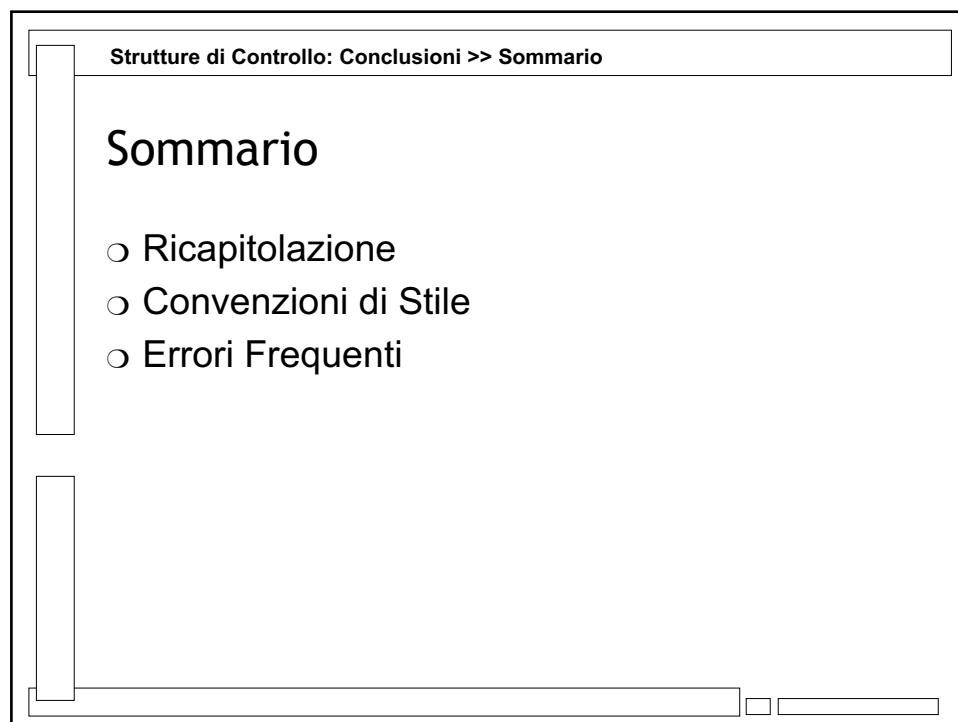




1



2

Strutture di Controllo: Conclusioni >> Ricapitolazione

Ricapitolazione

- Strutture di controllo
 - ⇒ elementi fondamentali della programmazione procedurale
 - ⇒ consentono di eseguire o meno una porzione di programma sulla base di condizioni
- Due categorie
 - ⇒ istruzioni condizionali
 - ⇒ istruzioni iterative o cicli

3

Strutture di Controllo: Conclusioni >> Ricapitolazione

Ricapitolazione

- Istruzioni condizionali
 - ⇒ servono ad eseguire una porzione oppure un'altra sulla base del valore di una condizione
- Condizione
 - ⇒ espressione a valori booleani
- Due istruzioni (in tutti i linguaggi)
 - ⇒ IF-THEN-ELSE
 - ⇒ IF-THEN

4

Ricapitolazione

- Istruzioni iterative
 - ⇒ servono ad eseguire più volte una porzione di programma sulla base del valore di una condizione
- Due istruzioni (in tutti i linguaggi)
 - ⇒ WHILE (cicli “aperti” oppure cicli “chiusi”)
 - ⇒ FOR (cicli “chiusi” o a conteggio)

5

Convenzioni di Stile

- In questo corso (e nei successivi)
 - ⇒ convenzioni sullo “stile di scrittura” del codice
- Obiettivi
 - ⇒ uniformare il modo di scrivere il codice
 - ⇒ per renderlo più leggibile
 - ⇒ e per favorire la collaborazione nei gruppi di lavoro
- Tendenza internazionale

