# Theorie

## Aufgabe 1 Kommunikation

**Beschreiben Sie das Kommunikationsmodell nach Friedemann Schulz von Thun.**



* Eine Sachinformation (worüber ich informiere) – **blau**
* Eine Selbstkundgabe (was ich von mir zu erkennen gebe) – **grün**,
* Einen Beziehungshinweis (was ich von dir halte und wie ich zu dir stehe) – **gelb**,
* Einen Appell (was ich bei dir erreichen möchte) – **rot**.

Siehe auch <http://www.schulz-von-thun.de/index.php?article_id=71>

## Aufgabe 2 Kommunikation

**Geben Sie ein Beispiel für das Senden inkongruenter Nachrichten.**

„Wie geht es dir?“ 🡪 „Gut“, aber Appell der Appell ist: „Lass mich in Ruhe“, Körpersprache passt nicht zur Aussage. Siehe auch <http://www.teachsam.de/psy/psy_kom/psy_thun/psy_kom_thun_5_2.htm>

## Aufgabe 3 Software Qualität

**Erklären Sie den Unterschied zwischen statischer und dynamischer Typisierung.**

Statische Typisierung: Typen sind festgelegt zur Compilezeit (z.B. bei Java, C#, …).

Dynamische Typisierung: Typen sind interpretiert bzw. variabel (z.B. php) und werden erst zur Laufzeit festgelegt.

## Aufgabe 4 Qualitätssicherung

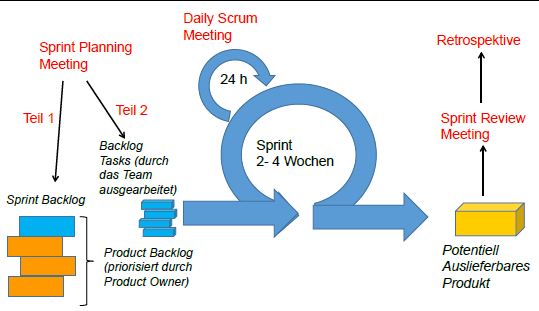
**Nennen Sie zwei typische Notationskonventionen für Programm Code und warum es jeweils sinnvoll ist, diese Konvention zu nutzen.**

* **Pascal Case:** Bezeichner, sowie jedes enthaltene Wort startet mit Großbuchstaben.
* **CamelCase:** Bezeichner startet mit Kleinbuchstaben, jedes weitere Wort groß.
* **Uppercase:** Komplett in Großbuchstaben
* **Lowercase:** Komplett in Kleinbuchstaben

Warum sinnvoll: Zur Unterscheidung zwischen Methoden, Variablen, Enums, …

## Aufgabe 5 Scrum

**Beschreiben Sie den Ablauf eines Scrum Sprints.**



Der Sprint als zentraler Container für die Arbeit:

Ein Sprint beinhaltet und umfasst das Sprint Planning, die Daily Scrums, die Entwicklungsarbeit, das Sprint Review und die Sprint Retrospektive.

Während des Sprints:

* werden keine Änderungen vorgenommen, die das Sprint-Ziel gefährden,
* wird der Qualitätsanspruch nicht geschmälert, und
* der Anforderungsumfang kann zwischen ProductOwner und Entwicklungsteam geklärt und neu ausgehandelt werden, wenn sich neue Erkenntnisse ergeben haben.

## Aufgabe 6 Scrum in großen Projekten

**In Scrum Projekten mit mehreren Scrum Teams müssen die Teams untereinander synchronisiert werden. Beschreiben Sie, wie diese Synchronisation erfolgen kann.**

* Scrum of Scrums
* ProductOwner-Team
* ScrumMaster Group
* Virtuelle Teams für spezifische Querschnittsaspekte (Architektur, Dokumentation, Tests, …)
* Gemeinsame Planning Meetings
* Evtl weitere Meetings

