

Tipi semplici 1

Contenuti rieditati delle slide della
Prof. L. Caponetti

Dati

- I dati rappresentano oggetti e proprietà della realtà rilevanti per il problema in esame
- I dati sono una **astrazione** di oggetti del mondo reale

Astrazione

- Che cosa si intende per astrazione?
 - *Processo secondo cui si ignorano gli aspetti di una realtà **non rilevanti** per lo scopo del momento in modo da concentrarsi completamente sugli aspetti **più rilevanti***
- Una astrazione è una semplificazione

Linguaggi di programmazione

- Un **linguaggio di programmazione** rappresenta un **elaboratore astratto**, capace di comprendere i termini del linguaggio, che generalmente sono astrazioni degli oggetti della macchina reale

Ad esempio una variabile è una astrazione della memoria centrale

Dati semplici e strutturati

- I dati **semplici** o **elementari** rappresentano oggetti, entità, proprietà **elementari** o **atomiche** non decomponibili
- I dati **strutturati** rappresentano oggetti, entità, proprietà più complessi, sono aggregati di dati elementari

Identificatori di variabili e costanti

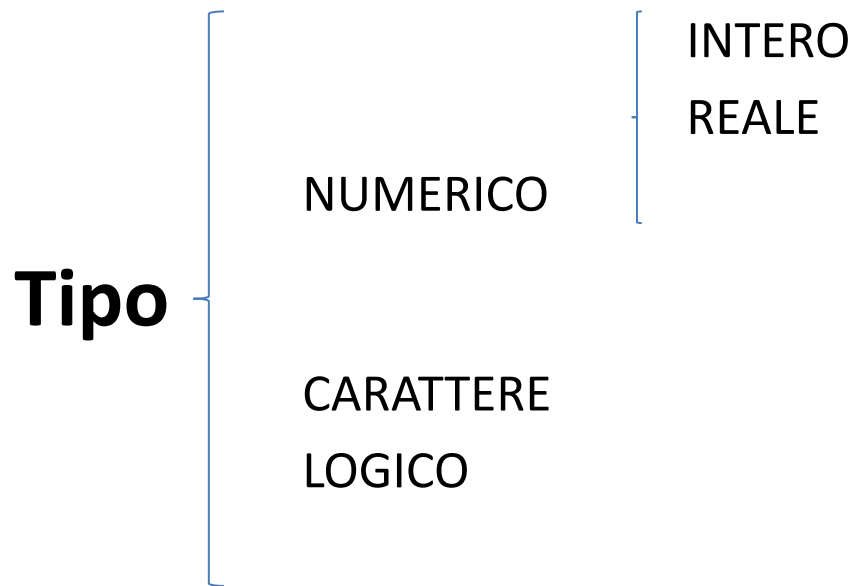
- Una **variabile** è identificata da un nome, detto **identificatore**, ed assume un **valore** che può essere modificato durante l'esecuzione di un programma
- Una **costante** è identificata da un nome, detto **identificatore**, ed assume un **valore** che **non** può essere modificato durante l'esecuzione di un programma

Tipo di dato

- Ad ogni dato, **variabile** o **costante**, deve essere associato un **tipo**
- Generalmente tale associazione è dichiarata nella **sezione dichiarativa del programma**

Tipi di dato

- I tipi sono: numerici, logici, alfanumerici o caratteri



Tipo di dato

- Un tipo di dato o semplicemente **tipo** indica:
 - Un insieme di valori D (**dominio**)
 - Un insieme di operazioni O che si applicano agli elementi di D – **operatori e predicati**
 - Un insieme di **costanti**, cioè di simboli con cui si denotano i valori del dominio D

Dichiarazione di una variabile

- Nella dichiarazione di una variabile **a** è necessario associare ad **a** un tipo, per indicare:
 - I valori che la variabile può assumere
 - Le operazioni che si possono effettuare sull'insieme di tali valori
 - I simboli con cui si indicano tali valori all'interno di un programma
- Esempio di dichiarazione in C: **int a;**

Costanti

- Esempio di dichiarazione in C di una costante intera identificata da n

const int n = 100;

