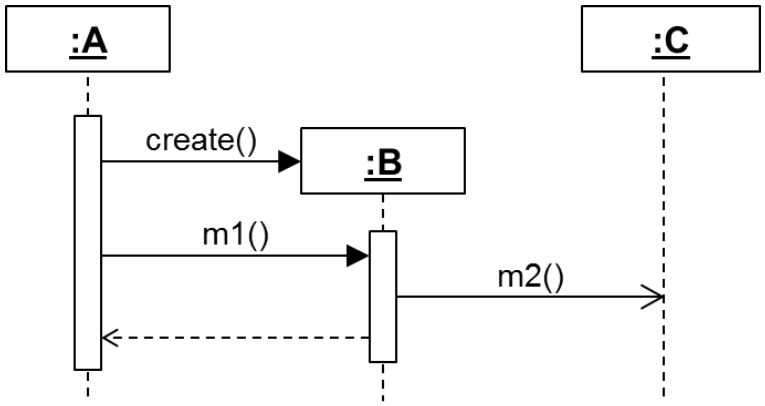
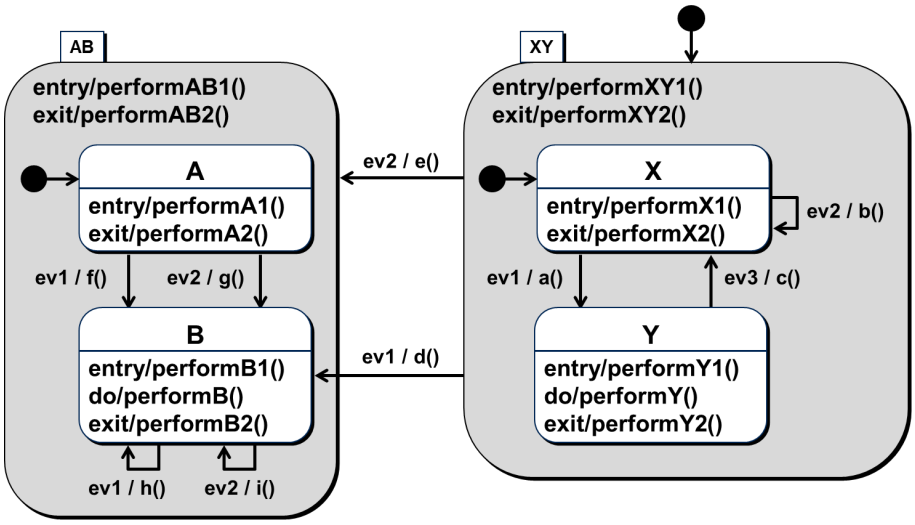
Question 1 (12 marks)

Explain the meaning of the three types of ***messages*** create(), m1(), and m2() shown in the following sequence diagram in one sentence each.



Question 2 (12 marks)

Given the following state machine, assume that you are in state X. List the ***actions*** in the order in which they are executed when event ev1 and then event ev2 occur.



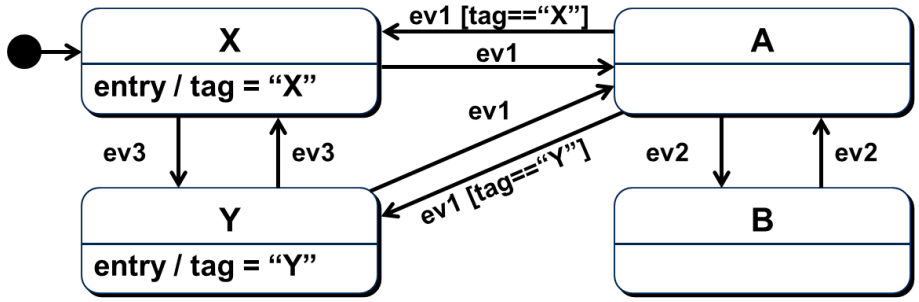
Question 3 (15 marks)

Given the following ***Umple specification***, draw the corresponding state machine. Make sure that all elements shown in the Umple specification appear in the state machine.

```
class MyClass {
	status {
		S1 {
			ev1 [isActive()] / { count = 1; } -> S2;
			ev1 [isNotActive()] / { count = 0; } -> S4;
		}
		S2 {
			entry / { taskA(); }
			exit / { taskB(); }
			ev2 -> S5;
			S3 {
				ev1 / { count++; } -> S4;
			}
			S4 {
				entry / { taskC(); }
				exit / { taskD(); }
				ev1 / { count--; } -> S3;
			}
		}
		S5 {
			ev1 -> S1;
		}
	}
}
```

Question 4 (12 marks)

Simplify the following state machine with the help of a ***history state***.

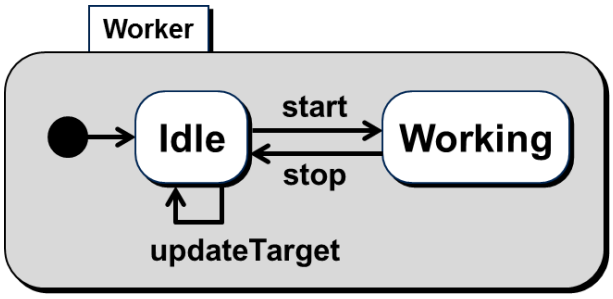


Question 5 (12 marks)

Describe three advantages of the ***Model-View-Controller (MVC) Pattern*** in two to three sentences.

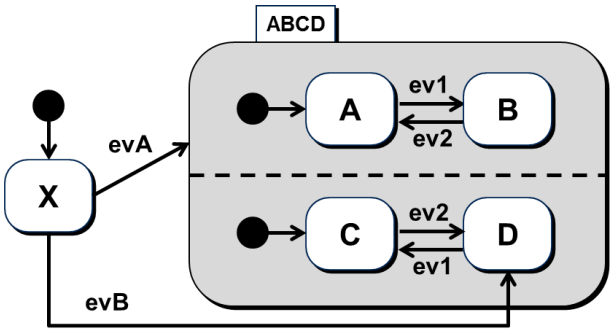
Question 6 (10 marks)

List the ***fields and methods*** that are ***generated*** by Umple for class Worker from the following state machine. You do not need to state the details of the generated code nor do you need to discuss the method bodies.



Question 7 (15 marks)

Flatten the following state machine.



Question 8 (12 marks)

Describe <u>two</u> ways how regions in a state machine may ***interact*** with each other in <u>one or two</u> sentences.