

## Президент Латвии Раймонд Вейонис,

министр образования и науки Марите Сейле, заведующий Государственным центром содержания образования при МОН Гунтар Цатлак на торжественной церемонии 5 октября в Доме Черноголовых наградили лауреатов международных олимпиад по учебным предметам и их педагогов



Награду получает ведущий преподаватель наших курсов  
С.И.Мельник.

## Новый спецкурс Ruby on Rails

Ruby on Rails (RoR) - современный фреймворк для разработки веб-сайтов и сервисов.

В курсе вы познакомитесь с языком программирования Ruby (кстати, сам Ruby on Rails написан на Ruby). Создатель языка Мацумото при разработке Ruby сделал акцент на малое время разработки, понятность и простоту синтаксиса. И ему действительно это удалось. Вы увидите насколько просты и элегантны синтаксические конструкции языка. Вы это поймёте, если станете на путь Руби (Ruby Way).

Программирование будет весёлым и продуктивным.

Вместе с рубином (Ruby) вы получаете и 6392 драгоценных камней\* (gems - так называют библиотеки для Руби). Один из самых ценных камней это Ruby on Rails.

Создать сайт без хорошего фреймворка - это сложный и долгий труд. Это, как строить дом с нуля - фундамент, стены, крыша... Разве не лучше было бы сразу заниматься интересными вещами такими, как внешний вид, интерьер, технические навороты. RoR позволяет строить сайт, не думая о низкоуровневых деталях, каркас веб-сайта или сервиса уже написан, остаётся только начать его использовать.

RoR основан на архитектуре Модель-Представление-Контроллер (MVC). Звучит, возможно, не совсем понятно, но на практике вы увидите всю логичность и последовательность такого подхода.

Веб-сайты используют большое количество технологий, мы сконцентрируем своё внимание на серверной (back-end) части. HTML, CSS, Javascript - всё это тоже встречается, но практически всё это сделает фреймворк за нас. Нам нужны будут только минимальные знания. Поэтому от вас требуется только:

0) желание (будет много материала для самостоятельного изучения);

1) знание ООП;

2) навык программирования на любом языке с C-подобным синтаксисом (C, C++, Java и т.д.)

Высокоуровневые детали курса: мы разберёмся как работает веб; познакомимся с Unix средой (фреймворк кроссплатформенный, поэтому вы сможете всё то же делать и в среде Windows); рассмотрим паттерн ActiveRecord (работа с базой данной), будем отсылать письма, загружать файлы, займёмся интернационализацией страниц (i18n), поговорим о тестировании и отладке, о безопасности веб-систем. Это даст нам возможность построить небольшой социальный сайт: посты, комментарии, друзья, лайки, лента и др. Будут в курсе и интерактивные уроки: консоль в браузере - требуется написать программный код, чтобы пройти дальше. Кому нравится решать загадки, то почему бы не делать это на Руби.

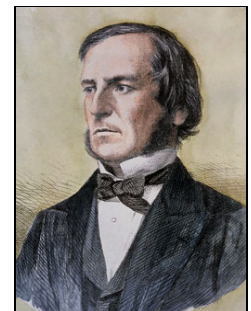
Мы будем использовать только open source программы, вы всегда сможете заглянуть им внутрь. И, кстати, в мире Ruby приветствуется, если вы видите код. Документация к ним есть и довольно хорошая, но ещё лучше видеть код.

\*кол-во библиотек на официальном сервере  
rubygems (2015.11.01)

## George Boole

02.11.1815-08.12.1864.

Английский математик Д. Буль в 1854 году опубликовал работу "Исследование законов мышления", заложив основу того, что мы сегодня знаем как математическую логику. В этой работе Буль свел логику к чрезвычайно простому типу алгебры - алгебре логики, которая представляла собой систему символов и правил,



применяемую к высказываниям - утверждениям, которые могут быть истинными или ложными. Теория, основанная на трех основных действиях — AND (и), OR (или), NOT (не), — стала в XX веке основой для разработки проекта ЭВМ.

## Жизнь после университета: что значит работать программистом?

Александр Шантар

Александр - выпускник 2000 года элитной группы курсов «Progmeistars», выпускник University of Waterloo (Канада) 2005 года. Во время учебы начал работать в IBM. Сейчас работает программистом там же. Первая статья Александра, которая называется «Счастливого выбора!», была опубликована в Курсовике №132.



