

Analisi: 10Chess_x

Analisi del problema:

Il problema chiede di creare un programma che simuli il gioco degli scacchi. Basterà definire la classe **Pedina** da cui deriveranno le classi rappresentanti i pezzi: **Pedone**, **Alfiere**, **Cavallo**. Il campo può essere rappresentato da una matrice di oggetti **Pedina** in cui le varie caselle saranno riempite con gli oggetti che raffigurano i vari pezzi degli scacchi. I vari oggetti hanno lo stesso comportamento che però varia a seconda del pezzo. Infatti ogni pezzo può spostarsi sulla scacchiera in maniera differente.

Classe Astratta: Pedina

Attributi della classe:

Nome Attributo	Tipo	Descrizione e utilizzo:
<i>colore</i>	bool.	Attributo che indica il colore della pedina: true=nero false=bianco.
<i>posVert</i>	Intero a 16 bit.	Attributo che contiene la posizione verticale della pedina nella scacchiera.
<i>posOriz</i>	Intero a 16 bit.	Attributo che contiene la posizione orizzontale della pedina nella scacchiera.

Costruttore della classe:

- **Firma:** public **Pedina**(bool **colore**, int **posVert**, int **posOriz**)
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori per ogni oggetto delle **classi derivate**. Sarà possibile passare un booleano per indicare il colore dell'oggetto pedina e due valori interi che indicano le coordinate della pedina nel campo.

Metodi della classe:

Metodi Get e Set:

- **Firme:**
 - public void **SetColore**(bool **colore**)
 - public bool **GetColore**()
 - public void **SetPosVert**(int **posVert**)
 - public int **GetPosVert**()
 - public void **SetPosOriz**(int **posOriz**)
 - public int **GetPosOriz**()
- **Descrizione:** i metodi Get e Set permettono di modificare(Set) e di recuperare(Get) i dati presenti negli attributi dell'oggetto.

Metodo Sposta:

- **Firme:** public abstract bool **Sposta**(int **posVert**, int **posOriz**)
- **Descrizione:** il metodo Sposta permette all'oggetto pedina di cambiare la sua posizione con le coordinate, ritornerà un booleano che indicherà se lo spostamento della pedina può essere fatto o meno.

Classe: Torre →Pedina

Costruttore della classe:

- **Firma:**
`public Torre(bool colore, int posVert, int posOriz) : base(colore, posVert, posOriz)`
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori dell'oggetto.

Metodo Sposta:

- **Firme:** `public override bool Sposta(int posVert, int posOriz)`
- **Descrizione:** il metodo **Sposta** sovrascrive il metodo della classe base **Pedina**, dove verrà implementato il metodo di spostamento di una torre degli scacchi.

Classe: Cavallo →Pedina

Costruttore della classe:

- **Firma:**
`public Cavallo(bool colore, int posVert, int posOriz) : base(colore, posVert, posOriz)`
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori dell'oggetto.

Metodo Sposta:

- **Firme:** `public override bool Sposta(int posVert, int posOriz)`
- **Descrizione:** il metodo **Sposta** sovrascrive il metodo della classe base **Pedina**, dove verrà implementato il metodo di spostamento di un cavallo degli scacchi.

Classe: Alfiere →Pedina

Costruttore della classe:

- **Firma:**
`public Alfiere(bool colore, int posVert, int posOriz) : base(colore, posVert, posOriz)`
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori dell'oggetto.

Metodo Sposta:

- **Firme:** `public override bool Sposta(int posVert, int posOriz)`
- **Descrizione:** il metodo **Sposta** sovrascrive il metodo della classe base **Pedina**, dove verrà implementato il metodo di spostamento di un'alfiere degli scacchi.

Classe: Re →Pedina

Costruttore della classe:

- **Firma:**
`public Re(bool colore, int posVert, int posOriz) : base(colore, posVert, posOriz)`
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori dell'oggetto.

Metodo Sposta:

- **Firme:** `public override bool Sposta(int posVert, int posOriz)`
- **Descrizione:** il metodo **Sposta** sovrascrive il metodo della classe base **Pedina**, dove verrà implementato il metodo di spostamento del re degli scacchi.

Classe: Regina → Pedina

Costruttore della classe:

- **Firma:**
`public Regina(bool colore, int posVert, int posOriz) : base(colore, posVert, posOriz)`
- **Descrizione:** Il costruttore della classe servirà ad inizializzare i valori dell'oggetto.

Metodo Sposta:

- **Firme:** `public override bool Sposta(int posVert, int posOriz)`
- **Descrizione:** il metodo **Sposta** sovrascrive il metodo della classe base **Pedina**, dove verrà implementato il metodo di spostamento della regina degli scacchi.

Classe: Form1

Attributi della classe:

Nome Attributo	Tipo	Descrizione e utilizzo
<i>scacchiera</i>	Matrice di oggetti Pedina.	Matrice contenente gli oggetti delle classi derivate dalla classe base Pedina.

Metodi della classe:

Metodo per la visualizzazione:

- **Firma:** `private void Display()`
- **Descrizione:** questo metodo controllerà ogni elemento della matrice **scacchiera**, se l'elemento contiene un oggetto, verrà visualizzato in un **DataGridView**.

Metodo conta:

- **Firma:** `private void Sposta()`
- **Descrizione:** questo metodo richiamerà il metodo **Sposta** della pedina selezionata che si vuole spostare.