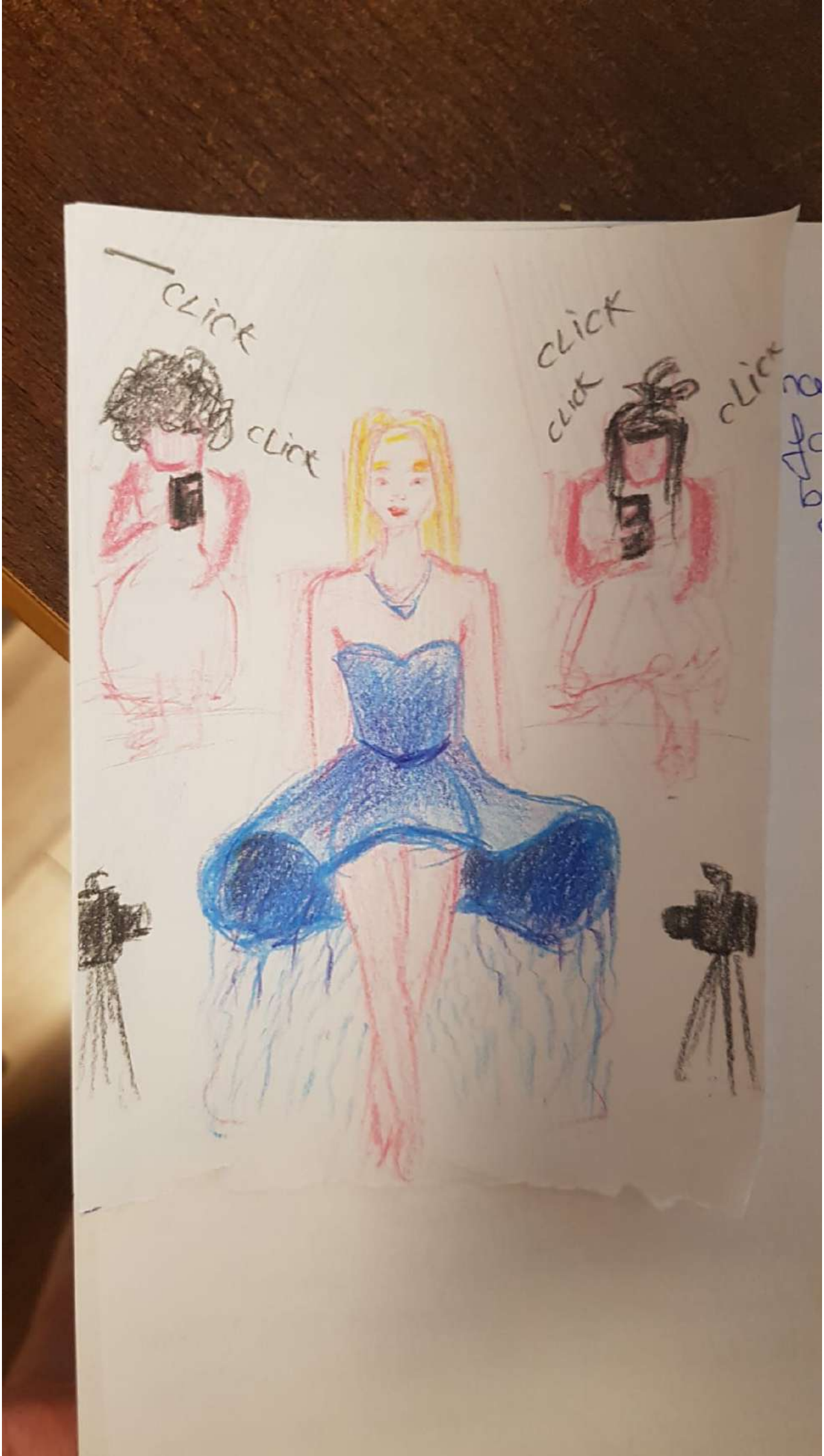
CLICK
CLICK
CLICK
CLICK
CLICK

DEL CHECKING

LINCU ALICE RAMONA

The dixit word represent the concept of model checking. The „model" is represented by an actual ~~model~~ fashion model at a fashion shows. The „checking" part is illustrated by cameras and phones that check the fashion model. ~~and~~

ii. Short Question

HINCU ALICE

Finite State Machines are a fundamental concept used to model systems with a finite number of states and well-defined transitions between these states. An FSM consists of a limited set of states, transition functions, initial state and final states. The transition functions determine how the system shifts from one state to another, typically in response to external inputs. This makes FSMs ideal for designing software and hardware systems like digital circuits, parsers and control systems. The power of FSMs lies in their ability to represent complex behaviours through a combination of simple, discrete states, providing a structured and manageable framework for analyzing and designing systems. In computer science and engineering, FSMs are instrumental in understanding the behavior of systems and creating predictable, efficient solutions.

iii Long Question

Finite State Machines (FSMs) and Petri Nets (PNs) are both models used for describing systems with dynamic behaviors, but they differ in their structure and the types of systems they are best suited to represent.

~~FSMs~~ One of the key differences between these two lies in their approach to concurrency. In FSMs, the assumption is often that events occur in a sequential and deterministic manner. However, PNs capture the notion of concurrency, allowing for the modeling of systems where multiple states can be active at the same time - this makes them a more natural fit for scenarios where the exact sequence of events is not strictly predetermined.

FSMs are easier to design and understand because of their simpler and more straightforward structure. They are well-suited for systems with a clear linear or cyclic pattern of operation. PNs offer a higher degree of expressiveness, capable of representing more complex interactions and states. However, this increased complexity can make them more challenging to design and analyze.

↳

HINCU ALICE
RAMONA

In conclusion, while both FSMs and Petri Nets are powerful tools for modeling dynamic systems, they each have their unique strenghts and limitations. FSMs are good in environments where simplicity, determinism, and sequential processes are key, whereas PNs are good in scenarios requiring the modeling of concurrency, complex interactions and asynchronous processes. The choice between using a FSM or a PN largely depends on the specific requirements and nature of the system being modeled.