

# О больших проектах на примере курсовой работы по компьютерной графике

Кострицкий А. С., Куров А. В.

Москва — 3 июля 2024 г.

## Содержание

<b>1</b>	<b>Процесс и этапы</b>	<b>1</b>
<b>2</b>	<b>Этапы разработки</b>	<b>4</b>
2.1	Первоначальная декомпозиция . . . . .	4
2.2	Этап анализа . . . . .	4
2.3	Этап проектирования . . . . .	5
2.4	Технологический этап . . . . .	5
2.5	Проведение исследования . . . . .	7
2.6	Подготовка к защите и защита . . . . .	9
2.6.1	Расчётно-пояснительная записка . . . . .	9
2.6.2	Презентация результатов . . . . .	11
2.6.3	Средства оформления . . . . .	11
<b>3</b>	<b>Как написать для руководителя черновик задания</b>	<b>12</b>
3.1	Некоторые общие правила . . . . .	12
3.2	Алгоритм . . . . .	16
3.3	Самые частые ошибки . . . . .	17

## 1 Процесс и этапы

Курсовая работа отличается от лабораторной или домашнего задания не только наличием финальной документации, но и сроками – на курсовую отводятся 14 недель, в течении которых несколько раз студентом вместе с руководителем принимаются различные решения по дальнейшему выполнению.

Модель выполнения курсовой работы более всего похожа на каскадную модель разработки программного обеспечения, которая представлена на рисунках 1 и 2.

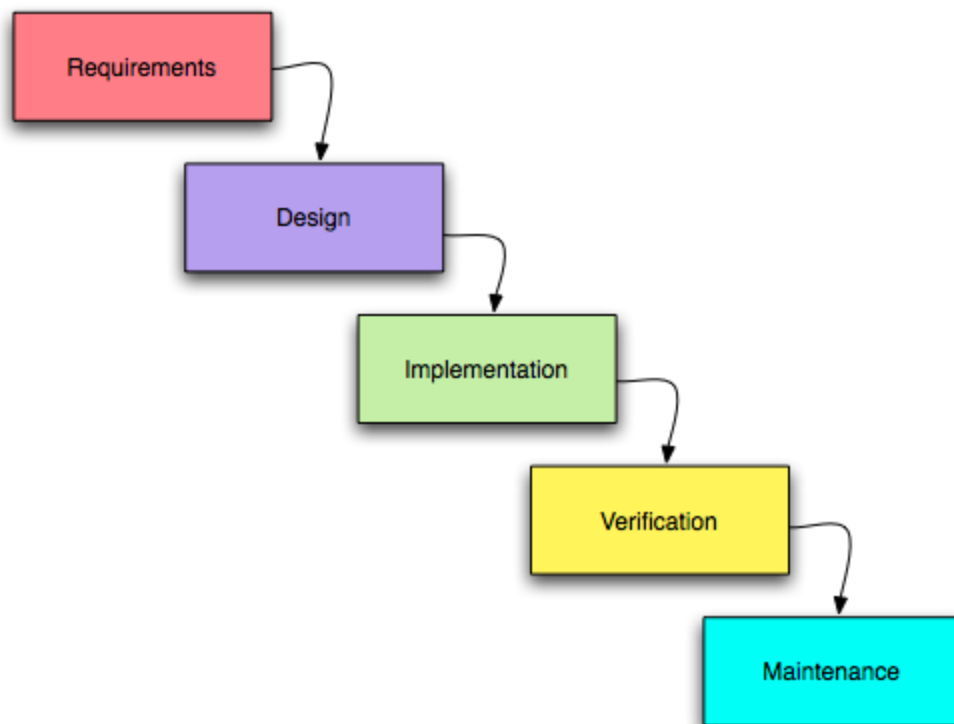


Рис. 1: Каскадная модель разработки ПО, предложенная У. У. Ройсом.

