

---

Arbeitsblatt zum Praktikum  
**Wissenschaftliches Rechnen**  
WS 2012/13 — Blatt 1

---

**Abgabe:** 23.10.2012 per E-Mail

**Aufgabe 1.1 (Gitter)**

Implementieren Sie ein eindimensionales Gitter `SimpleGrid`, welches auf einem Intervall  $[a, b]$ ,  $a, b \in \mathbb{R}$  in  $N$  äquidistante Teilintervalle der Länge  $\frac{b-a}{N}$  zerlegt werden kann. Bei der Umsetzung dieses Gitters ist darauf zu achten, dass sich das Konzept möglichst einfach auf komplexere Gitter (z.B. mehrdimensionale oder adaptive Gitter) übertragen lässt. Aus diesem Grund sollte das Gitter zumindest aus den folgenden Klassen bestehen:

**Geometry** Eine Geometrie-Klasse kennt die Anzahl der Ecken, die globalen Koordinaten der Ecken und kann lokale Koordinaten einer Referenzgeometrie auf diese Geometrie abbilden und umgekehrt. Des Weiteren kann Sie weitere nützliche Informationen bereitstellen, z.B. das Volumen der Geometrie.

**Entity** Eine Entität-Klasse sammelt alle Informationen, die zu einer Entität gehören, d.h. sie kann z.B. ein Objekt vom Typ `Geometry` zurückgeben.

**GridStorage** Diese Klasse speichert und verwaltet alle Entitäten, die später zum Gitter hinzugefügt werden sollen.

**LevelIterator** Diese Iterator-Klasse soll das Durchlaufen des Gitters ermöglichen. Dazu wird intern ein Objekt vom Typ `Entity*` gespeichert.  
Hinweis: Die Reihenfolge, in der die Entitäten des Gitters durchlaufen werden, sollen i.Allg. von der Anordnung der Entitäten im Gitter unabhängig sein.

**LevelIndexSet** Diese Klasse stellt eine Methode bereit, die jeder Entität eines Gitters eine eindeutige Nummer  $n \in \mathbb{N}_0$  zuweist. Diese Nummer soll konsekutiv sein, d.h. zwischen 0 und  $\# \text{Entitäten} - 1$  liegen.

**Grid** Die Gitter-Klasse bündelt alle Informationen. In erster Linie wird eine Menge von Entitäten gespeichert, die das Gitter darstellen. Weiter kann ein Iterator zurückgegeben werden, der über das Gitter iterieren kann und einen Zugriff auf die jeweilige Entität liefert. Des Weiteren kann ein Objekt vom Typ `LevelIndexSet` zurückgegeben werden.

Gehen Sie zur Implementierung des Gitters wie folgt vor:

- (a) Laden Sie die Quelldateien von der Vorlesungshomepage  
`http://wwwmath.uni-muenster.de/num/Vorlesungen/WissenschaftlichesRechnen\_WS1213/`  
herunter.
- (b) Erstellen Sie die Dokumentation, indem Sie den Befehl `make doc` eingeben.
- (c) Implementieren Sie das Gitter.
- (d) Erweitern Sie die Dokumentieren Ihres Programms mithilfe von `doxygen`, falls nötig.
- (e) Kompilieren Sie Ihr Programm, führen Sie es aus und kontrollieren Sie die Ausgabe.

Hinweis: Die Quelldateien auf der Homepage sind nur ein Vorschlag. Sollten Sie Ideen zur Verbesserung des Konzepts haben, können Sie diese gerne umsetzen.

**Schicken Sie die fertigen Programme an `stefan.girke@uni-muenster.de`.**