## Aufgabe 7 Requirements Engineering

**Nennen Sie drei unerwünschte Effekte, die bei natürlich sprachlich formulierten Anforderungen auftreten können, beschreiben Sie jeweils, was damit gemeint ist und geben Sie jeweils ein Beispiel.**

* Unspezifisch: „Das System soll Nachrichten senden“ 🡪 Wann? An wen?
* Fehlerträchtig: Natürliche Sprache ist nicht eindeutig 🡪 „Er will sie nicht.“ oder „Er will, sie nicht.“ 🡪 Daraus können logische Fehler auftreten.
* Schwierig zu prüfen: Ungenaue Anforderungen sind schwer zu prüfen 🡪 Wie oben: „Das System soll Nachrichten senden“ 🡪 Woher weiß man, ob die Nachricht ankommt?

<http://www.konjugation.de/posts/list/452_satzzeichen_koennen_leben_retten.page>

Offene Fragen (nicht spezifizierte Inhalte, die man übersieht), Verlust definierter Formalismen (Sprachbarriere, Formulierung etc. nicht übertragbar und eindeutig/ standardisiert), Zweideutigkeiten (aussagen mehrdeutig auslegbar, Kombination aus den ersten beiden)

## Aufgabe 8 Requirements Engineering

**In Ihrem Projekt verwenden Sie ein Tool zur Anforderungsverwaltung. Sie definieren Attribute, die für eine Anforderung aufzunehmen sind.**

**Nennen Sie die 4 aus Ihrer Sicht wichtigsten davon.**

Annahme: Es sind Attribute von Userstories gemeint

* Wer hat die Anforderung erstellt? (User / Gruppe / Rolle)
* Wann wurde sie erstellt?
* Was definiert die Anforderung? Was soll damit erreicht werden?
* Historisierung der Änderungen

## Aufgabe 9 Maven

**Erklären Sie das Prinzip der Remote und der lokalen Repositories von maven.**

**Warum ist es meistens notwendig, einen Repository Manager zu verwenden? Nennen Sie drei Gründe.**

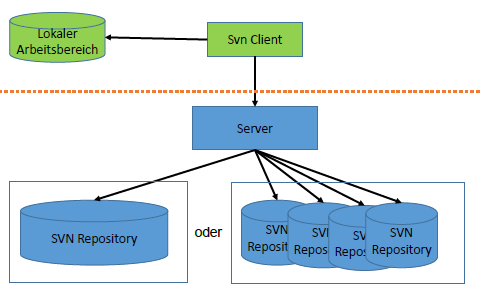
Maven sucht zunächst im lokalen Repository nach Plugins, ist das Plugin dort nicht vorhanden, so wird im zentralen Repository gesucht.

* Remote Repositories liegen außerhalb eigener Kontrolle
* Umgang mit lizensierter Software dabei? 🡪 Daten liegen ja nicht in der Firma
* Eigene Artefakte sollen nicht publiziert werden, aber innerhalb der Firma zugänglich sein

## Aufgabe 10 Subversion

Subversion kann in zwei Architekturvarianten installiert werden. Skizzieren Sie diese Varianten in einem einfachen Architekturbild und nennen Sie für jede Variante einen Vorteil gegenüber der anderen Variante.