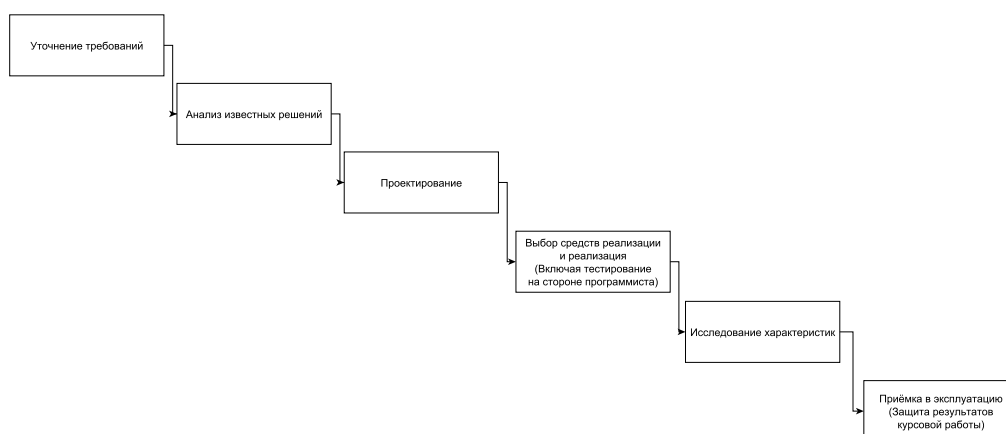


Рис. 2: Рекомендованная модель выполнения курсовой работы.

Процесс разделён на этапы таким образом не просто так:

- Каждый из этапов соответствует разделу РПЗ и нескольким слайдам презентации.
- Каждый из этапов имеет вполне конкретный результат, который ожидается комиссией зачастую во вполне определённом виде на формальном языке.

Результатом курсовой работы является разработанное программное обеспечение (ПО), которое состоит из самой программы и документации к ней. К документации предъявляются требования как по наполнению – главой курса, так и к оформлению – кафедрой.

На защиту представляются:

1. Расчётно-пояснительная записка (РПЗ) к курсовой работе.
2. Презентация результатов.
3. Реализованное по документации ПО.

Рассмотрим каждый из этапов, отмечая, в каком виде и в какой части документации должны быть отражены результаты.



Рис. 3: Рекомендованная модель выполнения курсовой работы в виде диаграммы Ганта. Единственное мероприятие на уровне кафедры – защита курсовой работы на 15-17 неделях и последующее закрытие ведомости, все остальные сроки обговариваются персонально с преподавателем.

## 2 Этапы разработки

### 2.1 Первоначальная декомпозиция

До начала этапа анализа требуется определиться с целью и задачами работы, которые описываются во введении РПЗ.

Цель работы:

1. Может быть только одна.
2. Формулируется всегда в виде существительного, но не глагола.
3. Формулируется на основе задания на курсовую работу.

**Хуже** "Цель работы – **разработать** программное обеспечение,..."

**Хуже** "Целью работы является **создание** программного обеспечения,..."

**Лучше** "Цель работы – разработка программного обеспечения,..."

**Лучше** "Целью данной работы является разработка программного обеспечения,..."

**Лучше** "Цель работы – разработка программного обеспечения для создания реалистичного изображения мыльных пузырей..."

Далее во введении указывается, что «для достижения поставленной цели требуется решить следующие задачи», которые и перечисляются. Задач должно быть не менее четырёх, причём на каждом из дальнейших этапов разработки решается не менее одной.

### 2.2 Этап анализа

На этапе анализа требуется описать предметную область; методы и алгоритмы решения основных задач компьютерной графики: построения теней, отражений, лучей, обработки трёхмерных моделей, etc. Алгоритмы на данном этапе можно описывать в достаточно общем виде, указывая со ссылкой на источник свойства и требования к входным данным. К выбранным алгоритмам словесное описание или схемы нужно будет давать уже на этапе проектирования. Крайне желательно продумать и ввести единую систему обозначений уже сейчас, если она планируется. Требуется описать допустимые диапазоны и ограничения, накладываемые на входные данные.

