

Name: _____

Matrikelnummer: _____

Studiengang: _____

Unterschrift: _____

Bitte beachten:

DAS SCHREIBEN AUF DEM KLAUSUREXEMPLAR VOR DEM STARTSIGNAL UND AUCH DAS SCHREIBEN AUF DEM KLAUSUREXEMPLAR NACH DEM ENDESIGNAL FÜHRT OHNE WEITERE WARNUNG SOFORT ZUR UNGÜLTIGKEIT DER KLAUSUR. DAS GILT AUCH FÜR DAS SCHREIBEN VON NAMEN UND MATRIKELNUMMER NACH DEM ENDESIGNAL!

1. **Klausur nicht öffnen bis zur Erlaubniserteilung durch die Klausuraufsicht!**
2. **Bitte Personalausweis als auch Studentenausweis auf den Tisch legen. Prüfen Sie, ob Ihre Matrikelnummer auf der ausgehändigten Liste aufgeführt ist. Wenn nicht, so**
 - (a) Füllen Sie einen Vorbehalt (Proviso) aus, der Ihnen von der Klausuraufsicht ausgehändigt wird. Sie dürfen die Klausur erst beginnen, nachdem Sie den Vorbehalt ausgefüllt und abgegeben haben.
 - (b) **Füllen Sie auch den “Formzettel zur Nachmeldung” aus.** Lassen Sie das Formular vom Studiensekretariat abstempeln und unterzeichnen und bringen Sie diesen Zettel *persönlich* in das STS-Sekretariat (Harburger Schloßstr. 20, 2. Stock, Frau Hantschmann); bringen Sie auch ihren Personalausweis mit.
3. Bearbeitungszeit: **90 Minuten. Zusätzliche Hilfsmittel sind nicht erlaubt.** Das Symbol “⌚” gibt Hinweise zur Bearbeitungszeit.
4. Es befindet sich ausreichend Platz für Ihre Antworten auf den Aufgabenzetteln. Wenn Sie *zusätzliches Papier* von der Klausuraufsicht erhalten, schreiben Sie auch Ihren Namen, Matrikelnummer sowie eine Seitennummer auf die Zettel. Wenn Sie den Prüfungsraum verlassen müssen, **so melden Sie sich geräuschlos und informieren Sie die Klausureinsicht. Verlassen Sie Ihren Platz nicht unaufgefordert.**

Mit dem Öffnen der Klausur bis zur Aufforderung warten

1. COCOMO

Die Berechnung des Aufwandes erfolgt nach der Formel

$$A = C * KLOC^B$$

mit A = Entwicklungsaufwand in MM, und B, C Konstanten gemäß Tabelle

Projekt	C	B
Einfach	2,4	1,05
Mittel	3,0	1,12
Komplex	3,6	1,20

Die laut Schätzmethode optimale Entwicklungszeit T soll bei gegebenem Entwicklungsaufwand A in MM vereinbarungsgemäß wie folgt bestimmt werden.

$$T = D * A^E$$

mit

T: Entwicklungszeit

D,E : Konstanten gemäß Tabelle

Projekt	D	E
Einfach	2,5	0,38
Mittel	2,5	0,35
Komplex	2,5	0,32

Berechnen Sie nach COCOMO (a) den Aufwand, (b) die Entwicklungszeit, (c) die durchschnittliche Zahl von Programmierern, (d) die Produktivität des gesamten Teams pro Arbeitstag (1 MM = 20 Arbeitstage) für ein einfaches Projekt mit geschätzten Umfang von 39800 lines of code.

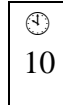
x	$x^{0,32}$	$x^{0,35}$	$x^{0,38}$
10	2,09	2,24	2,40
20	2,61	2,85	3,12
30	2,97	3,29	3,64
40	3,26	3,64	4,06
50	3,50	3,93	4,42
60	3,71	4,19	4,74
70	3,89	4,42	5,03
80	4,06	4,64	5,29
90	4,22	4,83	5,53
100	4,37	5,01	5,75
110	4,50	5,18	5,97
120	4,63	5,34	6,17
130	4,75	5,49	6,36
140	4,86	5,64	6,54
150	4,97	5,78	6,71
160	5,07	5,91	6,88
170	5,17	6,03	7,04
180	5,27	6,16	7,19
190	5,36	6,27	7,34
200	5,45	6,39	7,49

$$\begin{aligned} 39,8^{1,05} &= 47,84 \\ 39,8^{1,12} &= 61,92 \\ 39,8^{1,20} &= 83,14 \end{aligned}$$

10

2. Function-Point Methode

Für ein Software-System wurde mit Hilfe der Angaben im Lastenheft eine Aufwandsabschätzung nach der Function-Point-Methode durchgeführt. Finden und korrigieren Sie die in dieser Berechnung enthaltenen Fehler!



