

Оглавление

Учебная практика не на семестр или на год – мы просто раз в семестр отчитываемся о прогрессе работы

- Научно-исследовательская или программно-инженерная работа:
 - Решение более-менее сложной практической либо научной задачи
 - Отчёт (текст)
 - Код (очень желательно)
- По формату близка к научной статье и выступлению на конференции
- Тема должна быть интересна той кафедре, на которой планируем защищаться

Каждый год делают приложение для заказа еды в Яме

Никто (кроме кафедры СП) не против доказательства теоремы в качестве практики

Если планируете погрузиться в большую тему, то лучше в первом (третьем) семестре прочитать 80 источников и написать обзорную статью

Текст пишем в процессе работы и периодически показываем научнику

Хорошие работы желательно опубликовать на весенних конференциях

Обязательно сдать отчёт и отзыв научника

На кафедре СП просят ещё много чего

Отзыв научника должен быть подписан

Если научник вам не отвечает, значит он хочет отправить вас на пересдачу

Им проще отчислить студента без практики, чем покрывать его перед гос. комиссией

Определение 1. Консультант – ставит задачу, читает и ревьюит код, помогает с техническими проблемами (е. г. Яков)

Определение 2. Научный руководитель – преподаватель (обязательно), отвечает за адекватность задачи, следит за методическими вопросами, следит за ходом работы, помогает с текстом и подготовкой к защите. Пишет отзыв, так что поддерживаем хорошие отношения (е. г. Чернышёв)

Определение 3. Руководитель практики – общая организация процесса, следит, чтобы у каждого была тема и научник, поможет найти научника, если тот потерялся (е. г. Литвинов)

Определение 4. Комиссия – преподаватели кафедры, представитель индустрии (обычно сидит кто-нибудь типа Кирилла):

- Все защиты всегда с комиссией
- Защищаетесь в комиссии той кафедры, с которой научник

На комиссию ходят те преподы, которые хотят самоутвердиться. Над вами **будут** смеяться (“О боже, это что, дефис вместо тире в презентации?”)

Где брать темы:

- Есть сайт кафедры СП, куда всем кафедрам рекомендуют выкладывать темы. Однако, пока что, все это игнорируют (в том числе, СП). Бывает на бумажке на двери кафедры
- У преподавателя по программированию
- На стажке
- Если были в ЛШ – продолжить начатое

Определиться с выбором темы надо **до конца сентября**

Если планируете где-то работать (или стажироваться) – узнавайте, можно ли у них что-то защитить (иначе бегите оттуда. Не вытянете)

На матмехе никто не занимается юнити и питоном – это не для матмеха

Если пишете игру на юнити – пишите, но не рассчитывайте на помощь научника

- Сентябрь – определиться с темой
- Сентябрь – начало декабря – работа над практикой:
 - Быстрый мини-обзор
 - Введение, постановка задачи, научиться убеждать окружающих в актуальности темы
 - Обзор
 - Проектирование
 - Реализация (если предполагалась)
 - Апробация/эксперименты (если предполагались)
 - Написание текста
- Конец декабря – защиты

Пишете игру на юнити – докажите, что в неё будет кто-то играть

Пишете декомпилятор – извольте показать на большом количестве примеров, что оно что-то декомпилирует

Апробация – проверка, что программа делает то, что от неё хотят **пользователи** (это не тесты)

Кафедра СП ожидает, что хотя бы каждый второй семестр будет код. Остальные разрешают чисто теоретические работы. Минимум раз в неделю отчитываться научнику о ходе работы – используем [сайт кафедры СП](#)

Часы работы в присутствии преподавателя – это для тех научных, кто предпочитет общаться лично. Остальным это **не надо**

На сайте кафедры СП есть раздел “Студентам” – там все работы, которые успешно защитились на кафедре [Шаблон текста](#)

[Чеклист по оформлению репозитория](#) – **обязательно** всё должно быть выполнено, иначе засмеют на защите

[Чеклист по презентации](#)

В тимс Литвинов будет писать организационные моменты – что-то пропустил – пересдача

Кинут табличку, надо будет записаться

Ограничений на команду нет, но защита **по-отдельности** (можно попроситься по порядку). Отчёты должны быть полностью разные – иначе может показаться, что кто-то пытается вкатиться в IT на плечах своего товарища

Структура отчёта:

- Титульный лист
- Оглавление
- Введение в предметную область, постановка задачи
- Обзор литературы существующих решений
- Описание предлагаемого решения (архитектура, реализация)
- Апробация/эксперименты
- Заключение
- Список литературы

30-40 страниц не пишем (можно только если 20 из них – картинки)

На ютуб ссылаться не стыдно, на втором курсе так точно

Введение:

- Известная информация, “Background”

- Неизвестная информация, “Gap”:
 - Актуальность темы
 - Практическая значимость
 - Кому конкретно это надо. Обязательно надо понимать, **зачем** ему это надо
- Кратко про наш подход к решению и почему он приведёт к цели (“Гипотеза” и “Подход”)

Не пишите, что “мы хотим написать выживач на юнити, выйти в топы стима и заработать миллионы денег” – не поверят

Лучше сказать, что это для портфолио – все поразятся вашей сознательности

Постановка задачи:

- Цель работы:
 - Одним предложением – что конкретно надо сделать
- Задачи:
 - Отчуждаемые – их можно от нас отнять (изучить юнити – не задача. Надо говорить “сделать обзор”)
 - Специфичные – если бы я хотел проектировать микроволновку, какие бы у меня были задачи? Если они совпали с вашими - пересдача
 - Решение которых приведёт к цели (смотрим нашу формулировку цели)

“Удобное”, “гибкое”, “быстрое” не пишем – это нельзя измерить

Обзор:

- Обзор существующих решений:
 - Цель обзора, критерии отбора материалов
 - Критерии сравнения
 - Таблица с результатами
 - Выводы
- Обзор используемых чужих результатов:
 - Всё, написанное и придуманное не нами – в обзор
- Должен соотноситься с темой работы

В обзоре подробно описываем, что было сделано в (e. g. Desbordante) до нас

Если потребуется какая-то нестандартная библиотека – пишем “Нужна библиотека под такую задачу, есть такие три, выбрали эту, потому что...”

Выбор языка тоже хорошо бы аргументировать

Описание решения:

- Желательно, чтобы разделы соответствовали списку задач
- Аргументированное обоснование принятых решений и отказа от альтернатив
- Выбор инструментария
- Описание архитектуры, алгоритмов, ... – умеете UML диаграммы – рисуйте, не умеете – рисуйте хоть что-то

“Я взяла эту библиотеку, потому что она первая попала под руку” – сразу минус. Любое сложное решение (думали больше пяти минут) аргументируем

Если работаете в (e. g. Desbordante), рисуем целиком архитектуру проекта и выделяем цветом, что мы делали

Описание решения (продолжение):

- Рисунки и диаграммы:

- Лучше использовать UML – он стандартный
- Подписи (рисунок 1: то-то то-то)
- * Чужие рисунки – со ссылкой на источник – иначе плагиат

Пишем текст так, как будто будем его печатать (чтоб рисунки было видно, ссылки – настоящими ссылками)

Апробация:

- Доказать, почему всё, что мы делали, вообще осмысленно
- Апробация – внешняя “оценка” работы
 - Отзывы пользователей, лучше количественные (оценка числом)
 - Внедрение, релиз, оценки
- Эксперименты - численное доказательство, что ваш результат лучше аналогов:
 - Замеры производительности, точности, ...
 - Отдельная большая наука, делаем аккуратно
- Если до апробации не дошли – опишите продуманный план апробации и экспериментов

Выложили в гугл плей, приложили скриншот с оценкой “4.9” – апробация

Выступление на конференции (если вас не освистали) – апробация

Хороший эксперимент содержит “угрозу валидности” – почему результат может быть некорректным (например, “опрашивали друзей, которые не хотят нас расстроить”)

На матмехе принято использовать **system usability scale**

Результатом эксперимента являются параметры случайной величины – мат. ожидание и дисперсия

Среднее по больнице (без дисперсии) мерить смысла нет

Заключение:

- Перечисление результатов, выносимых на защиту
- Должно быть согласовано с постановкой задачи (вплоть до полного её повторения)

Пример. Выбрать библиотеку – выбрана библиотека **такая-то**

- Должно быть согласовано с текстом (никаких результатов из ниоткуда)
- Если практика предполагает предположение, реалистичные планы на дальнейшую работу

Пример. В весеннем семестре оптимизировать такие-то компоненты системы

- Ссылка на репозиторий и свой ник (чтобы можно было фильтровать по коммитам)

Если в постановке задачи есть пункты, которых нет в заключении – сразу передача

Литература:

- Ссылок примерно столько же, сколько страниц в работе
- Обязательно на каждый пункт ссылаться из текста
- Лучше ссылаться на научные статьи
 - Ещё лучше – на книги, но по предметной области
 - Смотрим на индекс Хирша и число цитирования
- Реально прочитанные работы
- ГОСТ Р 7.0.5-2008
 - Так, как это делает BibTeX
 - Сам ГОСТ имеет порядка 25 страниц примеров вообще для всего, лучше его прочитать
 - Ссылки генерятся командой `\cite`
- В литературу только литературу:

- Ссылки подстраничными сносками (\footnote)
- В литературу можно только с датой обращения (иначе сайт может поменяться)

Никогда не публикуйтесь в журналах за деньги (где вы платите за публикацию)

За плагиат – отчисление без права восстановления (даже если пробелы заменить пробелы на какие-то пробельные символы)

Копипаст даже одного предложения без указания источника – незачёт (определения не копируем. Они нам не нужны)

Нейросети не стоит подключать для написания текста (для улучшения готового текста – можно)

Стоит порепетировать выступление

Из презентации должно быть предельно понятно, что и зачем мы делаем и при чём тут наша кафедра

Как антиплагиат используется blackboard – он репортит, что у вас страница с оглавлением называется “Оглавление”, так же как у других

Лучше научник над вами посмеётся, чем комиссия

Озаботьтесь получением отзывов заранее – на сессии у научника тоже сессия

Обязательно CI, юнит-тесты, README, лицензия

Дипломы пишем самостоятельно – товарищ может подставить

Выступление на конференции за защиту **не** засчитывается (там нет комиссии)

Тему и научника можно менять когда угодно, только надо предупредить Литвинова

Если вам сказали в Яндексе “мы вам практику зачли, передайте научнику” – это так не работает