Для всех методов и алгоритмов решения подзадач графики составляются количественные критерии сравнения, приводится сравнительная таблица, из которой следует обоснование выбора алгоритмов для дальнейшей реализации.

После завершения этапа анализа:

**В РПЗ** добавляется **аналитический раздел** с описанием предметной области, основных задач компьютерной графики, имеющих отношение к курсовой работе, известных алгоритмов решения этих задач и их свойств в виде сравнительной таблицы.

**В выводах к аналитическому разделу** указывается, что «были выбраны алгоритмы. . . »

**В презентацию** добавляется, как минимум, слайд с постановкой задачи, например, в виде IDEF0-диаграммы верхнего уровня, и слайд(ы) с таблицей(ами) сравнения алгоритмов по определённым критериям.

О том, как правильно аргументировать свою позицию, и о том, почему критерии обязаны быть количественными, будет рассказано в рамках НИР осеннего семестра.

## 2.3 Этап проектирования

Результатом этапа проектирования служит разработанная функциональная модель (IDEF0) программного обеспечения или «большая» схема алгоритма составления одного кадра. Выбираются соответствующие структуры данных.

После завершения этапа проектирования:

**В РПЗ** добавляется **конструкторский раздел** с формальным описанием декомпозиции разрабатываемого ПО, например, в нотации IDEF0; с формальным описанием всех используемых или модифицированных алгоритмов единообразно в форме схем алгоритмов или по пунктам.

**В выводах к конструкторскому разделу** указывается, что «было спроектировано программное обеспечение. . . »

**В презентацию** добавляется слайд с декомпозицией ПО и слайды с описанием основных алгоритмов.

## 2.4 Технологический этап

В рамках технологического этапа требуется выбрать средства реализации ПО, спроектированного по ходу курсовой работы, и показать практическую реализуемость – создать это самое ПО.

После завершения технологического этапа:

**В РПЗ** добавляется **технологический раздел** с обоснованием выбора средств разработки, отдельными примерами кода, описанием особенностей реализации, интерфейса пользователя, сценариев тестирования.

**В выводах к технологическому разделу** указывается, что «было реализовано программное обеспечение. . . »

**В презентацию** добавляется, как минимум, один слайд с указанием выбранных средств разработки и структурой разработанного ПО.

**ПО** к курсовой работе должно быть разработано, протестировано и готово к проведению исследования.

Самая частая и самая опасная ошибка студента на данном этапе заключается в попытке под уже выбранный из симпатии язык изобретать требования и подгонять обоснование. Честность – лучшая политика.

В ситуации, когда Вы заранее знаете, что хотите написать свою программу, например, на Си++, можно показать достаточность выбираемых средств реализации.

**Хуже** *«В качестве языка программирования был выбран Си++, потому что он объектно-ориентированный.»* – Язык Си++ мультипарадигмальный. Если имелось в виду объектно-ориентированное подмножество языка, следовало так и написать.

**Хуже** *«В качестве языка программирования был выбран Си++, потому что он имеет статическую типизацию, что уменьшает количество ошибок.»* – Тезис дан без обоснования.

**Хуже** *«В качестве языка программирования был выбран Python, потому что имеется большое число библиотек, созданных сообществом.»* – Тезис дан без обоснования; даже если предположить, что тезис верен, должна быть в дальнейшем показана его значимость (продемонстрировано в курсовой использование упомянутого большого числа библиотек).

**Лучше** *«В качестве языка программирования был выбран Си++ в силу следующих причин:*

- *В стандартной библиотеке языка присутствует поддержка всех структур данных, выбранных по результатам проектирования.* – Показано, что встроенных средств языка достаточно.
- *Средствами языка можно реализовать все алгоритмы, выбранные в результате проектирования.»* – Показано, что встроенных средств языка достаточно.

## 2.5 Проведение исследования

Не рекомендуется употреблять термин «эксперимент» в РПЗ, потому что существует отдельный ГОСТ, регулирующий проведение эксперимента. Члены комиссии могут потребовать представить план эксперимента и указать тип эксперимента. Рекомендуется употреблять слово «исследование».

