} weekdayT;
main(){
    weekdayT day;
    /* legge day */
    switch (day) {
        case Sunday: printf("Sunday"); break;
        case Monday: printf("Monday"); break;
        case Tuesday: printf("Tuesday"); break;
        case Wednesday: printf("Wednesday"); break;
        case Thursday: printf("Thursday"); break;
        case Friday: printf("Friday"); break;
        case Saturday: printf("Saturday"); break;
    }
    printf("\n");
}
```

9

Caratteri: Libreria ctype.h (1)



- La libreria standard del C ctype.h fornisce un insieme di funzioni per la manipolazione di caratteri
- Le seguenti funzioni ritornano un valore diverso da zero (TRUE) se l'argomento ch soddisfa la condizione richiesta, zero (FALSE) altrimenti

```
#include <ctype.h>
```

```
islower(ch) /* è minuscolo? */
```

```
isupper(ch) /* è maiuscolo? */
```

```
isalpha(ch) /* è una lettera alfabetica? */
```

```
isdigit(ch) /* è una cifra numerica? */
```

```
isalnum(ch) /* è una lettera oppure una cifra numerica */
```

```
ispunct(ch) /* è un caratteri di punteggiatura */
```

```
isspace(ch) /* <space> \n o \t */
```

10

Caratteri: Libreria ctype.h (2)



- Funzioni che convertono le lettere:

```
tolower(ch)
```

```
/* se ch è una lettera maiuscola ritorna la corrispondente
   lettera minuscolo, altrimenti restituisce ch */
```

```
toupper(ch)
```

```
/* se ch è una lettera minuscola ritorna la corrispondente
   lettera maiuscolo, altrimenti restituisce ch */
```

11

Esempio



- Per verificare quante lettere inserite sono vocali
- Prima convertiamo il carattere in minuscolo e poi effettuiamo il confronto con (a e i o u)

```
#include <stdio.h>
```

```
#include <ctype.h>
```

```
main() {
```

```
    char c, pausa;
```

```
    int count;
```

```
    count = 0;
```

```
    do {
```

```
        printf("Inserire una lettera: ");
```

```
        scanf("%c",&c); scanf("%c",&pausa);
```

```
        c = tolower(c);
```

```
        switch (c){
```

```
            case 'a': case 'e': case 'i':
```

```
            case 'o': case 'u': {count++; break;}
```

```
        }
```

```
    } while (c!='\0');
```

```
    printf("Il numero di vocali inserite è %d\n", count);
```

```
}
```

12