

Первые итоги внедрения гибкой системы обучения и выделение популярных траекторий обучения на базовой кафедре “Корпоративные информационные системы” факультета инноваций и высоких технологий МФТИ(НИУ)



Старичков Никита

Базовая кафедра “Корпоративные информационные системы”

Работает на факультете инноваций и высоких технологий Московского физико-технического института (НИУ)

Базовая организация кафедры - Фирма 1С





Гибкая система обучения / Архангельск 201

- Свобода в выборе учебных курсов
- Всего около 55 курсов на выбор
- Курсы разбиты на 5 уровней сложности
- Выбор курсов - в начале каждого семестра
- У курсов есть зависимости - студент может записаться на курс, только если закрыл все связанные курсы меньшего уровня

Любой курс можно сдать экстерном:

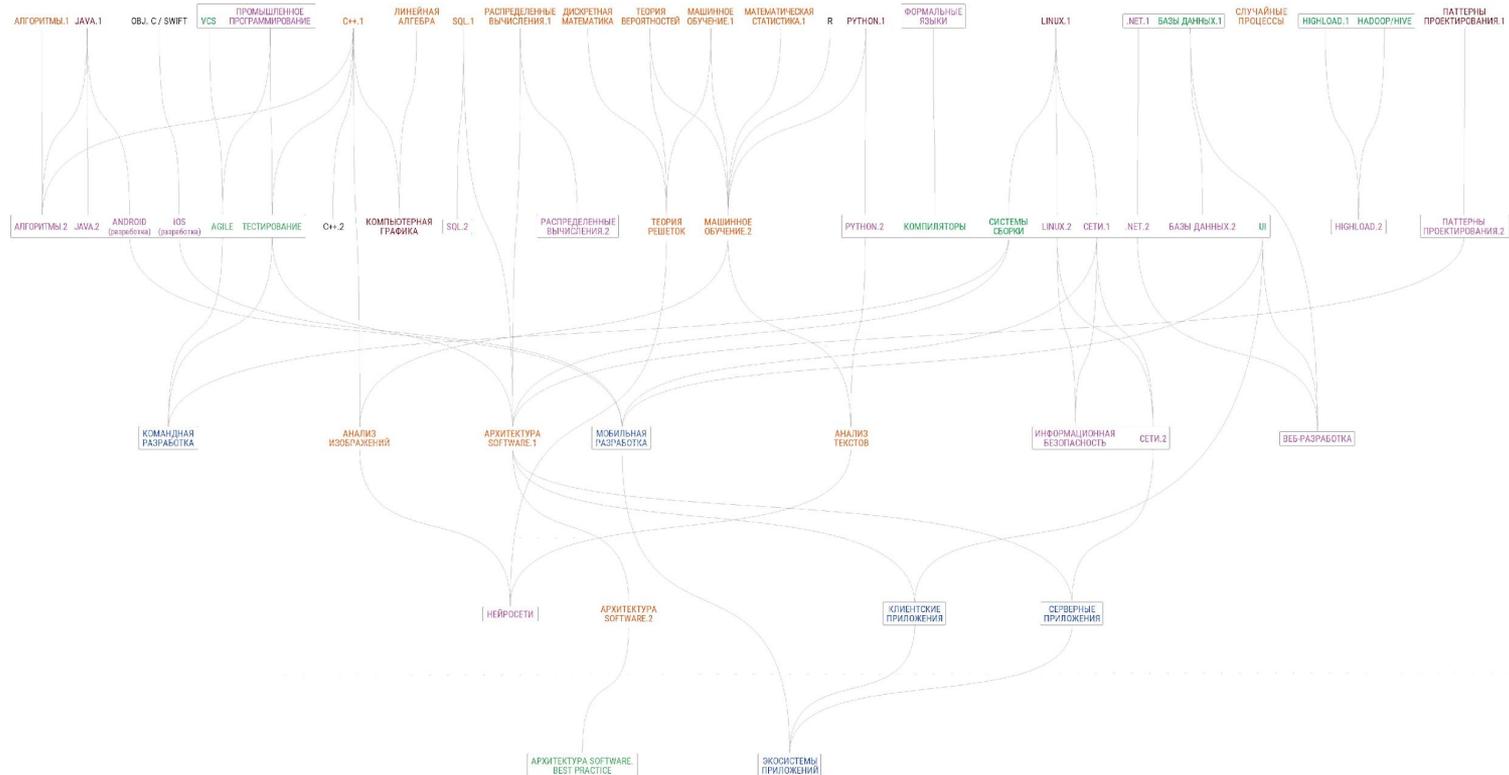
- Экзамен (в форме собеседования)
- Выполнение практического проекта
- Перезачет другого курса

