



CORSO DI PROGRAMMAZIONE

ISTRUZIONI DI SELEZIONE NIDIFICATE

DISPENSA 02.02

[02-02_Annidare_Decisioni_\[ver_15\]](#)



Questa dispensa è rilasciata sotto la licenza Creative Common CC BY-NC-SA. Chiunque può copiare, distribuire, modificare, creare opere derivate dall'originale, ma non a scopi commerciali, a condizione che venga riconosciuta la paternità dell'opera all'autore e che alla nuova opera vengano attribuite le stesse licenze dell'originale.

Versione del: **07/11/2015**

Revisione numero: **15**

Prof. Andrea Zoccheddu
Dipartimento di Informatica

**DIPARTIMENTO
INFORMATICA E TELECOMUNICAZIONI**





ISTRUZIONI DECISIONALI NIDIFICATE

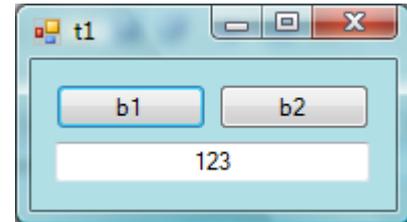
ANNIDARE ISTRUZIONI CONDIZIONALI

ISTRUZIONE SEQUENZA (BLOCCO)

PROGETTO GUIDATO

- prepara un form1 simile alla figura
- associa al pulsante button1 il seguente gestore di evento:

```
int num = Convert.ToInt16(textBox1.Text);  
if ( num > 10 )  
{  
    Top = Top + 10;  
    Left = Left + 10;  
    Text = Text + "x";  
    BackColor = Color.FromArgb(num, 1, 1);  
}
```



- associa al pulsante **button2** il seguente gestore di evento:

```
int num = Convert.ToInt16(textBox1.Text);  
if ( num > 10 )  
{  
    //blocco then  
    button1.Top = button1.Top + 10;  
    button1.Left = button1.Left + 10;  
    button1.Text = button1.Text + "x";  
    button1.BackColor = Color.FromArgb(num, 1, 1);  
}  
else  
{  
    //blocco else  
    button2.Top = button2.Top + 10;  
    button2.Left = button2.Left + 10;  
    button2.Text = button2.Text + "x";  
    button2.BackColor = Color.FromArgb(1, num, 1);  
}
```

- prova a eseguire il progetto

ISTRUZIONE SEQUENZA (BLOCCO)

Abbiamo visto in precedenza che i gestori di evento hanno un corpo, racchiuso tra parentesi graffe. E, osservando il programma, possiamo notare che ci sono numerose parentesi graffe aperte e chiuse. Ogni coppia di parentesi (che c# propone incolonnate correttamente) corrisponde a un elemento di programmazione che nella terminologia C si chiama «**blocco**».

L'istruzione sequenza è un caso particolare di blocco (vedremo altri casi leggermente differenti) che consente di raggruppare molte altre istruzioni.

Quando occorre mettere insieme delle istruzioni, perché formino un gruppo unitario, si può ricorrere al blocco.

Nell'esempio introduttivo abbiamo usato dei blocchi sia nel ramo **then** che nel ramo **else** dell'istruzione **if**; in verità è possibile usare un blocco anche in altre istruzioni. Nel caso dell'**if** sappiamo che ogni ramo (then



ed else) consente di utilizzare una sola istruzione; ma se occorre eseguire più di un'istruzione, è possibile superare l'ostacolo sfruttando il blocco.

SINTASSI DEL BLOCCO

Banalmente la sintassi del blocco è la seguente:

```
{  
  istruzione_1;  
  istruzione_2;  
  . . .  
  istruzione_K;  
}
```

dove si possono inserire un numero finito K di istruzioni.