Ein Repository pro Projekt oder ein Repository insgesamt



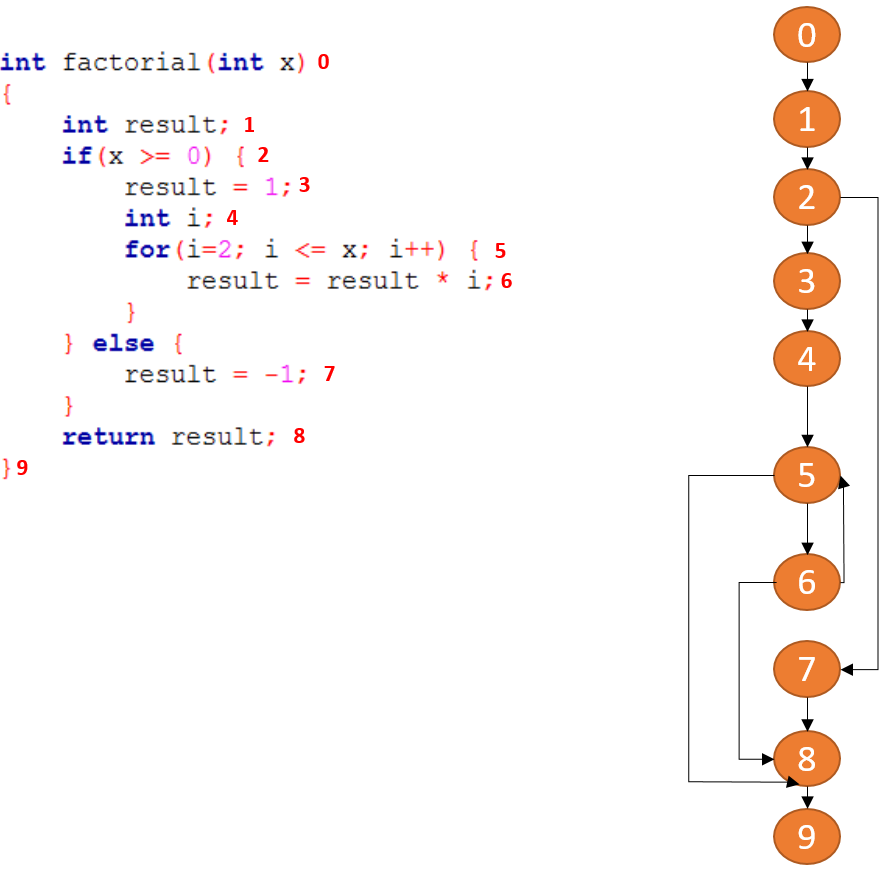
Vorteil 1 Repository pro Projekt: 🡪 Bessere logische Unterteilung der einzelnen Projekte

Vorteil 1 Repository insgesamt: 🡪 Nur ein Repository muss verwaltet werden

# Teil 2 Praxis

## Aufgabe 1 Testen

**1. Erstellen Sie einen Kontrollflussgraphen für folgende Funktion.**



**2. Definieren Sie Testfälle, so dass eine vollständige Anweisungsüberdeckung gegeben ist.**

„Testmenge wird so gewählt, dass alle Knoten des Kontrollflussgraphen mindestens einmal durchlaufen werden.“ 🡪 2 Testfälle:

* x < 0
* x >= 0 und 2 <= x, also x >= 2

**3. Definieren Sie Testfälle, so dass eine vollständige Zweigüberdeckung gegeben ist.**

„Jede Kante des Kontrollflussgraphen muss durch mindestens einen Testfall durchlaufen werden.“ 🡪 3 Testfälle:

* x < 0
* x => 0 und x < 2, also 0 <= x < 2
* x > 0 und x >= 2, also x >= 2

**4. Erklären Sie das Konzept der vollständigen Pfadüberdeckung.**

„Dann erfüllt, wenn für jeden möglichen Pfad von Eingangsknoten zu Ausgangsknoten ein separater Testfall existiert.“

**5. Wie viele Testfälle benötigen Sie für das Programm oben für eine vollständige Pfadüberdeckung?**

* 2 „normale“ (x < 0 und x > 0 und x < 2, also x = 1)
* 2x-1 für die Schleife

**6. Mit welcher Pfadüberdeckungsmethode können Sie diese Zahl einschränken (Name und Erklärung)? Definieren Sie Testfälle entsprechend dieser Methode.**

Boundary Interior Pfadüberdeckung:

Für jede Schleife drei Gruppen von Testfällen:

* Äußere Pfade: Schleifen werden nicht betreten.
* Grenzpfade: Genau eine Iteration.
* Innere Pfade: Mindestens eine weitere Iteration. Testfälle werden so gewählt, dass innerhalb der ersten beiden Iterationen alle möglichen Pfade abgearbeitet werden.

