# Quiz 1 – A

Q1) Please write down your name and student ID [5 pt].

Q2) Compare the concept of software architecture with software design based on the following criteria: 1) the level of abstraction (the granularity) of the information provided and 2) the type of requirements that they cover [20 pt].

Architecture:

- Developing the skeleton, high-level infrastructure including deployment/execution environments
- Handling both functional and non-functional requirements

Design:

- Focusing on code-level design, low-level details
- Dealing mainly with functional requirements

Q3) What is the difference between the “Autogenerated Stovepipe” and the “Stovepipe system” architectural anti-patterns? [15 pt]

- Autogenerated Stovepipe → When migrating a local system to a distributed environment while continuing to use the old interfaces without introducing new ones for adaptation. The scope is at the application/system level.

- Stovepipe System → It is the problem of ad-hoc subsystem integration without a global plan. At the enterprise level, this occurs when subsystems are integrated without a comprehensive integration plan for the overall system.

Q4) What did you want to be when you were a child? The following answers are also acceptable: prefer not to share, I don’t have any, who cares, etc. [5 pt].

Q5) A research laboratory is developing an image enhancement platform for processing large volumes of satellite imagery. The platform must support multiple stages of transformation, such as improving image clarity, extracting useful visual features, and preparing the data for downstream machine learning models. To keep the system maintainable, each stage should be independently testable, reusable, and replaceable without disrupting the rest of the flow. Moreover, the system should use a standardized intermediate image representation so that different processing components can be integrated smoothly, even if developed by different teams.

Which software architectural style would be best suited for designing a system to manage this flow? Justify your answer based on the provided requirement [55 pt].

The best architectural style is Pipe-Filter.

- Each image processing stage can be implemented as a filter, making it independently testable, reusable, and replaceable.
- The standardized intermediate image format acts as the pipe, enabling smooth data transfer between filters and integration across teams.
- Loose coupling improves maintainability and extensibility—new stages can be added or swapped without redesigning the whole system.
- The architecture scales well for large satellite imagery since filters can process data incrementally or in parallel.

Conclusion: Pipe-Filter fulfills the requirements of modularity, standard interfaces, independent development, and scalability