

'Moore Rock'

- Definieren Sie eine Klasse 'window'. Sie dient als Superklasse und besitzt eine id, Methoden zum Anzeigen ('show') und Verstecken ('hide') sowie eine rein virtuelle Methode 'draw'.
- Leiten Sie öffentlich von 'window' eine Klasse 'button' und eine Klasse 'checkbox' ab. Beide definieren 'draw'.
- Erstellen Sie einen 'vector' von 'unique_ptr' und füllen Sie ihn mit je einer Instanz von 'button' und von 'checkbox'. Nutzen Sie 'make_unique'.
- Rufen Sie auf allen Elementen des Vektors 'draw' auf und testen Sie, ob die richtige Funktion aufgerufen wird.

'Union Beach' – Teil 1

- Implementieren Sie eine abstrakte Klasse 'matrix', die eine quadratische Matrix für ganze Zahlen darstellen soll. Sie besitzt rein virtuelle Methoden, um Elemente zu lesen und zu schreiben ('at'), sowie eine Methode, um die Matrix auf 0 zu setzen ('make_zero').
- Der ctor bekommt die Dimension 'dim' übergeben und ein Member der Klasse speichert sie.
- Die Klasse besitzt eine weitere, nicht virtuelle Funktion zum Setzen der Einheitsmatrix ('make_one'). Diese setzt die Matrix auf 0 und danach die Diagonalelemente auf 1.
- Wie die Daten gespeichert werden, entscheiden die abgeleiteten Klassen.

'Union Beach' – Teil 2

- Leiten Sie die Klasse 'full_matrix' von 'matrix' ab. Sie legt ihre Koeffizienten in einem dynamischen Feld von 'int' ab.
- Implementieren Sie die fehlenden Methoden entsprechend und nutzen Sie 'unique_ptr' für das Feld.
- Leiten Sie die Klasse 'sparse_matrix' von 'matrix' ab. Sie legt ihre Koeffizienten in einer 'unordered_map' ab. Der Schlüssel wird aus x und y zu $y \cdot \text{dim} + x$ gebildet, so dass man einen Eintrag findet, wenn er existiert. Sonst ist er 0.
- Schreiben Sie aussagekräftigen Testcode für beide Klassen.
- Erweitern Sie die Klassen effizient um Operatoren.
- Machen Sie aus der Klasse ein Template.

'Peters Mines' – Teil 1

- Definieren Sie eine Klasse 'befoerderungsmittel', die eine Anzahl von Sitzplätzen als Member enthält.
- Definieren Sie eine Klasse 'fahrzeug', die eine Höchstgeschwindigkeit als Member enthält.
- Leiten Sie die Klassen 'automobile' und 'boot' jeweils von 'befoerderungsmittel' und von 'fahrzeug' ab. Die Klasse 'befoerderungsmittel' soll virtuell geerbt werden.
- Implementieren Sie eine Klasse 'amphibie', die von 'boot' und von 'automobile' erbt. Instanzen sollten dann nur eine Anzahl Sitzplätze besitzen, aber zwei Höchstgeschwindigkeiten.

'Peters Mines' – Teil 2

- Übergeben Sie den Konstruktoren jeweils alle notwendigen Parameter, d.h. die Initialisierung einer 'amphibie' mit vier Sitzplätzen und zwei Geschwindigkeiten sieht so aus:
- 'amphibie a{4,120,15};'
- Schreiben Sie aussagekräftigen Testcode und geben Sie insbesondere die Sitzplätze und die Geschwindigkeiten aus.

Erweiterung:

- Ergänzen Sie 'automobile' und 'boot' um weitere Member, die auch jeweils den Konstruktoren übergeben werden.

Hausübung

- Überlegen Sie sich Beispiele, wo Mehrfachvererbung Sinn macht oder wo Sie es vielleicht schon einmal vermisst haben.

Selbstkontrolle

- Ich habe alle Codes und Übungsthemen verstanden.
- Ich kenne 'virtual' und den Unterschied zwischen C++ und Java diesbezüglich.
- Ich kann erklären, warum es keine Interfaces in C++ gibt.
- Ich weiß, was eine abstrakte Klasse ist.
- Ich habe das Thema Mehrfachvererbung in C++ verstanden.