К моменту проведения исследования разработанное программное обеспечение уже чаще всего реализовано, протестировано простейшими методами и рассматривается исключительно как чёрный ящик.

Типовой алгоритм проведения исследования:

1. Выбираются некоторые варьируемые входные параметры, остальные выбираются фиксированными ИЛИ создаются несколько наборов фиксированных параметров.
2. Выбираются анализируемые выходные параметры.
3. Составляется сценарий исследования.
4. Проводится исследование.
5. Проводится интерпретация полученных результатов исследования – построение графиков, статистический анализ, etc.

Рассмотрим на примере. Пусть есть ПО, контекстная диаграмма которого изображена на рисунке 4.

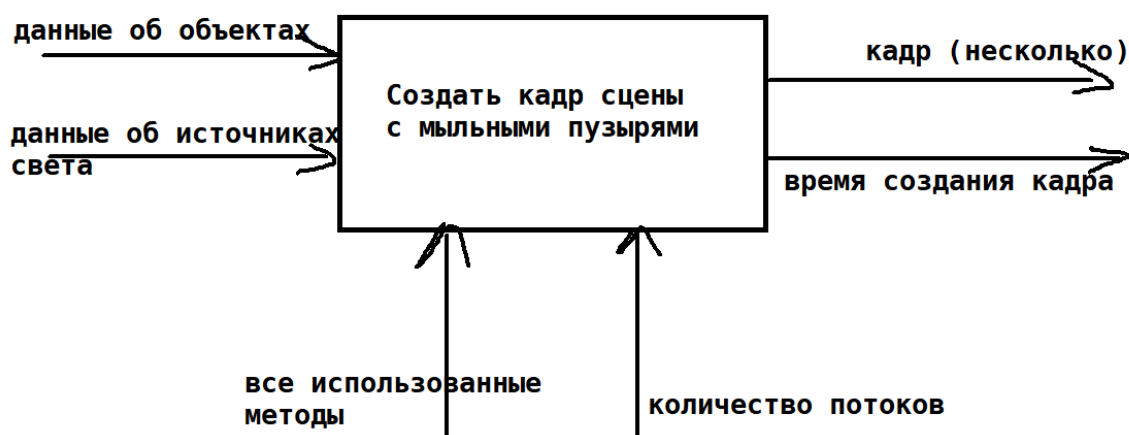


Рис. 4: Пример контекстной диаграммы разработанного ПО.

Допишем к входным и выходным данным некоторые их типовые характеристики. Результат изображён на рисунке 5.

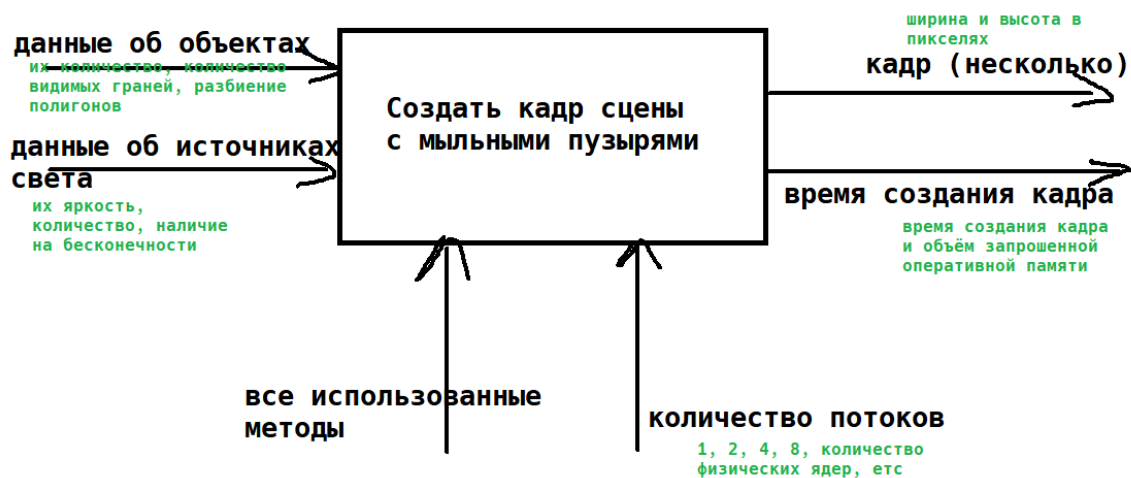


Рис. 5:

