

Aufgabe 1 : Aperitivi (12 Punkte)

a) Was versteht man unter Software Maintenance and zwischen welchen Arten unterscheidet man? (2 Punkte)

b) Was versteht man unter Refactoring?

Was sind die Vorteile von Refactoring?

Nennen sie zwei Refactorings.

Ist Refactoring eine Methodik des Reverse Engineerings, des Re-Engineerings oder des Restructurings? (4 Punkte)

c) Was versteht man unter den *Laws of Software Evolution*?

Beschreiben Sie das *Law of Increasing Entropy* und Mechanismen, um dessen Effekte zu minimieren.

Nennen Sie zwei weitere Laws.

(6 Punkte)

Aufgabe 2 : Antipasti misti (20 Punkte)

Beantworten Sie die folgenden Fragen:

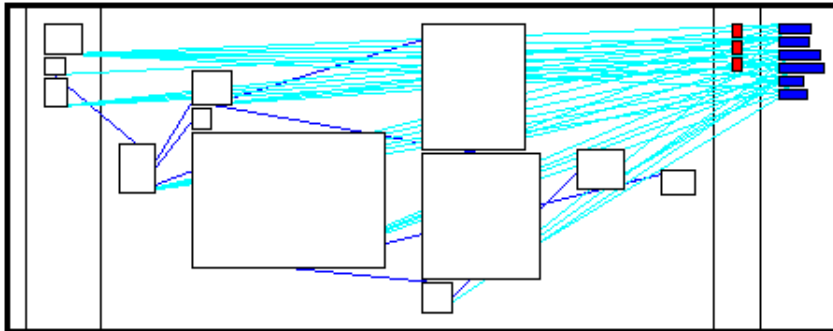
- a) Welche pre- und postconditions gibt es bei einem 'Rename Method'-Refactoring zu beachten? (2 Punkte)
- b) Warum ist LOC (Lines of Code) keine gute Metrik um die Komplexität einer Klasse auszudrücken?
Geben Sie 2 Metriken an, die besser dafür geeignet sind. (3 Punkte)
- c) Weshalb ist hohes Coupling zwischen Klassen/Komponenten schlecht für die Wartung? (2 Punkte)
- d) Ordnen Sie folgenden Wartungsaktivitäten den entsprechenden Aufwand (5%, 10%, 25%, 50%) zu:
(1) source code comprehension
(2) change planning
(3) change implementation
(4) change documentation
(5) change testing (2 Punkte)
- e) Was ist der Unterschied zwischen einem Software Configuration Management und einem Defect Tracking Tool? (3 Punkte)

- f) Wozu benötigt man im Reengineering ein Domänenmodell bzw. wozu kann man es besonders gut einsetzen? (3 Punkte)
- g) Warum muss man, bevor man zwei oder mehr Systeme und deren Evolution miteinander vergleicht, diese zuerst normalisieren? Was genau sollte normalisiert werden? (3 Punkte)
- h) Was bringt die Analyse von Bug Reports für die Software Evolution? (3 Punkte)

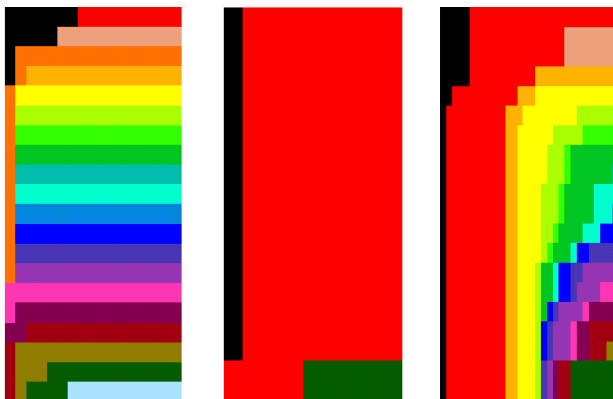
Aufgabe 3 : Primo e Secondo con carne / Piatto vegetariano (30 Punkte)

a) Interpretieren Sie den folgenden 'Class Blueprint'.

