



Оригинал публикации: <https://codeforces.com/blog/entry/67549>

09.06.2019

## ФИИТ УрФУ. Новая программа обучения разработчиков в Екатеринбурге

Автор [хороший](#)

Привет, я — Павел Егоров, 13 лет пишу промышленный код, 10 лет веду курсы и школы по программированию, 5 лет развиваю [онлайн платформу для обучения программистов](#), 3 года учу учить — руководжу отделом обучения разработчиков в компании СКБ Контур. А теперь я ещё и руководитель образовательной программы.

### О чём речь?

Образовательные программы СКБ Контур 10 лет делают и поддерживают разные движухи вокруг ИТ-образования: от спортивного программирования (в частности, [ICPC World Finals 2014](#)) и CTF до собственных курсов и [МООС-площадки](#). Если кому интересно — [тут много подробностей](#).

Много лет мы постепенно улучшали ИТ-образование в бакалавриате УрФУ, но у сторонней компании для этого не так много возможностей. А в прошлом году внезапно появилась возможность «сразу всё сделать правильно»: в очередной раз обсудив нашу давнюю мечту «сделать свой факультет» с директором СКБ Контур и ректоратом мы получили поддержку по всем фронтам. В итоге взяли под своё крыло направление подготовки Фундаментальная информатика и информационные технологии (ФИИТ) на бывшем Матмехе УрФУ.

Концепт у нас довольно простой: принять лучших студентов и уделить им столько внимания, сколько есть не во всех топовых столичных вузах. Через несколько недель на ФИИТ начинается набор, а с сентября первые 50 человек начнут учиться по-новому. Как именно — расскажу в статье .

И сразу же дисклеймер **1**: спортивное программирование — это круто и здорово, мы будем поддерживать ICPC (ниже будет конкретика), но мы делаем программу не для спортивных программистов, а для разработчиков.

Дисклеймер **2** мы постарались сделать этот пост максимально не рекламным, а

конкретным, но, поскольку мы рассказываем о том, что ещё только предстоит сделать, а не о случившемся, избежать этого полностью невозможно.

## Что изменится?

Все изменения происходят не на пустом месте. Мы поддерживаем ФИИТ с 2014 года, и это уже лучшее направление обучения для разработчиков в Екатеринбурге. Может и среди многих других городов тоже, но точно не знаем: мы ориентируемся на топовые вузы в столицах.

Прежде всего, мы отказались от советской модели обучения, по которой программу обучения для всех студентов направления определял кто-то умный, знающий, сколько и каких специалистов понадобится плановой экономике СССР через 5 лет. Современный мир слишком быстро меняется, чтобы за ним успевать, образовательные программы должны быть более гибкими. Кроме того, студенты часто добиваются больших успехов в тех областях, которые выбрали себе сами. Поэтому ключевой особенностью нашей программы является свободный выбор студентом траектории своего обучения.

Наш выпускник должен быть способен специализироваться, как ему надо: архитектор, менеджер продукта, специалист по машинному обучению, разработчик игр... Мы не можем научить всему, но «зародыши» нужных специализаций у выпускника должны быть наготове.

Обучение на ФИИТ состоит из двух блоков:

- На 1 и 2 курсе студентов ждёт фиксированный набор обязательных курсов — т.н. «ядерная программа». Это самые полезные и фундаментальные курсы, которые мы посчитали важными вне зависимости от того, кем захочет стать студент. Они дадут основу, после которой уже можно будет выбирать куда и как развиваться дальше. А ещё выходить на стажировки в ИТ-компании.
- На 3 и 4 курсе мы дадим свободу выбора отдельных дисциплин: курсы от ИТ-компаний, от профессоров матмеха УрФУ или Уральского отделения Академии наук, онлайн-курсы других университетов. Студенты смогут самостоятельно определять свою траекторию обучения и нести ответственность за своё развитие.

Прямо сейчас мы активно работаем над блоком обязательных курсов, а блоком курсов по выбору займемся чуть позже, после первого набора студентов. Поэтому в программе ниже не полный список курсов: например, пока не перечислены курсы по другим языкам и технологиям типа Java, C++..., нет некоторых курсов, которые уже читаются на Матмехе. Список будет пополняться по мере того, как мы будем эти курсы создавать и находить/одобрять. Можете написать в комментариях, какие курсы на ваш взгляд важны, и мы обратим на них дополнительное внимание.