Oltre alle istruzioni è possibile anche utilizzare delle dichiarazioni, come già illustrato nell'esempio introduttivo, per cui la sintassi può assumere la forma:

```
{  
  dichiarazione_A;  
  istruzione_1;  
  dichiarazione_B;  
  istruzione_2;  
  . . .  
  dichiarazione_Z;  
  istruzione_K;  
}
```

VITA DELLE VARIABILI LOCALI

Come già discusso in occasione dei gestori di evento, le variabili dichiarate in un blocco, iniziano ad esistere nel momento in cui si elabora la dichiarazione e continuano la loro esistenza fino ad incontrare la fine del blocco; quando l'elaborazione raggiunge la fine del blocco, la variabile è distrutta e la memoria occupata per essa, sarà liberata.

Vediamo un esempio, con una variabile x che ha vita solo nel blocco, ma non può essere usata al di fuori del blocco stesso:

```
int y = 17;  
{  
  int x = 13;  
  y = x + y; ;  
}  
//dopo la graffa chiusa, y esiste, ma x non esiste più!!!  
x = 0; //solleva un errore!!!
```

Un altro esempio analogo potrebbe essere:

```
int esterna = 100;  
if (esterna > 10 )  
{  
                                     //blocco then  
  int interna = 17;  
  esterna = interna + 13;  
}
```



```

else
{
//blocco else
    esterna = interna - 13;
}

```

dove il ramo else della istruzione solleva un errore, poiché la variabile `interna` non esiste più!

ANNIDARE IF

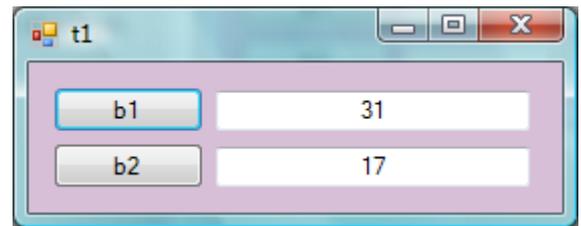
PROGETTO GUIDATO

- prepara un form1 simile alla figura
- associa al pulsante **button1** il seguente gestore di evento:

```

int a = Convert.ToInt16(textBox1.Text);
int b = Convert.ToInt16(textBox2.Text);
if (a > 10)
    if (b > 10)
        Text = "entrambe maggiori";
    else
        Text = "solo a maggiore";
else
    Text = "b non lo so";

```



- associa al pulsante **button2** il seguente gestore di evento:

```

int a = Convert.ToInt16(textBox1.Text);
int b = Convert.ToInt16(textBox2.Text);
if (a >= b)
    if (b >= a)
        Text = "a == b";
    else
        Text = "a > b";
else
    Text = "a < b";

```

- prova a eseguire il progetto

RIFLESSIONI SUL PROGETTO GUIDATO

Sappiamo che l'istruzione `if` consente di scrivere un'altra sua istruzione all'interno, sia nel ramo `then` che nel ramo `else`. Tra le istruzioni che possiamo scegliere (oltre a assegnazione e blocco) c'è una ulteriore istruzione `if`. Quando si inserisce un'istruzione `if` dentro un'istruzione `if`, si parla di annidamento.

Un annidamento appare quindi come per esempio:

```

if (Condizione_1)
    if (Condizione_2)
        Istruzione_A;

```

dove l'istruzione A è eseguita solo se sono vere entrambe le condizioni 1 e 2.

Come appare negli esempi, è possibile annidare `if` sia nei rami `then` che nei rami `else` ed è possibile addirittura annidarne ulteriormente con tre o più livelli di annidamento.