6

Strutture di Controllo: Conclusioni >> Convenzioni di Stile

Convenzioni di Stile

- Nelle lezioni successive
 - ⇒raffineremo progressivamente le nostre regole
 - ⇒in modo che valgano anche dopo (Java, C#)
- Per ora
 - ⇒a) regole sull'indentazione del codice
 - ⇒b) regole sulle parentesi graffe
 - ⇒c) regole sulle espressioni
 - ⇒d) regole sui nomi di variabili

7

Strutture di Controllo: Conclusioni >> Convenzioni di Stile

Convenzioni di Stile

- a) Indentazione
 - ⇒codice di un programma: insieme di blocchi nidificati (es: strutture di controllo)
- Per rendere più leggibile il codice
 - ⇒è necessario dentellarlo (o "indentarlo")
 - ⇒le istruzioni di ogni blocco sono rientrate rispetto alle parentesi
 - ⇒molto importante
- Regola: rientri di 4 spazi

8

Strutture di Controllo: Conclusioni >> Convenzioni di Stile

Codice senza Indentazione

```
#include <iostream>
using namespace std;
int main() {
float voto, sommaVoti, media;
int numeroEsami;
bool continua;
continua = true;
numeroEsami = 0;
sommaVoti = 0.0;
while (continua == true) {
cout << "Immetti il voto (0 per fermarti)" << endl;
cin >> voto;
if (voto == 0)
continua = false;
else {
numeroEsami++;
sommaVoti += voto;
}
}
if (numeroEsami > 0) {
media = sommaVoti/numeroEsami;
cout << "La media vale: " << media << endl;
} else
cout << "Non hai fornito nessun voto" << endl;
return 0;
}
```

9

Strutture di Controllo: Conclusioni >> Convenzioni di Stile

Codice senza Indentazione

```
#include <iostream>
using namespace std;
int main() {
float voto, sommaVoti, media;
int numeroEsami;
bool continua;
continua = true;
numeroEsami = 0;
sommaVoti = 0.0;
while (continua == true) {
cout << "Immetti il voto (0 per fermarti)" << endl;
cin >> voto;
if (voto == 0)
continua = false;
else {
numeroEsami++;
sommaVoti += voto;
}
}
if (numeroEsami > 0) {
media = sommaVoti/numeroEsami;
cout << "La media vale: " << media << endl;
} else
cout << "Non hai fornito nessun voto" << endl;
return 0;
}
```

10

Convenzioni di Stile

○ b) Posizione delle parentesi graffe

⇒ due convenzioni principali

Convenzione n.1

```
if (condizione) {
    <istruzioni>;
}
if (condizione) {
    <istruzioni>;
} else {
    <istruzioni>;
}
```

Convenzione n.2

```
if (condizione)
{
    <istruzioni>;
}
else
{
    <istruzioni>;
}
```

11

Convenzioni di Stile

○ Regola n.1

⇒ adotteremo la convenzione n.1

○ Regola n.2

⇒ utilizzeremo sempre le parentesi

⇒ anche nel caso in cui un blocco contenga un'unica istruzione

12

Convenzioni di Stile

○ c) Scrittura delle espressioni

- ⇒ espressione: fatta di operandi ed operatori
- ⇒ per renderla più leggibile è possibile inserire spazi tra operandi ed operatori

○ Esempio

- ⇒ `(sqrt(x) + (y-z)) == k`
- ⇒ `(sqrt(x) + (y - z)) == k`

13

Convenzioni di Stile

○ d) Nomi delle variabili

- ⇒ identificatori
- ⇒ regola n.1: iniziale minuscola (es: voto, continua)
- ⇒ regola n.2: convenzione "cammello"

○ Convenzione cammello

- ⇒ identificatore fatto di più parole concatenate
- ⇒ ogni iniziale a parte la prima è maiuscola
- ⇒ es: sommaVoti, numeroEsami