Testfälle:

* x < 0
* x => 0 und x < 2, also 0 <= x < 2
* x = 2
* x > 0 und x > 2, also x > 2

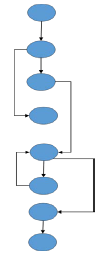
## Aufgabe 2 McCabe Metrik

**a. Wie lautet die Definition der McCabe Metrik?**

V(G) = e – n + 2 \* p

* e = Anzahl der Kanten
* n = Anzahl der Knoten
* p = unabhängige Teile des Graphs/ Zusammenhangskomponenten

**b. Berechnen Sie für folgenden Kontrollflussgraphen die McCabe Metrik**



V(G) = e – n + 2 \* p = 8 – 8 + 2 \* 1 = 2

## Aufgabe 3 Konfig Elemente

**Zu Beginn eines neuen Projekts definieren Sie, welche Elemente Sie als Konfigurationselemente verwalten wollen.**

**Ordnen Sie die Elemente der Tabelle durch ankreuzen einer der Kategorien (Konfigurations Element, kein Konfigurations Element, Mögliches Konfigurations Element) zu.**

|  |  |  |  |
| --- | --- | --- | --- |
| Element | Konfigurations Element | Mögliches Konfigurations Element | Kein Konfigurations Element |
| Quelltext | x |  |  |
| Projektpläne |  |  | x |
| Usecase Beschreibungen | x |  |  |
| Build Skripte | x |  |  |
| Externe Bibliotheken |  | x |  |
| Installationsanleitung |  |  |  |
| Liste offener Punkte |  |  | x |
| Entwicklungsumgebung |  | x |  |
| Meetingprotokolle |  |  | x |

## Aufgabe 4 Zielbaum

**Ihnen liegt folgende Beschreibung von Zielen für ein Ausleihsystem vor:**

**Das System muss eine komfortable Ausleihe von Leihobjekten gewährleisten. Dafür ist sowohl eine komfortable Erfassung der Entleiherdaten, als auch eine komfortable Erfassung der Leihobjektdaten, als auch eine komfortable Erfassung der Leihdauer nötig. Die komfortable Erfassung der Leihobjektdaten kann entweder durch eine Unterstützung bei der manuellen Erfassung erfolgen oder durch eine automatische Erfassung.**

**Stellen Sie diese Information in einem Zielbaum dar.**

* **Haben wir nicht durchgenommen im SoSe 2017**

## Aufgabe 5 Testen – Äquivalenzklassen

**Ein Programm zur Lagerverwaltung einer Baustoffhandlung besitzt eine Eingabemöglichkeit für die Registrierung von Anlieferungen.**

**Die Parameter der Funktion sind wie folgt spezifiziert:**

**Werden Holzbretter angeliefert, so wird die Holzart (Eiche, Buche oder Kiefer) eingegeben. Ferner wird die Länge in cm angegeben, die stets zwischen 100 und 500 cm liegt. Als Anzahl kann ein Wert zwischen 1 und 9999 angegeben werden. Weiterer Eingabeparameter ist eine Auftragsnummer, die bei Holzlieferungen mit dem Buchstaben H beginnt.**

**Aufgabe:**

1. **Definieren Sie für die einzelnen Eingabe Parameter Äquivalenzklassen für Tests. Berücksichtigen Sie dabei gültige und ungültige Eingaben.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Holz | Ja | Nein |  |  |
| Holzart | Eiche | Buche | Kiefer | Andere |
| Länge | ]-∞, 100] | ]100, 500[ | [500, ∞[ |  |
| Anzahl | ]-∞, 1] | ]1, 9999[ | [9999, ∞[ |  |
| Auftragsnummer | Beginnt mit H | Beginnt nicht mit H |  |  |

1. **Nach welchem Muster definieren Sie mit den Ergebnissen aus Teilaufgabe a die Äquivalenzklassen Tests?**