1 год		2 год	
Основы программирования 1	6 Основы программирования 2	6 ООП (Проектирование кода)	4 Основы проектирования баз дан
Основы проектной деятельности	3 Python	4 Python	4 Основы веб-разработки
Операционные системы	4 Архитектура ЭВМ	2 Сети и протоколы интернета	3 Основы компьютерной безопасн
Алгебра и геометрия 1	6 Алгебра и геометрия 2	6 Конкурентное программирование	2 Функциональное программирова
Математический анализ 1	6 Математический анализ 2	6 Дискретная математика	4 Алгоритмы и структуры данных
		6 Математический анализ 3	4 Машинное обучение
		Теория вероятностей	4 Математическая статистика
Научное мышление	3 Основы дизайна и композиции	2 Психология и социология	2 Публичные выступления
Эффективные коммуникации	3 Английский язык	4 Английский язык	3 Создание ПО
Английский язык	4	БЖД	2 История
Командное соревнование		Командный проект	

- Синий цвет — связанные с программированием,
- Зелёный — математика и около математические курсы,
- Красный — остальное.

## Ядерная программа

Основные изменения по сравнению с классическими образовательными программами коснутся не названий, а содержания курсов: каждая дисциплина осознанно заточена на подготовку разработчиков, и студенты будут понимать, где и как она пригодится. Этот подход поможет формировать ответственность за собственное развитие: мы хотим, чтобы студенты на 3-4 курсах осознанно подходили к выбору спецкурсов и своей специализации — необходимый фундамент для этого, как я уже говорил, им даст ядерная программа.

### Несколько примеров, как цели, которые мы ставим перед математическими дисциплинами, влияют на содержание

**ЦЕЛЬ:** Максимально приблизиться к программированию

**ПРИМЕР ВЛИЯНИЯ:** На практиках по алгебре вместо ручной прогонки методов надо будет их запрограммировать.

**ЦЕЛЬ:** Выпускнику не требуются года доучивания для того, чтобы увидеть возможность применения того или иного математического аппарата.

**ПРИМЕР:** На лекциях по матанализу будетделено дополнительное внимание обоснованию полезности, то есть где и в каком виде отдельные темы матанализа будут полезны разработчику: оценка сложности алгоритмов, понятийный аппарат для теорвера и оптимизаций, обработки изображений и т.п.

**ЦЕЛЬ:** Выпускник умеет понятно, строго и логично рассуждать.

**ПРИМЕР:** В математических курсах будет так называемая система "листочеков" —

список задач на неделю, решения которых студент должен устно дожожить преподавателю и получить обратную связь, на сколько у него получилось понятно, логично и строго.

**ЦЕЛЬ:** Выпускник понимает язык математики и способен читать, понимать и использовать в своей работе профессиональные и научные статьи.

**ПРИМЕР:** Тут особо менять программы не пришлось, но, например, выделился отдельный двухнедельный курс Основные математические концепции, который расскажет про отображения, группы, кольца, поля, индукцию, бинарные отношения..., чтобы те, кто не знал со школы, быстро подтянулись и могли использовать.

## По программистским дисциплинам мы ставим разные цели

На первом курсе, мы хотим:

- Дать основные концепции программирования и много практики на их освоение: ООП, алгоритмы, структуры данных...
- Дать языковой кругозор и научить осваивать новые языки.
- Провести ликбез про устройство мира: компьютер, операционная система, сети, протоколы интернета.
- Научить работать в командах над проектами. К сожалению, частенько за этими словами кроется профанация, но мы сделаем всё правильно! Подробнее ниже.

Со второго года начинается блок прикладных предметов: нужно было выбрать, какие оставить в обязательной программе, а какие сделать курсами позже. В итоге в программу взяли несколько коротких курсов «Основы чего-то там» по важным темам и запланировали их продолжения с глубоким погружением в курсах по выбору. Если интересно, в комментариях или следующем посте могу рассказать про логику появления каждого предмета.

Тут намеренно нет ни одной хайповой темы: IoT, VR, Blockchain... Обязательная часть программы не про хайп. Все эти модные слова появятся в курсах по выбору в двух последних годах обучения.

## Очень сложно оказалось справиться с гуманитарным блоком

В первую очередь, потому, что это не так интересно, как разбираться с математикой и программированием :) Во вторую — почти все гуманитарные предметы подконтрольны другим институтам, например, английский язык читается специально межинститутской кафедрой.

Мы стараемся найти преподавателей, которые готовы в рамках своих курсов сместить

акценты на воспитание софт-скиллов, которые потребуются разработчикам: вместо расстановки ударений — особенности коммуникации в команде, вместо истории жизни случайного философа — что такое качественная аргументация в споре. К сожалению в названиях предметов в программе эти тонкости не отразить.

Не готов пока заявить, что всё будет супер- круто, но пока нам удаётся договариваться с преподавателями, и мы рассчитываем постепенно делать огненным и гуманитарный блок тоже.

## Проектная и командная работа

Про проектную работу говорят все, кому не лень, каждый первый пытается её построить, и у большинства выходит пшик. Вы наверняка и сами с таким сталкивались. Это страшно дискредитировало проектный подход к образованию в России. При этом известно, что построить проектное образование возможно — например, почитайте о [Школе 42](#).