ESERCIZI

ESERCIZI

ESERCIZIO 1. CONFRONTI DI TRE VALORI

- prepara un form1 simile alla figura
- il pulsante **massimo** deve visualizzare nella etichetta il massimo valore delle tre caselle
- il pulsante **interno** deve visualizzare nella etichetta il valore intermedio delle tre caselle
- il pulsante **minimo** deve visualizzare nella etichetta il minimo valore delle tre caselle

massimo	31
interno	17
minimo	17

label1

ESERCIZIO 2. CONDIZIONI MULTIPLE

- prepara un form1 simile alla figura
- il pulsante **tutti** deve colorare di rosso l'etichetta se tutti i valori delle caselle sono uguali, altrimenti di verde
- il pulsante **nessuno** deve colorare di nero l'etichetta se tutti i valori delle caselle sono diversi, altrimenti di bianco
- il pulsante **alcuni** deve colorare di blu l'etichetta se due soli valori delle caselle sono uguali, altrimenti di giallo

tutti	31
alcuni	17
nessuno	17

label1

ESERCIZIO 3. GIOCO DEI DADI

- Si desidera realizzare un gioco con le seguenti regole:
- il giocatore scrive una puntata nella casella e preme il pulsante, che simula il lancio di tre dadi (a sei facce):
- se tutti i dadi sono identici, il giocatore vince il doppio della posta; se almeno due dadi sono identici, si pareggia, se sono tutti diversi il giocatore perde la posta.
- nella etichetta in basso si riporta la situazione (positiva o negativa)

Puntata: euro

dado1 dado2 dado3

Situazione: 0 euro

**ESERCIZIO 4. ALEA TRACTA EST**

- Si desidera realizzare un gioco con le seguenti regole:
- il giocatore scrive una puntata nella casella e preme il pulsante, che simula il lancio di tre dadi (a sei facce):
- se tutti i dadi sono pari si vince il valore della posta
- se tutti i dadi sono dispari si vince il doppio della posta
- altrimenti si perde la posta
- nella casella in basso si riporta il capitale posseduto: se arriva a zero il gioco finisce.

**ESERCIZIO 5. PROGRAMMA COMMERCIALE**

- Preparare un nuovo progetto per un programma di acquisto di frutta
- Il prezzo delle Arance dipende dalla quantità acquistata:
 - Fino a 50 Kg è di 1 euro / Kg
 - Fino a 100 Kg è di 0,75 euro / Kg
 - Oltre 100 Kg è di 0,55 euro / Kg
- Il prezzo delle Mele dipende dalla quantità acquistata:
 - Fino a 75 Kg è di 0,90 euro / Kg
 - Fino a 150 Kg è di 0,70 euro / Kg
 - Oltre 150 Kg è di 0,50 euro / Kg
- Il prezzo delle Pere dipende dalla quantità acquistata:
 - Fino a 100 Kg è di 1,50 euro / Kg
 - Fino a 200 Kg è di 1,25 euro / Kg
 - Oltre 200 Kg è di 1,05 euro / Kg
- Il button1 calcola l'importo e lo visualizza nella label5



**ESERCIZIO 6. GIOCO DELLE CARTE**

- Si desidera realizzare un gioco con le seguenti regole:
- Il giocatore scrive una puntata nella casella e preme il pulsante, che simula l'estrazione di tre carte (da 1 a 10 con semi ♥ ♦ ♣ ♠)
- Se le carte sono tutte dello stesso seme si pareggia (non si perde e non si vince)
- Se le carte sono tutte dello stesso valore si vince il doppio della posta
- Se le carte sono tutte dello stesso valore e dello stesso seme si vince il quadruplo della posta
- Negli altri casi si perde la puntata (decrementare il capitale posseduto mostrato in basso)
- Nella etichetta in basso si riporta la situazione (se arriva a zero euro il gioco finisce!!!)
- Migliorare l'estetica del gioco nel seguente modo:
 - ↳ Se la carta è di cuori o quadri allora il testo (ForeColor) delle caselle di testo diventa rosso;
 - ↳ Se la carta è di picche o fiori allora il testo (ForeColor) delle caselle di testo diventa nero;

