



Aufgabe 0.6

Geben Sie das Lösungswort des Quiz aus der Lernnotiz an.

Aufgabe 7

Ein neuer, leider nur in C geschulter Entwickler, hat die folgende Aufgabe in Java zu lösen:

Ziel ist ein vereinfachtes Mau-Mau-Programm für zwei Spieler, einem menschlichen und einem Computer-Spieler, zu entwickeln. Dabei können Spielkarten auf andere Spielkarten abgelegt werden, wenn sie entweder die gleiche Farbe (Karo, Herz, Kreuz, Pik) oder den gleichen Wert (7, 8, 9, 10, Bube, Dame, König, As) haben. Es wird mit den aus der Kombination von Farbe und Wert resultierenden 32 Karten gespielt. Weitere Regeln sollen nicht berücksichtigt werden. Am Anfang des Spiels erhält jeder Spieler fünf Karten auf die Hand, eine Karte wird auf den Tisch gelegt und der menschliche Spieler fängt an. Falls er nicht ablegen kann, wird automatisch eine Karte gezogen, ansonsten ihm angeboten eine Karte zu ziehen oder eine seiner passenden Karten abzulegen. Der Computerspieler legt immer ab, wenn er eine passende Karte hat, ansonsten zieht er automatisch. Sollten beim Ziehen keine Karten mehr zur Verfügung stehen, endet das Spiel unentschieden, ansonsten hat der Spieler gewonnen der zuerst keine Karten mehr auf der Hand hat.

Im Projekt `sqmAufgabeUeberdeckung` finden Sie eine Lösung. Ihre Aufgabe besteht darin die Qualitätssicherung für diese Lösung zu übernehmen.

- Lesen Sie sich in das Programm ein und erklären Sie informell, wie das Verteilen und Ziehen von Karten im Programm realisiert wird. Erklären Sie weiterhin informell, wozu die Variable `ableg` genutzt wird.
- Überlegen und dokumentieren Sie, welche verschiedenen Spielergebnisse am Ende genau herauskommen können.
- Die Klasse `Spiel` selbst darf von Ihnen nicht verändert werden. Die Klasse `EinAus` können Sie beliebig verändern. Schreiben Sie Tests für die Software, so dass die Coverage-Messung von Eclipse für die Methode `main` 100%-Überdeckung anzeigt. Schreiben Sie dazu zunächst eine kurze Übersicht, wie Sie vorgehen wollen. Überprüfen Sie mit Assertions, dass das gewünschte Endergebnis ausgegeben wird.

▼ Spiel.java	99,5 %
▼ Spiel	99,5 %
main(String[])	100,0 %

Hinweis: Natürlich können Sie sich für die Entwicklung zunächst kreative Ansätze zur Testerstellung überlegen, die auch Änderungen in der Klasse `Spiel` temporär enthalten. Wichtig ist nur, dass ihre endgültigen Tests die vorgegebene Klasse prüfen.

Erinnerung: Sie erinnern sich daran, dass Überdeckungen nur ein wichtiger Teilschritt und kein Ziel der QS sind.