Не буду останавливаться на пользе — гугл много может про это рассказать без меня. Гораздо интереснее: почему мы думаем, что у нас получится? Вообще, этому можно посвятить отдельный большой пост, но, если кратко, потому что мы сами работаем в продуктовой компании и знаем, как делать проекты и как работать в команде. И представляем, что именно часто вызывает трудности у студентов. Поэтому мы сделаем максимально плавный въезд.

Для начала студенты прослушают курс про командную и проектную работу, который готовят разработчики и менеджеры СКБ Контур: git, треккеры задач, целеполагание, декомпозиция на подзадачи, Agile-практики, планирование работы в команде и вот это вот всё. В рамках этого курса уже в первом семестре студенты выполнят крохотный проект, в каждом следующем семестре проекты будут становиться всё больше и сложнее, пока на третьем курсе не вырастут в индустриальный проект — работу на основе реальной задачи от реальной компании. Индустриальные проекты — это то, что позволяет ещё в рамках учёбы поработать над задачей максимально приближенной к рабочей и ближе познакомиться с ИТ-компанией, предложившей задачу. Индустриальные проекты сейчас стараются делать все ведущие ВУЗы, мы же дополнительно хотим ещё и хорошо подготовить студентов к этому на младших курсах.

За что мы точно не переживаем, так это за Хакатоны, которые тоже есть в программе. Их мы умеем делать круто. Вот, например, [отчёт об одном на Хабре](#). Если нужно больше, полистайте [ленту Контура](#) — там можно ещё и про наши образовательные активности прошлых лет почитать.

## И, наконец, реальная работа

Образование для инженеров-разработчиков не может быть оторвано от практики, поэтому все студенты после второго и третьего курса пройдут стажировки в ИТ-компаниях, и это будет засчитано за часть учебной нагрузки. Мы будем помогать с поисками подходящей компании. Студент сможет сам выбрать себе стажировку, а тем,

кто не справится, мы поможем: организуем ярмарку стажировок и познакомим с ИТ-компаниями города. Технологии науки и образования

# Что особого ждёт ісрсшника на ФИИТ?

Теперь немножко работы с целевой аудиторией codeforces :)

1. Самое очевидное: свободный выбор курсов на 3-4 годах подразумевает, что вы сможете выбрать реально полезные курсы. А у нас студенты смогут повлиять на то, какие курсы вообще появятся в программе.
2. На курсе Основы программирования олимпиадники попадают в группы, которые занимаются усложнёнными задачками с Тимуса под руководством кого-то из тусовки опытных олимпиадников Екб. Курирует это дело Lich\_Sandro.
3. Спортивное программирование легитимизировано, на старших курсах за него можно будет получать зачёты и экзамены.
4. Командное участие в контестах будет засчитываться за учебную практику. Конечно, если вы что-то при этом решаете, а не просто собираетесь 5 часов потусить.
5. Тренеры в УрФУ уже получают оплату за проведение занятий и тренировок. Мы собираемся нанять менеджеров, которые разгрузят тренеров, помогут с организацией тренировок, командировок и что-там-ещё-бывает-нужно. Обычно такие улучшения процессов напрямую влияют на результаты, так что мы рассчитываем получить на выходе более сильные команды.
6. Мы будем засчитывать летние интенсивы с решением кучи тренировок, дорешиваниями и участием в семинарах за учебную практику, а лучшим командам — за стажировку в компании. Спортивное программирование далеко от промышленной разработки, но мы не можем сходу сказать, что окажется более полезно для конкретного разработчика — придётся вам решать самим.

## Ещё не конец

Вы, наверное, заметили, что я много где пишу «скорее всего», «наверное» и подобные обороты будущего времени. Всё дело в том, что мы прямо сейчас находимся посреди процесса разработки этого учебного продукта. Не пугайтесь — так и должно быть. Быстро можно сделать только плохо — хорошее качество требует времени и усилий. Мы не выбиваемся из запланированного графика.

По факту же, окончательного продукта скорее всего не будет никогда — мы собираемся идти к идеалу, пока не закончатся силы, будем постоянно пробовать новое, отбрасывать неработающее и вот это вот всё. В том числе при участии студентов, с которых мы собираемся регулярно собирать обратную связь о происходящем (например, в конце недели у нас встреча со студентами и школьниками

из олимпиадников Екб). Если интересно — присоединяйтесь :)

В посте я постарался охватить только самое важное, и куча нюансов осталась за кадром. Задавайте вопросы — постараюсь на них ответить.

## PS: Почему нет нормального сайта / группы ВК / чего-нибудь ещё?

Если честно, потому что разработка программы занимает всё время, и мы не успели нанять пиарщика-сммщика-вот-это-вот-всё, который поставит рекламную работу на поток.