

Persönliche Angaben

Vorname:		Nachname:	
Studiengang:		Matrikelnummer:	
Übung	<input type="checkbox"/> A: Montag, 14:45 Uhr <input type="checkbox"/> B: Dienstag, 13:15 Uhr	<input type="checkbox"/> C: Dienstag, 16:00 Uhr <input type="checkbox"/> D: Mittwoch, 11:15 Uhr	

Übungsmaterial erhalten Sie über Stud.IP. Um Klausur-Bonuspunkte erhalten zu können, müssen Sie sich in unserem Webanmeldesystem unter

<https://anmeldung.se.uni-hannover.de>

angemeldet haben. Dort können Sie mit Ihren Login-Daten jederzeit Ihre aktuelle Punktzahl einsehen.

Die Übungspunkte können nur eingetragen werden, wenn Sie in dem Anmeldesystem eingetragen sind. Übungspunkte können *nicht* nachgetragen werden. Bei technischen Problemen wenden Sie sich bitte per eMail an swq@se.uni-hannover.de.

Bitte beachten Sie folgende Hinweise:

- Heften (umgangssprachl. „tackern“) Sie Ihre Abgaben und verwenden Sie normales Papier (kein Schmierpapier, kein Werbepapier).
- Schreiben Sie leserlich oder weichen Sie gegebenenfalls auf Computerausdrucke aus. Was die Tutoren nicht entziffern können, wird nicht bewertet.
- Bitte beantworten Sie die Fragen in kurzen und verständlichen Sätzen.
- Verwenden Sie den obigen Vordruck für „Persönliche Angaben“ oder notieren Sie auf dem ersten Blatt Ihrer Abgabe oben rechts Ihren Namen und darunter Ihre Matrikelnummer und Übungsgruppe.
- Bitte kreuzen Sie oben Ihre Übungsgruppe an. Die korrigierten Abgaben werden in der jeweiligen Übungsgruppe eine Woche nach Abgabe zurückgegeben.
- Quelltext ist bitte ausgedruckt und nicht handschriftlich abzugeben. Halten Sie sich bei der Formatierung an die Grundsätze, welche Sie in SWT kennengelernt haben. Geltend ist die Formatierung auf dem Ausdruck.
- Reger Austausch über die Inhalte der Vorlesung in Arbeitsgruppen wird ausdrücklich empfohlen. Dennoch soll sichergestellt werden, dass jeder Studierende die Vorlesungsinhalte auch verstanden hat. Gruppenabgaben sind nicht erlaubt.

Aufgabe 1 (7 Punkte)

Führen Sie die GQM-Methode für folgendes Teilziel durch:

G1: Verringere die Anzahl von Bugs, die nach dem Release von Nutzern gefunden werden.

Erstellen Sie dazu ein Abstraction Sheet und leiten Sie daraus jeweils zwei Fragen und zwei Metriken ab.

Aufgabe 2 (3 Punkte)

- Nach welcher Formel können Sie die Cyclomatic Complexity nach McCabe direkt aus dem Code ablesen?
- Wenden Sie die Formel auf die Funktion `decimalToRoman(int n)` auf Seite 3 an.
- Was können Sie tun, um die Komplexität zu reduzieren?

Zu Aufgabe 2:

```
public class ConverterFor50s {
    /**
     * Konvertiert eine Dezimalzahl in eine römische Zahl. Es werden nur
     * Tausender- bis Fünfhunderteranteile konvertiert.
     */
    public static String decimalToRoman(int n) {
        String romanNumber = "";
        if (n <= 0) {
            romanNumber = "Keine römische Zahl";
        } else {
            int thousands = n / 1000;
            n %= 1000;
            for (int i = 0; i < thousands; i++) {
                romanNumber += "M";
            }
            if (n >= 900) {
                romanNumber += "CM";

                n -= 900;
                if (n >= 50) {
                    romanNumber += "L";
                }
            } else {
                if (n >= 500) {
                    romanNumber += "D";
                    int hundreds = (n - 500) / 100;
                    for (int i = 0; i < hundreds; i++) {
                        romanNumber += "C";
                    }
                    n %= 100;
                    romanNumber += (n >= 50) ? "L" : "";
                } else {
                    if (n >= 400) {
                        romanNumber += "CD";
                        n -= 400;
                        romanNumber += (n >= 50) ? "L" : "";
                    } else {
                        if (n >= 100) {
                            int hundreds = n / 100;

                            for (int i = 0; i < hundreds; i++) {
                                romanNumber += "C";
                            }
                            n %= 100;
                            romanNumber += (n >= 50) ? "L" : "";
                        } else {
                            romanNumber += (n >= 50) ? "L" : "";
                        }
                    }
                }
            }
        }
        return romanNumber;
    }
}
```