**ESERCIZIO 7. ACCOPPIA O T'ACCOPPA**

- Si desidera realizzare un gioco con le seguenti regole:
- Il giocatore inizia con 1000 euro e può fare clic su uno dei 9 pulsanti per volta
- Ogni pulsante mostra una carta quando viene cliccato
- Appena compaiono due carte identiche (valore e seme) si vincono 100 euro e compare un messaggio col capitale attuale
- Se le carte finiscono si perdono 10 euro e compare un messaggio col capitale attuale
- In ogni caso il gioco ricomincia con 9 pulsanti senza scritte fino a quando il giocatore smette oppure perde tutti i soldi

**ESERCIZIO 8. SERRA**

- Si desidera realizzare la simulazione di una serra automatica
- L'utente preme il pulsante che genera tre valori casuali
- La temperatura deve essere compresa tra i -10 e i +50 gradi Celsius
- L'umidità deve essere compresa tra lo 0% e il 100%
- La pressione deve essere compresa tra lo 500 e 2000 bar





- ➔ Ora si controlla la situazione:
- ➔ La temperatura è considerata normale quando è compresa tra i 18° e i 25° gradi
- ➔ L'umidità è considerata normale quando è compresa tra il 25% e il 75%
- ➔ La pressione è considerata normale quando è compresa tra 1000 e 1200
- ➔ La situazione si considera:
 - ↪ NORMALE se tutti i tre parametri sono normali colore VERDE
 - ↪ ANORMALE se un parametro non è normale colore GIALLO
 - ↪ PERICOLOSA se un solo parametro è normale colore ARANCIO
 - ↪ DELETERIA se tutti i tre parametri sono NON normali colore ROSSO
- ➔ Scrivere nella etichetta la dicitura della situazione con il colore corretto

ESERCIZIO 9. POKER SOLITARIO

- ➔ Si desidera realizzare un gioco con le seguenti regole:
- ➔ Il giocatore punta da 1 a 5 euro poi preme il pulsante PLAY
- ➔ Il pulsante estrae 5 carte; e si usano le combinazioni del poker
- ➔ La **coppia** non fa vincere né perdere
- ➔ La **doppia coppia** fa vincere metà della posta divisa per difetto
- ➔ Il **tris** fa vincere la posta
- ➔ Una **scala** fa vincere fa vincere una volta e mezzo la posta
- ➔ Il **full** fa vincere il doppio della posta
- ➔ Il **colore** fa vincere il triplo della posta
- ➔ Il **poker** fa vincere fa vincere il quadruplo della posta
- ➔ La **scala reale** fa vincere fa vincere il decuplo della posta
- ➔ Altrimenti si perde la posta
- ➔ Nella etichetta in basso si riporta la situazione (se arriva a zero euro il gioco finisce!!!)
- ➔ Migliorare il gioco nel seguente modo:
 - ↪ Non possono mai uscire due carte identiche
 - ↪ Usare le lettere A per l'Asso, K per il re, Q per la donna, J per il fante.
 - ↪ Il carattere ♥ compare premendo Alt-Tastnum-3; analogamente per ♦♣♠





SOMMARIO

ANNIDARE ISTRUZIONI CONDIZIONALI 2

ISTRUZIONE SEQUENZA (BLOCCO) 2

 Progetto guidato..... 2

 Istruzione sequenza (blocco)..... 2

 Sintassi del blocco 3

 Vita delle variabili locali 3

ANNIDARE IF 4

 Progetto guidato..... 4

 Riflessioni sul progetto guidato 4

ESERCIZI 5

 Esercizio 1. Confronti di tre valori 5

 Esercizio 2. Condizioni multiple..... 5

 Esercizio 3. Gioco dei dadi..... 5

 Esercizio 4. Alea tracta est 6

 Esercizio 5. Programma commerciale 6

 Esercizio 6. Gioco delle carte 7

 Esercizio 7. Accoppia o t'accoppa 7

 Esercizio 8. Serra..... 7

 Esercizio 9. Poker solitario..... 8