

**Thema, Ziele:** Qt Vertiefung und PM in SwEng

### Aufgabe 1: Qt Counter: zweiter Zähler

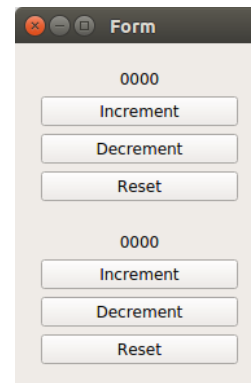
Erweitern Sie Ihr CounterWidget Programm von letzter Woche so, dass zusätzlich noch ein zweiter Zählerstand angezeigt wird und verändert werden kann – siehe nebenstehende Abbildung.

Die beiden Zähler sollen unabhängig voneinander funktionieren.

Erstellen Sie dazu ihr eigenes Widget mit dem Namen *CounterWidget*, welches die benötigten Widgets für einen Counter beinhaltet. Instanzieren Sie entsprechend zwei Objekte der *CounterWidget* Klasse um diese Aufgabe zu lösen.

Hinweis:

Ihr eigene *CounterWidget* Klasse soll von *QWidget* ableiten.



### Aufgabe 2: Projektmanagement und Software Engineering

Zitat (Autor unbekannt): *"Aus Fehlern wird man klug."*

Ergänzung: *"... sofern die nötige Grundklugheit vorhanden ist."*

Obwohl in der Raumfahrt bei der Software-Entwicklung konsequent Software-Engineering-Methoden eingesetzt werden, kann es auch hier zu spektakulären Unfällen kommen:

#### 2.1 "The Explosion of the Ariane 5" – 1996

Einen solchen Unfall beschreibt der Artikel *"The Explosion of the Ariane 5"* auf der nächsten Seite.

Quelle: "<http://www.ima.umn.edu/~arnold/disasters/ariane.html>".

Lesen Sie diesen Artikel und beantworten Sie anschliessend die untenstehenden Fragen.

Beachten Sie, dass aus diesem Artikel nicht hervorgeht, dass das bei der Ariane 5 für das INS ("Inertiales Navigationssystem") verwendete Softwaremodul weitgehend von der Ariane 4 übernommen wurde. Dort lief diese Software lange Zeit erfolgreich und wurde deshalb auch als sicher eingestuft.

(Details siehe "[http://de.wikipedia.org/wiki/Ariane\\_V88](http://de.wikipedia.org/wiki/Ariane_V88)")

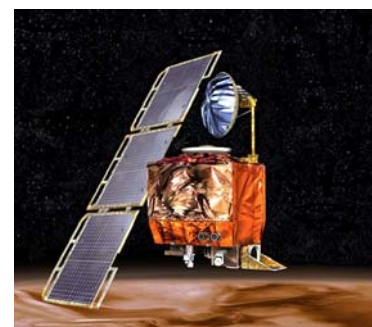
#### Fragen:

1. Was war der Fehler der zum Unfall mit der Ariane 5 führte?
2. Was war Ihrer Meinung nach die Ursache für diesen Fehler?
3. Wie hätte sich dieser Fehler Ihrer Meinung nach vermeiden lassen können?
4. Welche Schlüsse ziehen Sie aus diesem Fehler in Bezug auf die Entwicklung von Software im Allgemeinen und das Software Engineering im speziellen (Entwicklung, Testen, Fehler, Wartung, etc.)?
5. Was hat dieser Fehler gekostet? (nur direkte Kosten, ohne Entwicklungs- und Folgekosten)

#### 2.2 Verlust der NASA-Sonde "Mars Climate Orbiter" (MCO) – 1999

Lesen Sie den folgenden Wikipedia-Beitrag "[http://de.wikipedia.org/wiki/Mars\\_Climate\\_Orbiter](http://de.wikipedia.org/wiki/Mars_Climate_Orbiter)" und beantworten Sie anschliessend folgende Fragen.

1. Wie lange dauerte der Flug dieser NASA-Sonde bis der Fehler sich bemerkbar machte?
2. Was war der Fehler der zum Verlust führte?
3. Angenommen Sie würden die Fehlerursache nicht kennen. Hätten Sie sich dann vorstellen können, dass in der Raumfahrt solch "dumme" Fehler vorkommen können?
4. Was hat dieser Fehler gekostet?



### Aufgabe 3: Management von SwPrj Verständnisfragen

Erarbeiten Sie die Aufgaben in einer 2er oder 3er Gruppe. Die Lösungen können fortlaufend mit dem Praktikumsleiter besprochen werden.

- a) Zählen Sie drei völlig verschiedene Vorhaben auf, die eindeutig Projekte sind.
- b) Zählen Sie drei völlig verschiedene Vorhaben auf, die sicher keine Projekte sind. Geben Sie dabei an, warum sie keine Projekte sind.
- c) In einer Firma muss jedes Jahr im Dezember ein Neujahrsgeschenk für die Kunden "erfunden" werden. Da Sie als kreativer Kopf gelten, landet diese Aufgabe meistens bei Ihnen.  
Frage: Handelt es sich bei diesem Vorhaben um ein Projekt oder nicht?  
Führen Sie die Argumente auf, die für ein Projekt sprechen und diejenigen Argumente die dagegen sprechen.
- d) "Ein Mann braucht zum Bau einer 2 m langen Brücke 1.5 Tage. Wie lange brauchen 100 Leute für den Bau einer 2 km langen Brücke?"  
Berechnen Sie die Dauer mit Hilfe eines Dreisatzes.  
Erklären Sie warum diese "Milchmädchenrechnung" hier falsch ist.
- e) Zeichnen Sie eine zweidimensionale Grafik, die zeigt, wie während dem Projektverlauf sich bei einem typischen Projekt, das Wissen (in Prozent) über das zu realisierende Projekt darstellt. (x-Koordinate = Zeit, y-Koordinate = Wissen in Prozent).
- f) Stellen Sie in einer zweidimensionalen Grafik dar, wie sich bei einem typischen Projekt, während dem Projektverlauf der Handlungsspielraum des Projektleiters ändert (x-Koordinate = Zeit, y-Koordinate = Handlungsspielraum).
- g) Eine berühmte Aussage bezüglich Software-Projekten ist:  
*"adding manpower to a late software project makes it later"* (Fred Brooks, 1975)  
Erklären Sie diese Aussage, Brooks's law genannt, und begründen Sie, warum sie auch heute, 35 Jahre später für die Softwareentwicklung immer noch richtig ist.  
Erklären Sie allgemein bei welcher Art von Projekten diese Regel nicht gilt und geben Sie ein konkretes Beispiel für ein solches Projekt an.
- h) Gegenfrage zu Aufgabe g): Sollten demnach alle Mitarbeiter bereits von Projekt-Beginn an dabei sein?

### Aufgabe 4: Projektphasen Verständnisfragen

- a) Was ist das Ziel in der Projektentstehungsphase? Formulieren Sie Ihre Antwort so, dass diese aus maximal 3 Wörtern besteht. – Hinweis: Worin besteht die Hauptarbeit in dieser Phase?
- b) Was sind die Hauptaufgaben der jeweiligen Projektphasen? Geben Sie pro Phase mindestens eine Aufgabe an.