Итак, почти двадцать лет учебы за плечами, долгожданный диплом у вас в руках. Что делать дальше? Наступает один из наиважнейших и интереснейших с моей точки зрения моментов - поиск работы. В этой статье я опишу, что ожидает от вас работодатель, что вам ожидать от работодателя, и какие варианты «путей-дорог» и движения по ним лежат перед вами. Цель этой статьи - облегчить переход из академической среды в деловую, и ускорить рост вашей карьеры.

### Что работодатель ожидает от вас.

Первое, с чем вы столкнетесь - это работа в группе/команде/коллективе. До недавнего времени вы учились ОДИН. Успехи ваших сокурсников никак не отражались на вас. Поэтому надо (на пока) убрать своё "Я" подальше. Кстати, это требование относится не только к программистам.

В порядке важности, что вы должны проявить в работе.

1. Понимание и выполнение протоколов (бизнес процессов). Если в компании что-то делается вот так, то и от вас будут ожидать того же, даже если вам это не нравится/неудобно/странно. Очень часто стоимость значительной переделки процесса в разы выше пошагового улучшения. Поэтому, если хотите что-то поменять, будьте готовы продемонстрировать значительную выгоду.
2. Навыки общения (электронная почта, телефон, встречи, презентации, конференции). Но самое важное общение – это, конечно же, общение с начальством. А что начальство обычно хочет знать от вас? Что все идет по плану! Это значит - РЕГУЛЯРНЫЕ рапорты о текущем состоянии дел.
3. Умение концентрироваться на самом ВАЖНОМ... с точки зрения начальства. Не самом интересном, не самом перспективном, не самом сложном, не самом срочном. На вас будет висеть дел на десять лет вперед (и это хорошо, если все

дела сделаны, а новых нет - я бы стал волноваться). Единственное оправдание перед начальством почему что-то не сделано: делал другое, более важное. А что самое важное? Что приносит самую большую прибыль. Вы знаете КАК и ЗА КАКИЕ ДЕНЬГИ продается ваш код? Если нет, то лучше слушайте начальство. Кстати, это означает иногда бросить все и делать вот это. А потом и это бросить и делать вот то.

4. И вот только на этом месте стоят навыки программирования. Скорее всего вы будете работать с уже написанным кодом (не с нуля). Как быстро вы разберетесь что к чему? Как быстро сможете починить что-то, не сломав другое? Насколько хорош ваш код?

5. И, наконец, умение учиться. Нужно научиться, систематизировать данные, обобщать, делать выводы и применять эти выводы в дальнейшей работе. Нужно разбираться в своей компоненте, в ее сильных и слабых сторонах. как она взаимодействует с другими частями продукта, выявлять зависимости. Разбираться в стратегии развития продукта, динамике изменений в компонентах. Научиться понимать, что клиенту на самом деле нужно и как это «нужно», в свою очередь, повлияет на стратегию развития продукта лет на 10 вперед. Ну, а тут и начальство к вам прислушиваться станет.

Каким бы вы ни были хорошим программистом, если у начальства не будет уверенности в предыдущих пунктах, то оно просто не будет вам доверять.

**Что вам ждать от работодателя,** в порядке важности.

1. Интересную работу. Если работа неинтересна, то получается, что вы ходите на пытку за деньги. И, как следствие, падение продуктивности, что очень видно со стороны. Поэтому, ситуации, когда работа не нравится надо решать, и решать быстро. Например, можно поговорить с начальством, перевестись в другой отдел. Можно посмотреть на работу с другой стороны. Можно поменять работу. Можно поменять профессию.

2. Заработной платы. Знайте сколько стоят ваши услуги и требуйте соответствующей зарплаты. Заработная плата не обязательно ограничивается деньгами - есть отпуск, дополнительная медицинская страховка, скидки на услуги (например, корпоративный план на телефон/интернет/аренду машины), пенсионные накопления, возможности повышения квалификации (в том числе и сертифицирование). Также надо оценивать надежность компании и перспективы её существования лет на 10 вперед. При приеме на работу ВСЕГДА просите больше - в худшем случае просто не дадут. Можно попросить повышение и в неденежном эквиваленте - в виде лишней недели отпуска к примеру. Просить повыше зарплату на интервью очень ВАЖНО. Часто - это ваш единственный неконфронтационный шанс. Также часто рост зарплаты исчисляется в процентах (например 3% в год), так что иметь больше базу никак не помешает - это как ипотека наоборот.

3. Карьерного роста. Карьера - это все то, что вы сделали на работе за свою жизнь. Всегда старайтесь делать что-то новое. Смотрите на себя как на персонажа из компьютерной игры: ваша карьера - это ваш уровень в игре. Только рост в карьере обеспечит интерес в работе и постоянное повышение зарплаты.

(Окончание следует)