Exercise: Cohesion, and coupling

Student name: **Student name:**
Student number: **Student number:**

Question 1 (Cohesion) The Linux kernel configuration tree gathers all multi-media devices in one group, even though they (mostly) do not interact with each other. What is the likely reason for such grouping?

They are grouped by topic (=their function)

Question 2 (Cohesion) Please give an example of a god class that may be justified?

Utility class
Math library class
Generated code

Question 3 (Cohesion/Coupling) Give two different disadvantages of maintaining duplicate code in a system.

- *Could forget to propagate an error fix*
- *Twice as much code to read and maintain*

Question 4 (Cohesion/Coupling) What is a potential disadvantage of factoring out common code into a common module?

- *Increased coupling (components or projects sharing code have to coordinate)*
- *Increased testing effort (if multiple project share the common code rather than their own copies, changes to the common code driven by one project will trigger retesting all projects)*
- *Performance (need an extra call to the common code) --- but this is often not a sufficient reason not to factor out*

Question 5 (Coupling) Suppose that procedure ProcessRequest() calls procedure ProcessElement(Element element, bool isVerbose), where isVerbose determines whether the ProcessElement is executed silently on element or whether each step within ProcessElement is logged. Name the two types of coupling that exist between these two procedures?

Data and control

Question 5 (Coupling) Which of the three types of coupling, namely control, data, and stamp coupling, is the lowest one?

Data

Question 7 (Coupling) Name two different disadvantages caused by stamp coupling.

- *Increased security risk (the extra data is now also exposed)*
- *Imprecise interface (does not make clear what the real dependency is)*