

Die Aufgaben können (müssen aber nicht) in Gruppen von bis zu drei Leuten bearbeitet werden. Falls die Aufgaben in einer Gruppe bearbeitet werden, genügt **eine** Abgabe mit den Namen aller Gruppenmitglieder. Abgabe bis **spätestens Freitag 02.07.2010** per Mail mit dem Betreff **PMSS2010 HA8 <Matrikelnummer>** (z. B. „PMSS2010 HA8 12345678“) an **pm@cs.uni-kassel.de**. Für diese Hausaufgabe gibt es 27 Punkte.

Hinweis zur Abgabe: Die Abgabe **MUSS** als exportiertes Eclipse Projekt erfolgen. Falls Sie mehrere Projekte anlegen, können diese alle in **eine** .zip Datei gepackt werden (erledigt die Eclipse Export Funktion automatisch!). Sind die Projekte nicht korrekt exportiert, können diese bei der Korrektur nicht berücksichtigt werden (es bietet sich also an, den Import des exportierten Projekts auszuprobieren).

WICHTIG Benennen Sie ihre Projekte nach folgendem Schema:

```
PMSS2010 HA8 A<i> <Matrikelnummer>,
```

wobei <i> für die Aufgabennummer steht. Beispiel:

```
PMSS2010 HA8 A1 12345678.
```

Storydiagramme

In dieser Hausaufgabe sollen die bisher von Hand implementierte Methoden durch Fujaba Storydiagramme ersetzt werden. Für diese Aufgabe können sie sowohl ihr eigenes Fujaba Projekt, als auch ein aktuelles vom PM Blog verwenden. Hinweise zur Verwendung von Storydiagrammen können Sie sowohl dem Screencast zur Vorlesung als auch dem Screencast zur Übung entnehmen.

Aufgabe 1: JUnit Tests (7P)

Erstellen Sie folgende Storyboards:

- Spielinitialisierung: Der Test soll auf einem Mancala Objekt die Methode `initGame („Alice“, „Bob“)` aufrufen und auf eine korrekte Endsituation prüfen. Die genaue Endsituation können Sie aus HA 5 Aufgabe 1 entnehmen (3P).
- 6 Storyboards zur Methode `Pit::moveStones()` (je 0.5P). Sie können hier ihre Lösungen aus HA 6 oder die Storyboards aus dem Mancala Projekt vom PM Blog verwenden.
- 2 Storyboards zur Methode `Mancala::checkEnd()` (0.5P). Sie können hier ihre Lösungen aus HA 6 oder die Storyboards aus dem Mancala Projekt vom PM Blog verwenden.

Abzugeben sind also 9 Storyboards inkl. generiertem Code.

Aufgabe 2: Mancala::initGame(...) (6P)

Erstellen Sie für die Methode `Mancala::initGame(...)` ein Storydiagramm in Fujaba, welches analog zu HA 5 Aufgabe 1 das Spiel initialisiert. Abgegeben werden muss das Fujaba Mancala Projekt inkl. Storydiagramm und generiertem Code in der Klasse Mancala.

Aufgabe 3: Mancala::checkEnd() (4P)

Erstellen Sie für die Methode `Mancala::checkEnd()` ein Storydiagramm in Fujaba, welches analog zu HA 6 auf das Spielende prüft. Abgegeben werden muss das Fujaba Mancala Projekt inkl. Storydiagramm und generiertem Code in der Klasse Mancala.

Aufgabe 4: Pit::moveStones() (10P)

Erstellen Sie für die Methode `Pit::moveStones()` ein Storydiagramm in Fujaba, welches analog zur Implementierung in HA 6 die Steine eines Pits unter Berücksichtigung der Regeln verschiebt. Unsere offiziellen Mancala Regeln sind im PM Blog zu finden. Abweichungen von den Regeln führen zu Punktabzug. Abgegeben werden muss das Fujaba Mancala Projekt inkl. Storydiagramm und generiertem Code in der Klasse Pit.