BBM 486 – DESIGN PATTERNS

2021-22 Fall Midterm Exam

1. What is polymorphism? Briefly explain with an example. Convince me that you understand this concept and its use? **(10 points)**

2. Assume that you are building a symbolic AI engine that can do classification. The users will be able to interact with the engine in two different modes: *training* and *inference*. In the *training mode*, the users will be able to feed training data (positive and negative examples of a certain concept), and the engine will explore if there exists a discrete model that can serve as a classifier for this concept. The user should be able to *display* the models associated with a concept graphically. In the *inference mode*, the user will provide a data sample for the engine to infer if it is an example of that concept or not.

   The engine will support at least 3 types of discrete models: finite-state automata (FSA), visibly push-down automata (VPA) and push-down automata (PDA), as follows:

   A *finite state automaton* is defined as a quintuple (Q, I, δ, q_0, F), where
   - Q is a finite set of *states*,
   - I is a finite set of *input symbols*,
   - $\delta: Q \times I \to Q$ is the *transition function,*
   - q_0 is the *initial state*, and
   - $F \subseteq Q$ is the set of *final states*.

   A *pushdown automaton* is defined as a seven-tuple $(Q, I, \Gamma, \delta, q_0, Z_0, F)$, where
   - Q is a finite set of *states*,
   - I is a finite set of *symbols*,
   - Γ is a finite set of *stack symbols*, such that $\Gamma \cap I = \emptyset$,
   - $\delta: Q \times (I \cup \{\varepsilon\}) \times \Gamma \to Q \times \Gamma^*$ is the *transition function,*
   - q_0 is the *initial state*,
   - Z_0 is the i*nitial stack symbol*, and
   - $F \subseteq Q$ is the set of *final states*.

   A *visibly pushdown automaton* is defined as a seven-tuple $(Q, \Sigma, \Gamma, \delta, q_0, F)$, where
   - Q is a finite set of *states*,
   - Σ is a finite set of *symbols*, which is partitioned into three sets Σ_c, Σ_r and Σ_i denoting the set of *call* symbols, *return* symbols and *internal* symbols,
   - Γ is a finite set of *stack symbols*, which contain a special symbol $\perp \in \Gamma$ denoting the empty stack,
   - $\delta = \delta_c \cup \delta_r \cup \delta_i$ is the *transition function,* which is partitioned into three parts corresponding to call transitions, return transitions and internal transitions, as follows:
     $$\delta_c: Q \times \Sigma_c \to Q \times \Gamma$$
     $$\delta_r: Q \times \Sigma_r \times \Gamma \to Q$$
     $$\delta_i: Q \times \Sigma_i \to Q$$
   - q_0 is the *initial state*, and
   - $F \subseteq Q$ is the set of *final states*.

(a) Design your application and provide your class diagram. Explain which design patterns have you considered? **(25 points)**
(b) Provide a Java implementation of your design at a high-level. **(25 points)**

3. After graduating from Hacettepe University, you are hired to manage a large software development project. You analyzed the project scope and identified 4 subsystems (A, B, C, D) to be build or acquired. Here A is a generic module, and B, C and D are application-specific modules. The dependencies among A, B, C, D and your assessment of uncertainty over their interfaces are as follows:

Module	Module	Dependency	Uncertainty
A	B	Yes	Low
A	C	No	Low
A	D	No	None
B	C	Yes	Low
B	D	Yes	Low
C	D	Yes	High

(a) If you follow the advice given in the *Modules and Transactions* paper reviewed in the class, how would you go about implementing this project? Explain which governance structures would you choose for each module? **(20 points)**
(b) After evaluating this, you want to double check your approach using design principles "*Encapsulate what varies*" and "*Strive for loosely coupled designs between objects that interact*". What would you change in your approach if you follow these two design principles? **(20 points)**