В данном случае можно провести, например, одно из следующих исследований

- Поиск зависимости скорости генерации кадра от количества пикселей при наличии 1, 2, 3 объектов на сцене.
- Поиск зависимости скорости генерации кадра от количества используемых потоков при фиксированной сцене из трёх мыльных пузырей, расположенных друг за другом.
- Поиск зависимости скорости генерации нескольких кадров при использовании информации о тенях с предыдущего кадра.



## 2.6 Подготовка к защите и защита

В заключении РПЗ формально указывается, помимо прочего, что «все поставленные задачи были решены» и «цель работы была достигнута».

### 2.6.1 Расчётно-пояснительная записка

РПЗ к курсовой работе должна быть оформлена в соответствии с правилами МГТУ, представленным в виде приложения к положению о порядке подготовки и защиты выпускной квалификационной работы. В случае конфликта правил МГТУ и ГОСТ 7.32-2017 следует подчиняться правилам МГТУ.

Позиция кафедры по шаблонам и примерам оформления, которые Вы можете найти в сети, проста: кафедра не берёт на себя ответственность за соответствие любых найденных Вами шаблонов всем нормам МГТУ. Да, в большинстве случаев эти шаблоны могут упростить написание РПЗ, но некоторые правила могут обновляться. Иначе говоря, нельзя аргументировать что-либо тем, что Вы используете «готовый шаблон» – у кафедры нет в данный момент готовых шаблонов.

Этот документ, который Вы читаете, не оформлен согласно правилам МГТУ. В работах прошлых лет могут быть ошибки.

К замечаниям программы `vkcr.exe` следует относиться с большой аккуратностью и достаточным скепсисом.

Крайне желательно учесть следующие ситуации, которые не могут быть проверены в автоматическом режиме:

1. **Употребление буквы «Ё».** Допускаются варианты: без «Ё», везде «Ё», «Ё» по необходимости.
2. **Висячие строки.** Висячие строки при вёрстке по ширине допустимыми не являются, но система обычно знает это – если у Вас якобы возникла висячая последняя строка абзаца, проверьте, считает ли система её последней строкой абзаца. Пример такой висячей строки:

«Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину.»

Как должно быть:

«Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину. Медведь ел малину.»

3. **Расстановка неразрывных пробелов.** Неразрывный пробел должен ставиться в местах, в которых нужно заставить редактор при вёрстке текста по ширине или при переносе поместить слова заведомо рядом. Неразрывный пробел в Word поставить можно через выпадающее меню

«Вставка→Специальные символы→Неразрывный пробел», В LaTeX он обозначается символом «~».

Примеры: «т. д.», «т. е.», пробел между символом перед ссылкой на формулу и самой формулой в скобках, пробел между символом перед ссылкой на пункт библиографии и самой ссылкой, пробелы при упоминании тезиса Чёрча — Тьюринга — Дойча.

