一、选择题（30 分）

1-5：A　　A　　C　　B　　A

6-10：B　　A　　B　　C　　C

二、简答题（30 分）

1：Key Points:

Design patterns aim to enhance the maintainability and extensibility of the system. And their goal is to reduce the influence of changing in requirements. You can find many examples to explain the point.

UP, especially iterative process is another way to achieve this goal.

2：

Similarity：

(1) A well-known problem/solution pair that can be applied in new contexts. Be described with the same design patterns template, include name, intent, problem， context, enforce, solution, etc.

(2) Vocabulary of software designer. A way of reusing abstract knowledge about a problem and its solution

Difference：

(1) Content:

GRASP: General Responsibility Assignment Software Patterns, Responsibility-Driven

GOF: Solution scheme is specified by describing its constituent components, their responsibilities and relationships, and the ways in which they collaborate [POSA]. The component design.

(2) Abstraction Level: Low, High

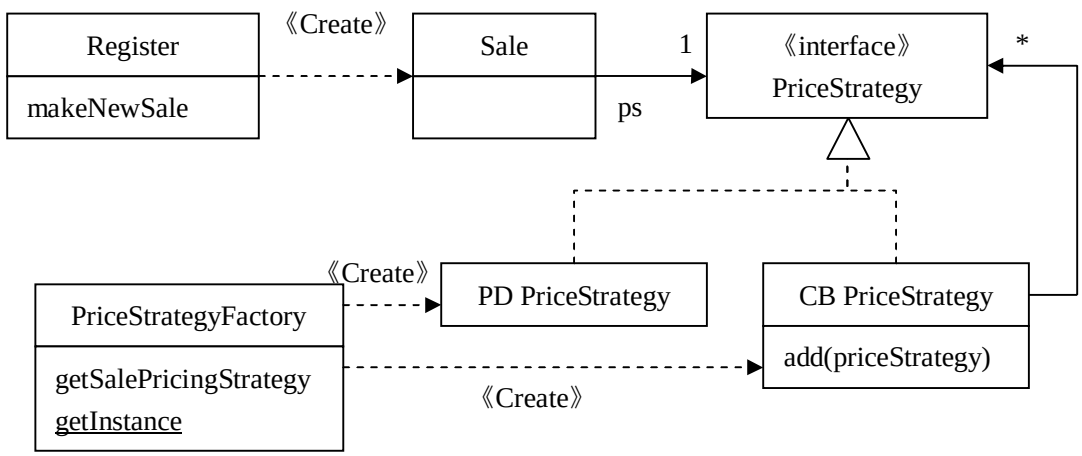
(3) Methods: Design principles, Component design techniques.

Relations:

GRASP always as a tools to explain the other design pattern. Such as components' responsibility, Low coupling & high cohesion.

三、作图题

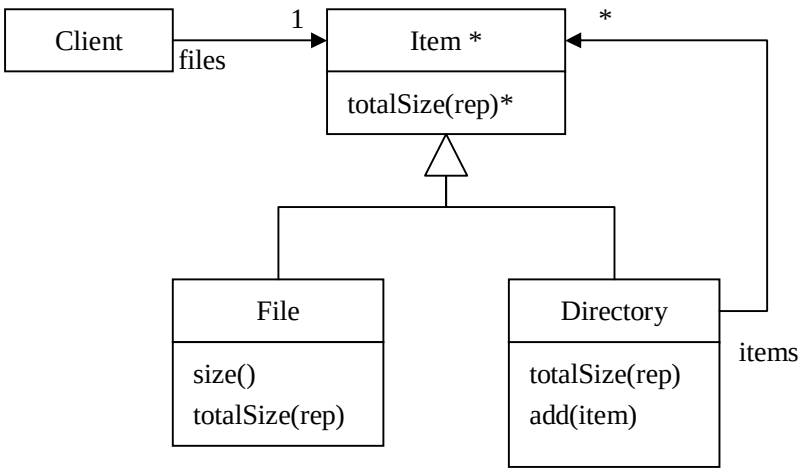
1:：注意，该题有多种答案，但一定要表达 CB PriceStrategy 与 PD PriceStrategy 组合关系；工厂的单实例实现概念（静态方法获得实例）。



2：答案要点：要保持通信图与顺序图的等价性。Activation bar 要正确反映通信图里的序号关系。类名前要有冒号，箭头为实心箭头。

四、案例分析

1 (a)

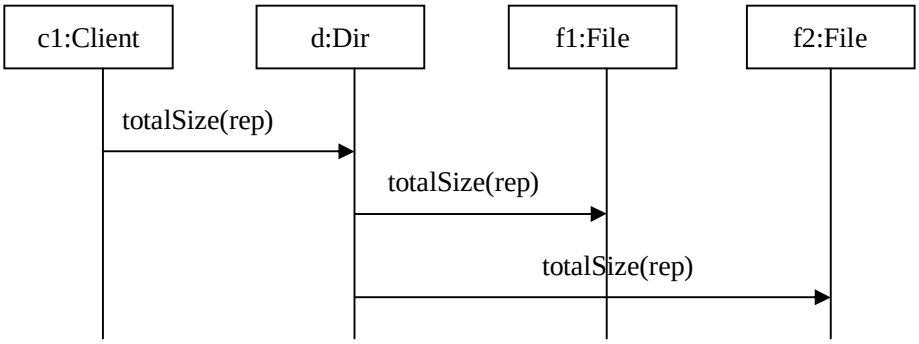


(b)

```
Class File
Public totalSize(rep)(){
   Rep.add(size());
}

Class Directory()
Public totalSize(rep){
   For each item in items do
      Item. totalSize(rep);
   End For
}
```

(c)



2 答案要点

可采用 Factory Method 模式或者 Service Locator 模式

要做到把创建连接的代码跟业务逻辑分离，如返回给业务逻辑方法的是一个借口而不是类

当创建连接的代码发生变更时（如数据库类型变了），业务逻辑的代码完全不需要作任何修改。