Требования от кафедры:

- Для бакалавриата - 3 закрытых курса третьего уровня
- Для магистратуры - 1 закрытый курс пятого уровня и 5 курсов третьего уровня



Граф курсов гибкой системы обучения





Курсы кафедры [в виде таблицы]

Уровень 1	Формат	Уровень 2	Формат	Уровень 3	Формат	Уровень 4	Формат	Уровень 5	Формат
(3) Java.1	Ф/СР	(26) Алгоритмы.2 [1, 2/3]	К	(48) Мобильная разработка.1 [19, 36/37]	Практика	(55) Нейросети [51, 52, 45]	К	(59) Экосистемы приложений [48/57, 58]	Практика
(7) Linux.1	К	(27) C++.2 [2]	СР	(49) Инф. безопасность [43]	К	(56) Архитектура.2 [53]	К	(60) Архитектура. Best Practice [56]	3
(14) .NET.1	К	(28) Java.2 [3]	К	(50) Сети [43]	К	(57) Клиентские приложения [53, 19]	Практика		
(16) Objective C / Swift	СР	(31) Машинка.2 [6, 11, 12, 24/25]	Ф к./в.	(51) Анализ изображений [31, 2]	Ф к./в.	(58) Серверные приложения [53, 50]	Практика		
(18) VCS	3	(32) Комп. графика [2, 10]	Ф/СР	(52) Анализ текстов [31, 44]	Ф к./в.				
(19) Создание UI	3	(34) Компиляторы [15]	3	(53) Архитектура.1 [47, 39, 40, 5/22]	К				
(20) Паттерны пр-ия.1	К	(35) .NET.2 [14]	К	(54) Командная разработка [46, 39, 40]	Практика				
(21) Базы данных.1	К	(36) Android [3]	Ф/СР	(61) Веб-разработка [19, 21, 35]	Практика				
(22) Highload.1	3	(37) iOS [16]	Ф/СР						
(25) R	СР	(39) Системы сборки [7]	3						
		(40) Тестирование [2, 17]	3						
		(41) Базы данных.2 [21]	К						
		(42) Highload.2 [22]	К						
		(43) Linux.2 [7]	К						
		(44) Python [62]	К						
		(45) Теория решеток [6, 8, 11]	Ф к./в.						
		(46) Agile [17, 18]	3						
		(47) Паттерны пр-ия.2 [20]	К						
		(17) Промышленное пр. [18, 20]	К						



Внедрение системы

- Гибкая система обучения была внедрена, начиная с набора 2016 года
- Первый набор - 16 человек
- Сейчас эти ребята уже учатся на 5-ом курсе (1-ый курс магистратуры)
- Суммарно на 3,4,5 курсе сейчас - 84 человека

Набор 2019 года:

Около 145 студентов на факультете, 125 пришли к нам на контрольную, 85 - мы позвали на собеседования. Получили приглашения на кафедру: **67**



Популярные траектории обучения

Несмотря на полную свободу студентов, можно выделить курсы, за которые они голосуют “ногами”. Видимо, это зависит от:

- Тематики курса
- Преподавателя и его личной харизмы, авторитета, знаний, отношения к работе
- Проработанности материалов и заданий по курсу

