**Эссе на тему «Haskell vs Java».**

Пушкарева Татьяна ВШ ИТИС 11-202

Рассуждая о преимуществах того или иного языка программирования, сразу задумываешься о том, для каких целей он может использоваться, за какой период его удастся освоить, насколько широки его возможности.

А теперь рассмотрим преимущества языков Haskell и Java, сравним их возможности и функционал. Начнем с особенностей Haskell. Во-первых, он функциональный, чего нельзя сказать о Java (хотя в новой версии Java 8 появились долгожданные элементы функционального программирования). Во-вторых, чистота. В функционально чистых частях программы многие ошибки исключаются на этапе компиляции. Более того, функционально чистые функции следуют основному закону Haskell-а: «Вызов функции с одними и теми же параметрами всегда даст один и тот же результат». В третьих, ленивость. Haskell вычисляет значение тогда, когда оно реально необходимо. Четвертая особенность состоит в самом синтаксисе языка и в том, как его нужно читать. Действительно, основная особенность языка состоит в компактности кода, несмотря на весь объем вычислений, которые он выполняет. Однако компактность достигается тем, что многие действия заменяются символами. На первый взгляд выглядит очень непривычно и неинформативно. Однако при непосредственном использовании этих символов код выглядит более компактным и аккуратным, чем в Java.

Теперь о Java. Сильной стороной Java является ее способность управлять несколькими потоками. Эта возможность используется многими веб-сайтами с высокой посещаемостью. В Haskell, как и во многих современных языках программирования, отсутствуют фигурные скобки. Однако они есть в Java, что помогает начинающим программистам в понимании вложенности блоков кода, в то время как вложенность блоков в Haskell отслеживается отступами (компилятор Haskell очень чувствителен к отступам).

В качестве примера реализация функции факториала на обоих языках.

Haskell

factorial n =

if n == 0

then 1

else

n \* factorial (n-1)

Java

int p = 1;

int n = 4;

for (int i = 1; i <= n; i++) {

p \*= i;

}

Из примеров видно, что разница объема кода в обоих случаях несущественная, однако при использовании языка Haskell мы избавляем себя от таких вещей, как указание типа переменной, циклов. Функция вычисляется рекурсивно.

Рассмотрев особенности этих языков все-таки прихожу к выводу, что однозначного ответа на вопрос «Какой язык лучше?» в принципе не существует. А выбор языка обусловлен поставленными задачами, опытом программиста и даже привычкой.

Если найти в себе силы изучить такой своеобразный язык как Haskell, то потенциальный работодатель наверняка оценит стремление к самообразованию и нестандартному мышлению.