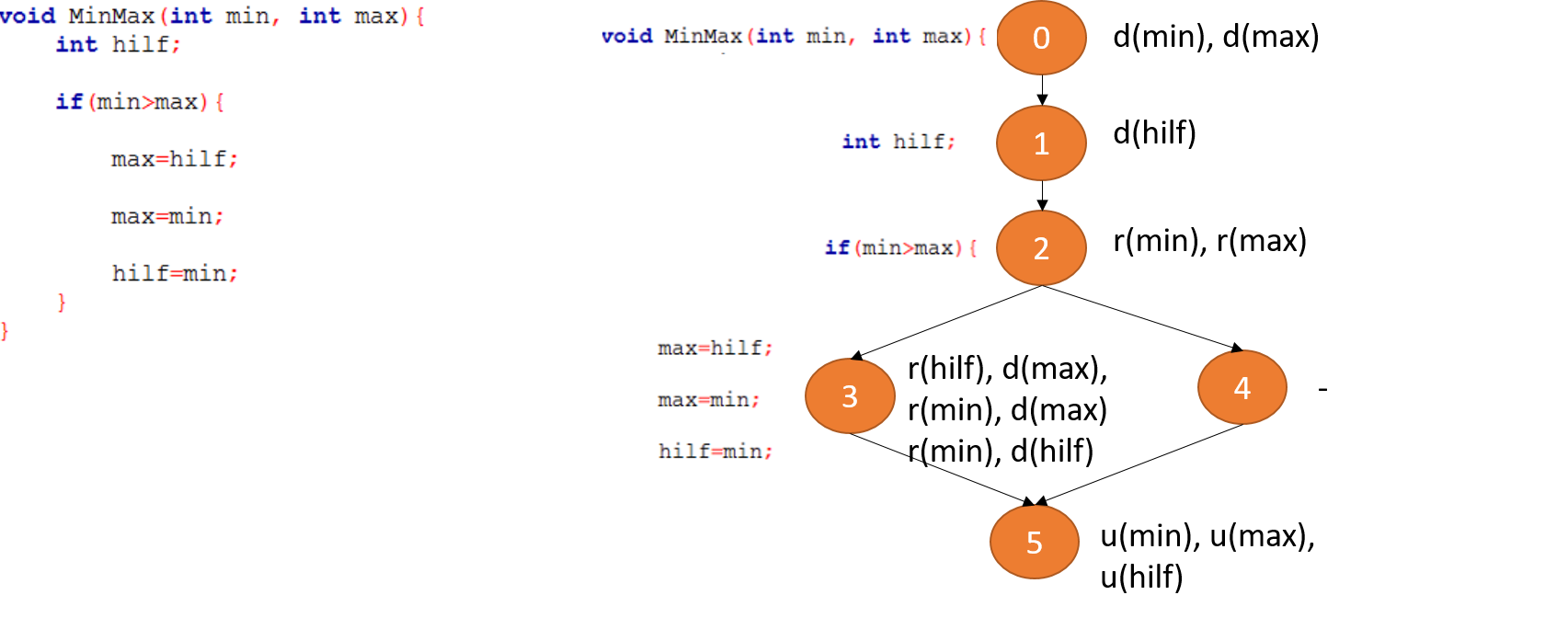
Für jede Äquivalenzklasse eines Parameters einen Vertreter definieren und dann alle möglichen Kombinationen von Vertretern der möglichen Äquivalenzklassen der einzelnen Parameter als Input für jeweils einen Testfall verwenden.

* Insgesamt: 2 \* 4 \* 3 \* 3 \* 2 = 144 Testfälle

Aufgabe 6 Anomalienanalyse

**Erstellen Sie eine Datenflussanalyse des folgenden Programms und markieren Sie gegebenenfalls Anomalien.**

****

****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 |
| min | d |  | r | rr |  | u |
| max | d |  | r | dd |  | u |
| hilf |  | d |  | rd |  | u |

Anomalie: dd, weil max zweimal geschrieben wird.