4. **Единообразное употребление точки или запятой в качестве разделителя дробной и целой частей по тексту.** Не существует ГОСТ, устанавливающего единственно правильный символ (не считая ГОСТ по строительной документации [ГОСТ Р 21.1101-2013, ГОСТ 21.508-93] – в них речь идёт про указание размеров, допускается только запятая). Не допускается использование одновременно в качестве разделителя и точки, и запятой. Всё это не относится к листингам исходного кода – в коде всегда указывается тот разделитель, с которым возможна успешная трансляция. Если хотите использовать запятую в LaTeX в качестве разделителя, то придётся обособлять её скобками, чтобы система не ставила сама после запятой минимальный пробел: «0\{,}1234».
5. **Кавычки.** В русской литературе принято ставить французские кавычки-ёлочки, «», а не кавычки-лапки, “”. В LaTeX кавычки-ёлочки выставляются с помощью обычных символов меньше/больше: <<Я в кавычках!>> Если необходимо по тексту, например, при цитировании, поставить кавычки-лапки, то учтите, что в LaTeX закрывающая кавычка уничтожает пробел после себя – в таких случаях обычно скормливают системе пустой символ: «\“текст\”\{ }».
6. **Стилистическая целостность.** После копирования из текстов статей всегда необходимо вычитывать текст: недопустимо употреблять в одном тексте слова и словосочетания «перцептрон» и «персептрон», «пиксел» и «пиксель», «8 байтов» и «8 байт»; если, конечно, это не оформлено как прямая речь.
7. **Схемы и графики только в векторе.** Если картинка заимствована, нужно рассмотреть возможность перерисовать её с учётом стиля своей работы, но это обсуждается в каждом отдельном случае. Ramus, draw.io, yEd, gnuplot, matplotlib позволяют экспортировать картинки в векторном формате – либо svg, либо pdf.
8. **Знаки препинания в формулах.** Формулы и любые другие выносные элементы являются частями текста, они выносятся лишь для удобства чтения. Поэтому в конце формулы в большинстве случаев должна стоять запятая или точка.

9. **Оформление формул.** Формулы запрещено вставлять как изображения, все формулы должны быть набраны в редакторе формул, если Вы работаете в Word. Одно обозначение  $n$  в строке это тоже формула.
10. **Списки.** Единственный допустимый символ при обозначении элементов нумерованного списка это длинное тире: «—». При описании элементов нумерованного списка учтите, что любой текст после точки должен начинаться с большой буквы.
11. **Подписи к выносным элементам.** Подписи к таблицам и рисункам выставляются в разных местах, подробности в приложении к положению о нормоконтроле.
12. **Счётчики.** Значения всех счётчиков должны формироваться автоматически – нумерация рисунков, таблиц, разделов, глав должна быть настроена соответствующим образом.

Оформление РПЗ досконально не проверяется на защите, но за любые найденные ошибки итоговая оценка может быть снижена.

## 2.6.2 Презентация результатов

Презентация к курсовой работе не имеет регулирующего ГОСТ или отдельных правил МГТУ, но должна быть оформлена в академическом стиле.

На титульном листе должна быть отражена информация о дисциплине, теме курсовой работы, группе и полных ФИО автора, научном руководителе. Должны быть указаны город и год.

Герб МГТУ лучше на титульный лист не помещать, потому что с недавних пор появились правила по его цветовому оформлению и размещению как элемента.

Подробности осенью.

## 2.6.3 Средства оформления

Много лучше использовать специализированные средства для создания схем алгоритмов, диаграмм в различных нотациях, etc. Время, потраченное на выравнивание прямоугольников на схеме, в счёт трудозатрат на курсовую работу не идёт.

1. **yEd** намного лучше draw.io в некоторых аспектах, потому что программа знает и распознаёт некоторые виды диаграмм, а значит и размещает элементы как полагается в соответствующих нотациях.
2. **PlantUML**

## 3 Как написать для руководителя черновик задания

Задание на курсовую работу состоит в большинстве случаев из одного абзаца и начинается с указания *разработать программное обеспечение* или *программу*. В этом первом указании уже зашифрованы и цель, и задачи, о которых мы говорили до этого; в рамках курсовой детальная расшифровка не производится, хотя и может быть дана на усмотрение ответственного за курсовую работу.

### 3.1 Некоторые общие правила

Аббревиатуры должны быть расшифрованы перед первым упоминанием. Английские аббревиатуры понятий, не имеющих устоявшегося перевода на русский язык, должны быть расшифрованы на русском языке перед первым упоминанием.

Каждая часть задания должна быть выполнима – это значит, что результат выполнения должен быть оцениваем в принципе. Не всегда эта оценка количественная или бинарная, она может быть экспертной – результат может оценивать как руководитель, так и комиссия на защите.