„Das System umfasst 8 komplexe Funktionen,
16 komplexe Daten und 21 komplexe Ausgaben“

Funktionen	8	einfach * 7	56
		mittel * 10	
		komplex * 15	
Ausgaben	21	einfach * 4	105
		mittel * 5	
		komplex * 7	
Daten	10	einfach * 5	160
		mittel * 7	
		komplex * 10	

Summe unbewertete Function-Points: 321

„Einflussfaktoren: An die Zuverlässigkeit und Benutzbarkeit des Systems werden extrem hohe Anforderungen gestellt. Besondere Produktleistungen liegen nicht vor, weshalb dieser Einflussfaktor auf 0 gesetzt wird. Alle anderen Punkte werden auf neutrale Werte gesetzt (insgesamt 24 Punkte). Es ergeben sich 25,5 Einflusspunkte, was einem Faktor von 0,95 entspricht.“

Einflussfaktoren	(0-6 Punkte)
Produktleistungen	0
Qualitätsanforderungen	1
Benutzeroberfläche	(3)
Nichtfunktionale Anforderungen	(3)
Schnittstellen	(3)
Algorithmische Komplexität	(3)
Architektur	(3)
Werkzeugeinsatz (umgekehrt proportional)	7
Erfahrung	(3)
Reife des Entwicklungsprozesses (umgekehrt proportional)	(3)

Summe Einflusspunkte: 25
 Faktor: 0,95
 Bewertete FPs: $321 * 0,95 = 304,95$

3. Vergleich von Metriken

Vergleichen Sie die folgende Metriken, indem Sie ihre Vorteile und Nachteile schildern. Erläutern Sie auch, welchen Zweck die Metriken haben.

- LOC
- Halstead
- McCabe

Halstead	MCCabe
<p>Umfang $V = (N1+N2) * \lg(n1+n2)$ mit</p> <ul style="list-style-type: none"> - $n1, n2$ Anzahl der unterschiedlichen Operatoren, Operanden - $N1, N2$ Gesamtzahl verwendeter Operatoren, Operanden - Operator kennzeichnet Aktionen (+, *, While, For, ...) - Operand kennzeichnet Daten (Variablen, Konstanten, ...) 	<p>$V(g) = e - n + 2p$</p> <p>e ... Anzahl der Kanten n ... Anzahl der Knoten p ... Anz. verbundener Komponenten</p>

Name:

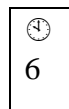
ID:

4. Tests Überdeckung

Gegeben sei folgendes Quellcodestück:

```
result := 0
if (x > y) or (x = 0)
    then result := x
    else result := y
endif
```

Geben Sie Testdaten an, sodass Sie (a) Anweisungsüberdeckung, (b) Zweigüberdeckung und (c) Bedingungsüberdeckung erreichen und begründen Sie jeweils Ihre Wahl.



5. Semantik der Vor- und Nachbedingungen

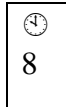
Gegeben sei folgendes Programmfragment:

$S_1; S; S_2$

wobei S ein Statement mit $\{V\}$ als Vorbedingung und $\{P\}$ als Nachbedingung ist. Wir nehmen an, das obige Programmfragment wird abgearbeitet.

Was ist richtig? (Mehrfachankreuzung möglich)

- ☐ S wird ausgeführt, wenn V gilt.
- ☐ Vor der Ausführung von S muss V gelten.
- ☐ Nach der Ausführung von S gilt P .
- ☐ Nach der Ausführung von S gilt P , wenn vorher V gilt.
- ☐ Wenn V vor der Ausführung von S gilt und S ausgeführt wird, gilt nach der Ausführung von S die Bedingung P , sonst gilt P nicht.
- ☐ Wenn V vor der Ausführung von S gilt und S ausgeführt wird, gilt nach der Ausführung von S die Bedingung P .