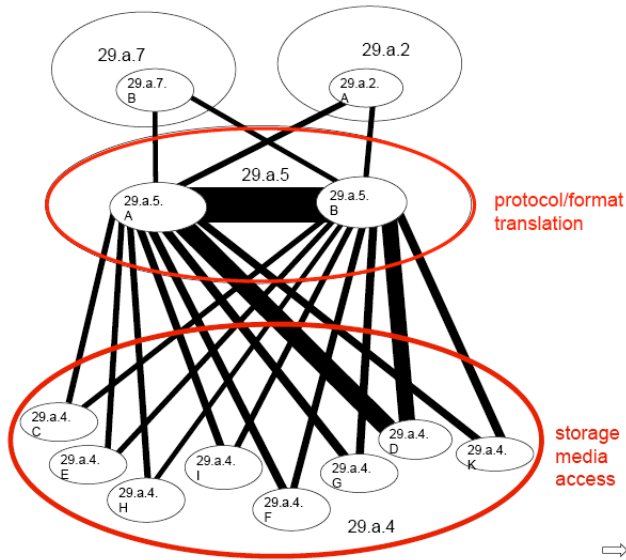
(6 Punkte)



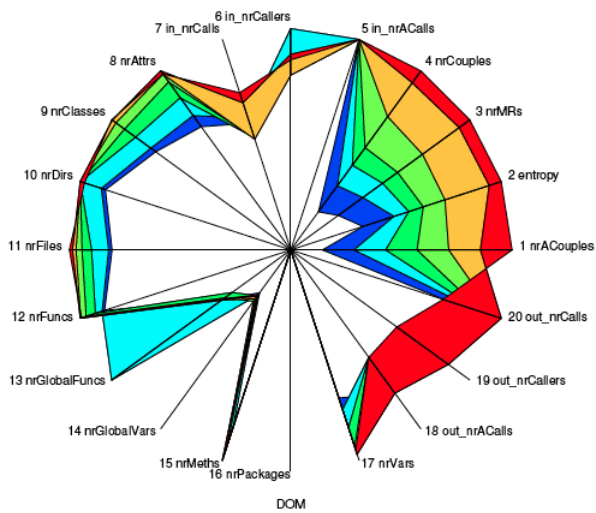
b) Interpretieren Sie die drei Module, deren Evolutionsgeschichte nachfolgend graphisch dargestellt ist hinsichtlich Ähnlichkeiten, gemeinsame Änderungen und Wachstum. (6 Punkte)



c) Interpretieren Sie die nachfolgende Evolutionsgeschichte des Package 'protocol/format translation'. (6 Punkte)



d) Interpretieren Sie folgendes Kiviati-Diagramm (Radar-Chart) für das DOM Modul aus Mozilla. (6 Punkte)



e) Interpretieren Sie die Evolutionsgeschichte der Klassen A bis E.

(6 Punkte)

		ENOM	LENOM	EENOM					
A	<table border="1"><tr><td>2</td><td>4</td><td>3</td><td>5</td><td>7</td></tr></table>	2	4	3	5	7	7	3.37	3.25
2	4	3	5	7					
B	<table border="1"><tr><td>2</td><td>2</td><td>3</td><td>4</td><td>9</td></tr></table>	2	2	3	4	9	7	5.75	1.37
2	2	3	4	9					
C	<table border="1"><tr><td>2</td><td>2</td><td>1</td><td>2</td><td>3</td></tr></table>	2	2	1	2	3	3	1	2
2	2	1	2	3					
D	<table border="1"><tr><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td></tr></table>	2	2	2	2	2	0	0	0
2	2	2	2	2					
E	<table border="1"><tr><td>1</td><td>5</td><td>3</td><td>4</td><td>4</td></tr></table>	1	5	3	4	4	7	1	5.12
1	5	3	4	4					

Aufgabe 4 : Caffe' (2 Punkte)

Warum sind *Code Clones* nicht grundsätzlich problematisch?

(2 Punkte)