

настоящее время термин «Корабельный Якорь» стал означать классы и методы,, которые по различным причинам не имеют какого-либо применения в приложении и в принципе бесполезны. Они только отвлекают внимание от действительно важного кода.

**Busy spin.** Техника, при которой процесс непрерывно проверяет изменение некоторого состояния, например ожидает ввода с клавиатуры или разблокировки объекта. В результате повышается загрузка процессора, ресурсы которого можно было бы перенаправить на исполнения другого процесса. Альтернативным путем является использование сигналов. Большинство ОС поддерживают погружение потока в состояние «сон» до тех пор, пока ему отправит сигнал другой поток в результате изменения своего состояния.

**Caching Failure.** «Кэширование Ошибки» – тип программного бага (bug), при котором приложение сохраняет (кэширует) результаты, указывающие на ошибку даже после того, как она исправлена. Программист исправляет ошибку, но флаг ошибки не меняет своего состояния, поэтому приложение все еще не работает.

### Задания к главе 5

#### Вариант А

Выполнить описание логики системы и использовать шаблоны проектирования для определения организации классов разрабатываемой системы. Использовать объекты классов и подклассов для моделирования реальных ситуаций и взаимодействий объектов.

1. Создать суперкласс **Транспортное средство** и подклассы **Автомобиль**, **Велосипед**, **Повозка**. Подсчитать время и стоимость перевозки пассажиров и грузов каждым транспортным средством.
2. Создать суперкласс **Пассажироперевозчик** и подклассы **Самолет**, **Поезд**, **Автомобиль**. Задать правила выбора транспорта в зависимости от расстояния и наличия путей сообщения.
3. Создать суперкласс **Учащийся** и подклассы **Школьник** и **Студент**. Определить способы обучения и возможности его продолжения.
4. Создать суперкласс **Музыкальный инструмент** и классы **Ударный**, **Струнный**, **Духовой**. Определить правила организации и управления оркестром.
5. Создать суперкласс **Животное** и подклассы **Собака**, **Кошка**, **Тигр**, **Мустанг**, **Дельфин**. С помощью шаблонов задать способы обитания.
6. Создать базовый класс **Садовое дерево** и производные классы **Яблоня**, **Вишня**, **Груша**, **Слива**. Принять решение о пересадке каждого дерева в зависимости от возраста и плодоношения.

### Тестовые задания к главе 5

#### Вопрос 5.1.

Какой шаблон создает объекты путем их копирования?

- 1) Factory;
- 2) Prototype;
- 3) Builder;
- 4) Singleton.

**Вопрос 5.2.**

Какие из шаблонов относятся к порождающим? (выберите два)

- 1) Factory;
- 2) Command;
- 3) Strategy;
- 4) Singleton.

**Вопрос 5.3.**

Какой шаблон позволяет обращаться к группе объектов таким же образом как и к одному?

- 1) Visitor;
- 2) Composite;
- 3) Prototype;
- 4) Observer.

**Вопрос 5.4.**

Какой шаблон реализует постоянную часть алгоритма, а реализацию изменяемой оставляет потомкам?

- 1) Strategy;
- 2) Decorator;
- 3) Template Method;
- 4) Visitor.

**Вопрос 5.5.**

Какой шаблон подменяет собой сложный объект и контролирует доступ к нему.

- 1) Adapter;
- 2) Decorator;
- 3) Proxy;
- 4) Bridge.