(критерии выше - это то, что называют студенты)



Популярные траектории обучения

Траектория “Проектирование”

Курсы:

- Паттерны проектирования
- Метaprogramмирование
- Архитектура ПО

На курсы этой траектории весной 2019г. ходит 30 человек (около 36% студентов)

Траектория “Классика разработки”

Курсы:

- Технологии программирования
- Linux
- Сети

На эти курсы ходит 28 человек (33%)

Практики:

- Серверные и клиентские приложения
- Мобильная разработка
- Веб-разработка

Практики сейчас проходят 17 человек (20%)



Траектория “Проектирование”

Курс “Паттерны проектирования”

Двухсеместровый курс о паттернах проектирования. Рассматриваются порождающие, структурные и поведенческие паттерны. В рамках курса студенты выполняют большое число практических заданий.

Курс “Метапрограммирование”

Курс читается один семестр. Рассматриваются вопросы, связанные с программированием на C++ с использованием шаблонов: списки типов, генерация иерархий, обобщенные функторы и множественная диспетчеризация.



Траектория “Проектирование”

Курс “Архитектура ПО. Часть 1”

В курсе рассматриваются критерии хорошей и плохой архитектуры, базовые принципы проектирования программных систем, различные технологические типы архитектуры.

Курс “Архитектура ПО. Часть 2”

Практический курс, в рамках которого студенты отрабатывают навыки, полученные на предыдущих курсах. Перед студентами ставится задача разработать архитектуру большой системы с детализацией вплоть до диаграммы классов.



Траектория “Классика разработки”

Мини-курсы “Технологии программирования”

Набор занятий по следующим темам: системы контроля версий, системы сборки, инструменты отладки и тестирования ПО, инструменты организации процесса разработки.

Курс “Linux”

Читается два семестра. Курс начинается с базовых понятий и инструментов семейства ОС Linux (bash, учетные записи пользователей, файловые системы, система X, загрузка системы, POSIX и pthreads, сигналы и др.) и переходит к продвинутым (устройство ядра, аппаратная и программная виртуализация, прерывания, синхронизация кода ядра, управление памятью и др.)



Траектория “Классика разработки”

Курс “Сети”

Дисциплина, посвященная программированию сетевого взаимодействия. Читается введение в модель OSI, далее подробно рассматривается каждый из уровней: канальный(включая Ethernet, Bluetooth, RFID и др.), сетевой, транспортный, прикладной, а также основные понятия и технологии обеспечения безопасности.

Практики “Серверные приложения”, “Клиентские приложения”, “Мобильная разработка”, “Веб-разработка”

В рамках каждой из этих практик студенты разрабатывают законченный программный продукт, а основной метрикой качества является оценка с точки зрения пользователя.



Выводы

Стоит отметить, что само наличие популярных траекторий обучения не говорит о том, что в гибкой системе обучения нет практического смысла. Несмотря на то, что большинство студентов следуют по популярным траекториям, некоторые выбирают отличающиеся индивидуальные траектории.



Выводы

Ключевое - гибкая система обучения в целом не требует кратного увеличения вкладываемых ресурсов по сравнению с жесткой системой.

В текущем семестре кафедра читает 12 дисциплин для студентов 3,4,5 курсов. По жесткой системе обучения читалось бы 8 дисциплин (2 - для 3-его курса, по 3 - для 4-ого и 5-ого курсов).

Т.е. затраты выросли на 50%.

*“популярные учебные траектории” - “жесткие учебные планы”
“курсы для отдельных студентов” - “факультативы”*



Выводы и дальнейшее развитие

Гибкая система обучения доказала свою работоспособность и привлекательность для студентов.

Дополнительно это подтверждается тем, что со следующего учебного года (2019-2020) аналогичная система будет внедрена для продвинутого потока магистратуры факультета инноваций и высоких технологий МФТИ(НИУ).



Спасибо за внимание!

Старичков Никита

stan@1c.ru