**Описание консольной игры-стратегии «Олимпийские игры»**

Цель игры: завоевать как можно больше медалей, то есть попасть на лидирующие позиции в топе команд из разных стран(желательно на первое место).

В начале игры пользователь выбирает страну, за которую он будет играть: Норвегия или Россия. Далее он должен сформировать свою команду. В состав команды могу входить только фигуристы, биатлонисты и хоккеисты (спортсмены других видов спорта не допущены из-за допинга). Чтобы сформировать команду, пользователю нужно будет ввести количество спортсменов по каждому виду спорта (максимально 15 человек по фигурному катанию, 20 биатлонистов и 25 хоккеистов).

Фигуристам задается счёт – их баллы за прокат. У каждого есть выносливость и энергия, вторая по мере выполнения упражнений будут уменьшаться. Если параметр обнулиться, спортсмен теряет сознание и на этом заканчивается его выступление.

Хоккеисты имеют энергию – изменяемый параметр, точность и выносливость – статические. Могут забивать шайбу и допускать возможность того, что соперник забьет им.

Биатлонисты могут стрелять, ускорятся и отдыхать во время езды на лыжах. Поэтому есть атрибут скорость, который изначально для разных стран разный. Результативность попаданий зависит от энергии и точности в данный момент времени. Есть 4 огневых рубежа, в каждом по 5 выстрелов. За каждый промах +1 минута. Побеждает тот, у кого наименьшее время.

Так как есть две страны, у которых одинаковые спортсмены с одинаковыми функциями, но разными параметрами, нам удобно использовать фабрику для производства этих спортсменов.  
Для каждого вида спортсменов создается свой класс. В нем определяются одинаковые атрибуты у всех спортсменов такого вида. Далее в классах-наследниках уже для каждой страны переопределяются те атрибуты, которые должны иметь разные значения для каждой страны. В классе AthleteFactory мы объявляем виртуальными функции создания спортсменов, и далее уже пишем фабрику, которая служит для создания продуктов всех основных типов, для каждой конкретной страны. Это удобно, так как мы создаем взаимосвязанные объекты, сам процесс порождения объектов скрывается. Например, в классе Game мы как раз создаем спортсменов. Но мы не пишем, что хотим создать Российского фигуриста, пишем просто «создать фигуриста» так как программа уже сама поняла, из какого государства будет спортсмен.