14

Strutture di Controllo: Conclusioni >> Convenzioni di Stile

```
#include <iostream>
using namespace std;
int main() {
    int voto, sommaVoti, numeroEsami;
    float media;
    bool continua;
    continua = true;
    numeroEsami = 0;
    sommaVoti = 0;
    while (continua == true) {
        cout << "Immetti il voto (0 per fermarti)" << endl;
        cin >> voto;
        if (voto == 0) {
            continua = false;
        } else {
            numeroEsami++;
            sommaVoti += voto;
        }
    }
    if (numeroEsami > 0) {
        media = ((float)sommaVoti) / numeroEsami;
        cout << "La media vale: " << media << endl;
    } else {
        cout << "Non hai fornito nessun voto" << endl;
    }
    return 0;
}
```

a) rientri di 4 spazi
NO più istruzioni per linea

b) posizione delle parentesi
NO convenzione n. 2
+ parentesi sempre

c) espressioni
NO (continua==true)

d) nomi delle variabili
NO numeroesami
NO somma_voti

15

Strutture di Controllo: Conclusioni >> Errori Frequenti

Errori Frequenti

- Un errore sintattico frequente
 - ⇒ parentesi graffe mancanti per i blocchi delle istruzioni di controllo
- Due casi
 - ⇒ entrambe le parentesi mancanti
 - ⇒ solo la parentesi finale è mancante

16

Un Esempio: Media con For

```
#include <iostream>
using namespace std;
int main() {
    int voto, sommaVoti;
    float media;
    int numeroEsami, i;
    cout << "Quanti esami hai sostenuto ?" << endl;
    cin >> numeroEsami;
    sommaVoti = 0;
    for (i = 0; i < numeroEsami; i++) {
        cout << "Immetti il voto di un esame" << endl;
        cin >> voto;
        sommaVoti += voto;
    }
    if (numeroEsami > 0)
        media = ((float)sommaVoti) / numeroEsami;
    cout << "La media vale: " << media << endl;
    else {
        cout << "Non hai fornito nessun voto" << endl;
    }
    return 0;
}
```

manca la parentesi di chiusura

mancano le parentesi del I blocco

17

Errori Frequenti

- Nel primo caso (parentesi di chiusura)
 - ⇒ il compilatore considera il blocco aperto fino alla fine del file e alla fine del file segnala l'errore
 - ⇒ mediaFor.cpp :28:1:error: expected `}' at the end of input
 - ⇒ "Blocco di istruzioni senza parentesi chiusa"
- Nel secondo caso (parentesi dell'if)
 - ⇒ il compilatore considera il blocco fatto di una sola istruzione; errore quando incontra "else"
 - ⇒ mediaFor.cpp :24:6: error: `else' without a previous `if'
 - ⇒ "Else fuori posto"

18

Errori Frequenti

- Un errore logico frequente
 - ⇒ nei cicli a conteggio programmati con while, gestione scorretta della variabile di ciclo
 - ⇒ mancanza dell'inizializzazione
 - ⇒ mancanza dell'incremento
- In questi casi
 - ⇒ il ciclo si comporta in modo scorretto
 - ⇒ nel secondo caso si verifica un loop

19

Errori Frequenti

- Manca inizializzazione
 - ⇒ il comportamento è casuale
 - ⇒ dipende dal valore di i
- Manca l'incremento
 - ⇒ tipicamente si verifica un "loop"

```
int i;
while (i < 5) {
    cout << "Valore di i: ";
    cout << i << endl;
    i++;
}
```

```
int i;
i = 0;
while (i < 5) {
    cout << "Valore di i: ";
    cout << i << endl;
}
```

20

Strutture di Controllo: Conclusioni >> Sommario

Riassumendo

- Ricapitolazione
- Convenzioni di Stile (ATTENZIONE)
- Errori Frequenti

21

Termini della Licenza

Termini della Licenza

- This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
- Questo lavoro viene concesso in uso secondo i termini della licenza “Attribution-ShareAlike” di Creative Commons. Per ottenere una copia della licenza, è possibile visitare <http://creativecommons.org/licenses/by-sa/1.0/> oppure inviare una lettera all’ indirizzo Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

22