Курсовая работа может быть посвящена любой теме, которая будет одобрена руководителем и утверждена заведующим кафедрой. При предложении темы можно опираться на то, что подавляющее большинство работ прошлых лет включают в себя алгоритмы трёхмерной графики – алгоритмы расчёта теней, отражений и отображения трёхмерных объектов за объектами.

Черновик задания готовится студентом для руководителя, подписывается<sup>1</sup> при согласии и подаётся на утверждение заведующему кафедрой в двух экземплярах.<sup>2</sup>

Задание студент обязан выполнить полностью, что, в свою очередь, требует аккуратности в формулировках и определениях. В первую очередь, следует обращать внимание на корректность **уточнений**.

Предложив руководителю обязать Вас *моделировать движение сакуры под порывами ветра*, Вы будете должны представить впоследствии модель *сакуры* — дерева, листа, цветка *сакуры*, а не общую модель дерева, указать отличия частного от общего – Вы обязаны будете продемонстрировать, чем сакура отличается от всех остальных деревьев. Это не значит, что нужно писать как можно более общие слова, но значит, что следует на этапе форму-

---

<sup>1</sup>Обратите внимание, что в конце многих официальных документов подписывающиеся указываются в порядке подписания. Поэтому в задании, которое руководитель назначает студенту, сначала указан руководитель; а расчётно-пояснительную записку же к выполненной курсовой работе сначала подписывает студент как исполнитель, и лишь потом – руководитель.

<sup>2</sup>В случае курсовой работы по компьютерной графике задание подаётся на утверждение лектору. Порядок был оглашён в середине семестра.

лировки задания определить допустимые пределы своей работы и то, какие сущности станут особенностями работы, а какие могут быть обобщены без увеличения трудозатрат на разработку.

**Хуже** Указать в задании *«Разработать программу для изображения движения деревьев под порывами ветра.(...)»* – Тема слишком общая и не имеет особенностей.

**Хуже** Указать в задании *«Разработать программу для изображения опадающих лепестков сакуры под порывами ветра.(...)»*, не будучи готовым описать модель лепестка сакуры в работе.

**Лучше** Указать в задании *«Разработать программу для изображения опадающих лепестков сакуры под порывами ветра.(...)»* и быть готовым на этапе анализа описать отличия рассматриваемых сущностей от более общих, например, типичные цвет лепестков, вес и форму.

**Хуже** Указать в задании *«Разработать программу для изображения опадающих лепестков сакуры под порывами ветра.(...)»*, будучи готовым на этапе анализа описать отличия рассматриваемых сущностей от более общих, но при этом в итоге уделить слишком большое внимание не относящимся к теме работы аспектам: плотности лепестков, зависимости их количества от времени года, etc.

**Хуже** Указать в задании *«Разработать программу для изображения вазы на столе.(...)»* – Тема слишком общая и не имеет особенностей.

**Хуже** Указать в задании *«Разработать программу для изображения вазы из прозрачного стекла на металлическом столе.(...)»*, не будучи готовым учесть прозрачность вазы и отражающую способность поверхности стола.

**Лучше** Указать в задании *«Разработать программу для изображения вазы из прозрачного стекла на металлическом столе.(...)»*, будучи готовым описать и учесть оптические характеристики поверхностей, модель внутреннего отражения стекла.

**Хуже** Указать в задании *«Разработать программу для изображения вазы из прозрачного стекла на металлическом столе.(...)»*, в итоге уделив большую часть внимания описанию только отражения света от металлической поверхности.

Не менее важно быть корректным при **обобщении**. Предложив руководителю обязать Вас *разработать программу для визуализации детского конструктора*, Вы по сути должны будете либо обеспечить на защите возможность визуализации любого конструктора, либо обосновать, почему только в середине работы было произведено уточнение.

