

Read the following scenario CAREFULLY then answer ALL SIX questions:

Welcome to OSCORP Cooperation; a future science-fiction company that creates genetically mutated monsters to fight against Spiderman. It is required to create a software to *create*, *control* and *monitor* these monsters.

Creating a monster is a very complicated task. Before finally pressing the create button, the scientist must define all layers of the monster body. The system must expose the methods: `buildBones()`, `buildMuscles()`, `buildBlood()`, `buildNervousSystem()`, etc. to the scientist using the software. Once the system successfully constructed one monster stereotype, his/she can create *thousands* of them by simply *cloning* the original one. Directly after the cloning processes, the scientist is allowed to change only few DNA values to create the mutant. The rest of the DNA-values remain unchanged for the same monster *generation*. The system must keep track of all DNA-values and vital values for *all* monsters in an efficient way.

The scientists in the *control* room must be able to issue commands to the monsters on the field. These commands are very generic, since every generation of monsters might handle the command in a different way. However, the mechanism must always work. Each monster receives each and every command and tries to execute without hesitation. If it fails, it reports to its leading monster, till the failure is either locally avoided or it is reported to the command center.

In the command center, the scientists *monitor* the battle field. Each monster changing its position by more than 20 feet must be seen on a radar-like map in a blinking red color. The scientists should be able to select a group of related monsters and monitor their vital values; such as common DNA-values and can drill down to a single monster to get its individual values such as the energy level.

Question 1 (10 grades):

Draw a simple use case diagram illustrating the interaction of the scientists with the system.

Question 2 (30 grades):

Identify 2 creational design patterns, 1 structural design pattern, and 3 behavioral design patterns. Explain briefly (2-3 sentences) the usage of each design pattern in the system.

Question 3 (10 grades):

Identify one use for the *Marker Interface* design pattern in your solution. Explain its usage briefly (2-3 sentences).

Question 4 (20 grades):

Draw a detailed UML class diagram for each of the design patterns you applied in Question 2. Use the proper naming conventions in the context of OSCORP.

Question 5 (10 grades):

Write the code skeleton that implements your design in Question 4.

Question 6 (10 grades):

Draw the sequence diagram between the monster, its direct leader, and the control system.

GOOD LUCK

CP/221