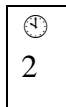


Richtige Antworten: je +1 Punkt
 Falsche Antworten: je -1 Punkt
 Keine Antwort: 0 Punkte
 Minimale Punktzahl insgesamt: 0 Punkte

6. Spezifikation und Implementierung

a) Gegeben sei die folgende Spezifikation:

```
factorial(n) = n * factorial(n-1) , n > 0
              1                , n = 0
```



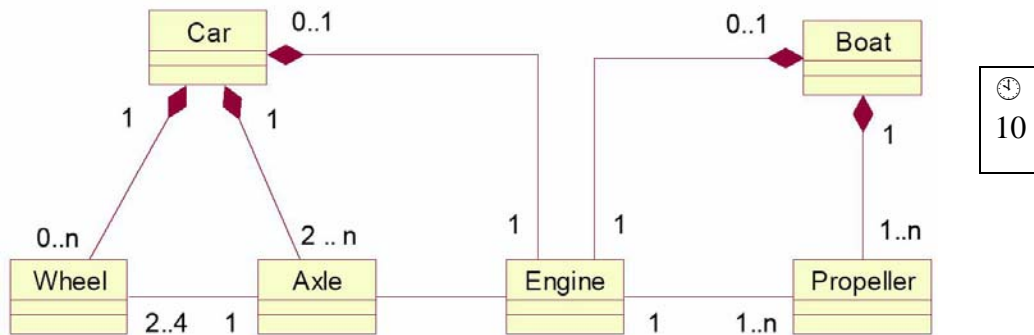
Gegeben sei auch die folgende Implementierung,

```
int res = 1;
int i = 1;
while ( i <= n ) {
    res = res * i ;
    i = i + 1;
}
```

Wie nennt man die Überprüfung der Implementierung gegenüber die Spezifikation?

b) Gegeben sei eine Spezifikation für die Operation von Ampeln an einer Straßenkreuzung. Wie nennt man die Überprüfung, ob diese Spezifikation vollständig and korrekt ist?

7. UML Klassenmodell



- a) Erlaubt das obige UML Klassenmodell, dass ein bestimmter Motor (Engine) gleichzeitig bei einem Auto (Car) *und* bei einem Boot (Boat) eingebaut ist? Warum oder warum nicht?
- b) Kann ein bestimmter Motor gleichzeitig in zwei verschiedenen Autos eingebaut sein? Begründen Sie.

Name:

ID:

- c) Erklären Sie, welche Änderungen im Diagramm vorzunehmen sind, so dass es Auto-spezifische Motore und Boot-spezifische Motore gibt. Zeichnen Sie ggf. ein neues Diagramm.

- d) Wie würden sie die Multiplizitäten zwischen Wheel und Car mittels OCL ausdrücken? Benennen Sie ggf. in der obigen Darstellung die Assoziationsrollen, die Sie benötigen.

8. UML und OCL Modellierung

Zeichnen Sie die UML Diagramme, die Sie für notwendig halten, um die folgende Beschreibung eines Roboter-Kasten-Szenarios möglichst vollständig zu modellieren. Setzen Sie ggf. OCL-Ausdrücke (Invarianten und Preconditions) ein. Sie erhalten die Maximalpunktzahl, wenn Sie zwei von drei der untenstehenden Bedingungen formalisieren.

- 1) Es gibt zwei Arten von Robotern, solche die Kästen tragen können, und solche die malen können. Aber es gibt auch Roboter, die sowohl die Fähigkeit zum Malen als auch zum Tragen besitzen, wenn auch nicht gleichzeitig.
- 2) Alle Roboter, die einen Kasten tragen können, sind auch in der Lage, Kästen zu stapeln. Ein Roboter kann einen Kasten anmalen, wenn dieser nicht gerade getragen wird (wo auch immer von welchem Roboter) und wenn keine weiteren Kästen darüber liegen.
- 3) Am Anfang sind alle Kästen weiß. Jeder Maler-Roboter hat nur eine Farbe, er kann malen, solange sein Vorrat an Farbe reicht, aber nicht wenn derselbe Kasten benutzt wird (d.h. gemalt oder getragen wird).

