

Code kata

Record

Indice generale

1	Generare un elenco di record.....	3
2	Generare un elenco di record da una matrice.....	4
3	Modificare i dati di un vettore.....	5
4	Filtrare i dati di un vettore/lista.....	6
5	Filtrare i dati di un vettore/lista (2).....	7
6	Ricerca di un elemento in un vettore.....	8
7	Ordinamento di un vettore.....	9

1 Generare un elenco di record

Problema

Dato il tipo `Atleta`, generare un vettore di atleti.

```
class Atleta
{
    public string Nominativo;
    public double Altezza;
    public double Peso;
}
```

Esempio:

```
Atleta[] atleti = GetAtleti(); // ->
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}
```

Variazioni

Generare una lista di record.

Aggiungere gli atleti a una lista esistente (globale).

2 Generare un elenco di record da una matrice

Problema

Sia data una matrice di stringhe Nx2, che rappresenta un dizionario. La prima colonna memorizza le "voci", la seconda colonna memorizza le "descrizioni" corrispondenti.

Creare un vettore di record del seguente tipo:

```
class VoceDizionario
{
    public string Voce;
    public string Descrizione;
}
```

Esempio:

```
string[,] dizionario =
{
    {"IDE", "Ambiente di sviluppo integrato"},
    {"Debugging", "Eliminazione degli errori"},
    {"...", "..."}
};
//->
// {Voce:"IDE"; Descrizione:"Ambiente di sviluppo integrato"}
// {Voce:"Debugging"; Descrizione:"Eliminazione degli errori"}
// {Voce:"..."; Descrizione:"..."}
```

Variazioni

Memorizzare l'elenco in una lista.

Aggiungere le voci a una lista esistente (globale).

3 Modificare i dati di un vettore

Problema

Dato il tipo `Ordine`:

```
class Ordine
{
    public string Descrizione;
    public double CostoUnitario;
    public int Quantità;
}
```

e dato un vettore di ordini, diminuire del 10% il costo unitario. (Modificare direttamente gli ordini del vettore.)

Variazioni

Restituire gli ordini modificati in un nuovo vettore, senza modificare il vettore originale.

Restituire gli ordini in una lista, senza modificare il vettore originale.

4 Filtrare i dati di un vettore/lista

Problema

Data una lista di atleti, restituire una nuova lista contenente solo quelli più alti di 1,80.

Esempio

```
List<Atleta> atleti = GetAtleti(); //  
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}  
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}  
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}  
  
//-> {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
```

Variazioni

Generalizzare il procedimento per un'altezza qualsiasi.

Supporre che gli atleti siano memorizzati in un vettore; restituire un vettore.

Requisiti (necessari / utili) per la soluzione

Trasformare una lista in un vettore; metodo `ToArray()`.

5 Filtrare i dati di un vettore/lista (2)

Dato una lista di elementi di tipo `Atleta`:

```
class Atleta
{
    public string Nominativo;
    public double Altezza;
    public Sesso Sesso;
}

enum Sesso
{
    Uomo,
    Donna
}
```

Restituire una lista delle donne.

Variazioni

Restituire una lista degli atleti del sesso specificato.

Restituire un vettore del sesso specificato.

6 Ricerca di un elemento in un vettore

Problema

Dati un vettore di atleti e il nominativo di un atleta, restituire la sua posizione nel vettore. Restituire -1 se non esiste un atleta con quel nominativo.

Esempio:

```
Atleta[] atleti = GetAtleti();
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}
string nominativo = "Verdi, Sonia";
// -> 1
```

Variazioni

Fornire versioni del procedimento con il ciclo `for`, `foreach` e `while`.

Restituire `true` se esiste, `false` se non esiste.

Restituire l'atleta (l'oggetto), oppure `null` se non viene trovato.

7 Ordinamento di un vettore

Problema

Dati un vettore di atleti, ordinarlo in base al nominativo.

Esempio:

```
Atleta[] atleti = GetAtleti();
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}

//->
    {Nominativo:"Bianchi, Filippo"; Altezza:1,77; Peso:68}
    {Nominativo:"Rossi, Andrea"; Altezza:1,91; Peso:92}
    {Nominativo:"Verdi, Sonia"; Altezza:1,70; Peso:57}
```

Variazioni

Ordinare il vettore in base all'altezza.