- const parola chiave
 - int specificatore di tipo
 - 10 valore assegnato alla costante n
- Una dichiarazione di costante associa permanentemente un **identificatore** ad un **valore**

Tipi e linguaggi di programmazione

- I linguaggi di programmazione consentono di:
 - Utilizzare tipi di dato **predefiniti**
 - Definire **nuovi tipi**
 - Convertire **una variabile di un tipo in una variabile di altro tipo** in modo implicito o esplicito

Tipi primitivi

- I tipi **primitivi** sono quelli direttamente disponibili nel linguaggio di programmazione. In C i tipi primitivi sono:
 - Tipi predefiniti:
 - **int**: per i numeri interi
 - **float**: per i numeri reali
 - **double**: per i reali in doppia precisione
 - **char**: per i caratteri
 - Tipi definiti dall'utente:
 - Tipi **enumerati** definiti tramite la parola chiave enum

Tipi semplici e tipi strutturati

- I tipi primitivi sono detti anche **tipi semplici** intendendo che il loro **dominio è elementare**, cioè costituito d elementi atomici non decomponibili (es. int)
- I tipi strutturati sono quelli il cui **dominio è composito**, cioè composto aggregando dati più elementari (es. vettore e struttura)

Dati numerici interi e reali

- I linguaggi di programmazione consentono di rappresentare ed elaborare dati numerici interi e reali
- Il linguaggio C fornisce 3 tipi predefiniti:
 - **int** per rappresentare ed elaborare un sottoinsieme dei numeri interi
 - **float** e **double** per rappresentare ed elaborare un sottoinsieme dei numeri reali

Tipo int

- I dati di tipo **int** costituiscono un sottoinsieme dei numeri interi, positivi e negativi
- Un **int** è rappresentato generalmente in una parola – **2 byte o 4 byte**
- Nel caso in cui la parola sia di 2 byte il dominio del tipo **int** è costituito dall'insieme
$$D = [-2^{15}, 2^{15} - 1] = [-32768, 32767]$$

Tipo int

- Il minimo e il massimo valore intero hanno una rappresentazione simbolica denominata **INT_MIN** e **INT_MAX**
- **INT_MIN** e **INT_MAX** sono identificatori di costanti predefinite e contengono il valore del più piccolo e del più grande intero rappresentabile
- La definizione delle 2 costanti è contenuta nel file **limits.h**

Tipo int – operatori aritmetici

Siano a e b 2 operandi di tipo int, gli operatori aritmetici restituiscono un valore di tipo int

+ **Addizione** $a+b$

- **Sottrazione** $a-b$

***** **Moltiplicazione** $a*b$

/ **Divisione** a/b (restituisce il quoziente della divisione, la parte intera)

% **Modulo** $a\%b$ (restituisce il resto della divisione)

Tipo int – operatori relazionali

Restituiscono un valore vero (1) o falso (0)

==	Uguale
!=	Diverso
<	Minore
>	Maggiore
<=	Minore uguale
>=	Maggiore uguale

Costanti intere

- I valori del dominio del tipo `int` sono denotati, all'interno di un programma, con un insieme di valori costanti (costanti intere)
- Ogni costante intera è rappresentata come un numero intero decimale eventualmente con segno
- Esempi: `0`, `-151`, `0`, `+41`

Espressione aritmetica

- Una espressione aritmetica intera è costituita da termini separati dagli operatori + e –
- Ogni termine è costituito da fattori, variabili e costanti, separati dagli operatori *, /, %

Esempi

`int a, b, c; /* dichiarazione di variabili intere */`

`a=4, b=2; /* espressione costituita da un solo
termine costante */`

`c=b/a-1; /* espressione costituita da 2 termini
*/`

`c=b%a; /* espressione costituita da un termine
*/`

In ogni caso il risultato è un intero

Compatibilità tra tipi

- Siano T1 e T2 due tipi differenti

Variabili di tipo differente possono comparire in una stessa espressione?

Generalmente **si**, se il linguaggio di programmazione fornisce delle **regole di conversione implicite** tra i due tipi