Питання

1. Можливі варіанти зв’язків між прецедентами.

* У мові UML визначено такі типи відносин: залежність , асоціація , узагальнення і реалізація . Ці відносини є основними єднальними конструкціями UML і також як сутності застосовуються для побудови моделей. Залежність (Dependency) – це семантичне відношення між двома сутностями, при якому зміна однієї з них, незалежної, може вплинути на іншого. Асоціація (Association) – структурне ставлення, що описує сукупність смислових або логічних зв'язків між об'єктами. Узагальнення (Generalization) – це відношення, при якому об'єкт нащадок може бути підставлений замість об'єкта предка. При цьому, відповідно до принципів об'єктно-орієнтованого програмування, нащадок успадковує структуру і поведінку свого предка. Реалізація (Realization) є семантичним відношенням між класифікаторами, при якому один класифікатор визначає зобов'язання, а інший гарантує його виконання.

2) Що таке нотації і метамодель в UML?

* Нотація - сукупність графічних об'єктів, які використовуються в моделях. Як приклад на діаграмах показано, як в нотації діаграми класу визначають поняття і предмети типу «клас», «ассоціацція», «множинність» і т. д.  
  Метамодель - діаграма, що визначає нотацію. Метамодель допомагає зрозуміти, що таке добре організована і синтаксично правильна, модель.

3) Назвіть мінімум 3 проблеми, з якими можуть зіштовхнутися розробники \_ при використанні ЮМЛ?

* По-перше / обмін думками з приводу концептуальної моделі можливі лише тоді, коли всі учасники дискусії говорять однією мовою. Як правило, при розробці проектів компаніям доводиться винаходити власні мови, і новачкові непросто здогадатися, про що йде мова. По-друге, не можна скласти уявлення про певні аспекти програмних систем без моделі, що виходить за межі текстового мови програмування. Так, призначення ієрархії класів можна, звичайно, зрозуміти, якщо уважно вивчити код кожного класу, але сприйняти всю структуру відразу і цілком не вийде.  
  По-третє, якщо автор коду ніколи не втілював в явній формі задумані ним моделі, ця інформація буде назавжди втрачена, якщо він змінить місце роботи. У кращому випадку її можна буде лише частково відтворити виходячи з реалізації.

4) Чому юмл називають мовою конструювання?

* відображення моделі на мову програмування дозволяє здійснювати, пряме проектування: генерацію коду з моделі UML в якийсь конкретний; мова. Можна вирішити і зворотну задачу: реконструювати модель за наявною; реалізації. Зворотне проектування не представляє собою нічого незвичайного. Якщо ви не закодували інформацію в реалізації, то ця інформація втрачається! при прямому переході від моделей до коду. Тому для зворотного проектування! необхідні як інструментальні засоби, так і втручання людини. Поєднання прямий генерації коду і зворотного проектування дозволяє працювати як в графічному, так і в текстовому поданні, якщо інструментальні програми забезпечують узгодженість між обома уявленнями

5) Назвіть і опишіть найчастіші галузі, де застосовуються ЮМЛ?

* Мова UML призначений насамперед для розробки програмних систем. Його використання особливо ефективно в наступних областях: інформаційні системи масштабу підприємства; банківські та фінансові послуги; телекомунікації; транспорт; оборонна промисловість, авіація та космонавтика; роздрібна торгівля; медична електроніка; наука; розподілені Web-системи. Сфера застосування UML не обмежується моделюванням програмного забезпечення. Його виразність дозволяє моделювати, скажімо, документообіг в юридичних системах, структуру і функціонування системи обслуговування пацієнтів у лікарнях, здійснювати проектування апаратних засобів.