**Хуже** *«Разработать программу для визуализации детского конструктора.(...)»*

**Лучше** *«Разработать программу для визуализации детского конструктора из кубиков.(...)»*

**Лучше** *«Разработать программу для визуализации детского конструктора из деталей. Перечень допустимых деталей указан в Приложении 1 к заданию.(...)»*

**Лучше** *«Разработать программу для визуализации детского конструктора из деталей в форме шара, куба или параллелепипеда.(...)»*

ТЗ это не поле для синонимов и точно не место для понятий, которые кажутся синонимичными на первый взгляд, но определены по-разному в ГОСТ. Руководители знают о существовании различий между «программой» и «программным комплексом», «архитектурой программы» и «структурой программного обеспечения», «разработкой», «проектированием» и «реализацией»!

В задании **не** указываются алгоритмы реализации, потому что алгоритмы выбираются обоснованно студентом на этапе анализа. Исключительными являются случаи, когда алгоритм явно выбран самим руководителем в силу каких-либо причин, но таких исключений не может быть слишком много в одной работе.

В задании **не** указываются средства реализации, выбираемые впоследствии обоснованно студентом на технологическом этапе. Исключительными являются случаи, когда алгоритм явно выбран самим руководителем в силу каких-либо причин, но таких исключений не может быть слишком много в одной работе. Эта рекомендация, помимо прочего, связана с потенциальным разрывом между амбициями студента в начале работы и его возможностями в конце.

**Недопустимо** Указать в задании *«(...)Требуется реализовать алгоритм отсечения объектов с использованием технологии CUDA(...)»* и не реализовать указанное.

**Хуже** Указать в задании *«(...)Требуется реализовать алгоритм отсечения объектов с использованием технологии CUDA(...)»* и реализовать указанное, опоздав ко сроку защиты курсовой работы.

**Лучше** Не указав в задании явно деталей реализации, реализовать обычную версию выбранного на этапе анализа алгоритма.

**Лучше** Не указав в задании явно деталей реализации, реализовать версию алгоритма с особенностями, а потом исследовать её характеристики.



Присутствие в работе моделирования какого-либо физического процесса поощряется, но следует помнить, что курсовая работа посвящена компьютерной графике. При упоминании моделирования в любом виде нужно быть готовым представить модель, поэтому существует разница между «моделированием движения качелей» и «визуализацией движения качелей».

Графический интерфейс в реализованной в ходе выполнения курсовой работы программе обязателен, хотя и не описывается подробно, так как студент второго курса ещё не знаком с формальной нотацией диаграмм прецедентов. В задании не указывается необходимость разработки графического интерфейса, только его наличие, потому что ответом на «разработку программного интерфейса» служит дополнительная документация.

Для всех варьируемых параметров ещё на этапе формулировки задания следует определить, будет предоставлена возможность изменения параметров пользователю на этапе выполнения или только программисту на этапе трансляции. Если это не указано, обычно подразумевается, что возможность изменять все входные параметры пользователю дана!

Задание формулируется в императивной форме по отношению к разработчику, но не к пользователю.

**Хуже** *Пользователь будет добавлять фигуры следующих типов: куб, пирамида, шар.* – Необоснованное утверждение. Сформулировано не в форме требования.

**Хуже** *У пользователя будет возможность добавлять фигуры следующих типов: куб, пирамида, шар.* – Сформулировано не в форме требования.

**Лучше** *Пользователю должна быть предоставлена возможность добавлять фигуры следующих типов: куб, пирамида, шар.*

**Лучше** *Должна быть реализована возможность добавлять фигуры следующих типов: куб, пирамида, шар.*

**Лучше** *Должна быть реализована возможность добавлять фигуры из предопределённого списка.* – Допустимо при условии, что в приложении к заданию указан этот список.

**Лучше** *Должна быть реализована возможность добавлять фигуры во время выполнения программы из предопределённого списка.* – Допустимо, но упоминание времени выполнения можно в данном случае опустить.

**Хуже** *Программа должна обеспечить пользователю выбор типа рельс: прямые, развилочные, закругленные.* – Программа не может долженствовать.

**Лучше** *Должна быть предусмотрена возможность выбора типа рельс: прямых, развилочных, закругленных.*

**Лучше** Обеспечить пользователю возможность типа рельс: прямых, развилочных, закругленных.

## 3.2 Алгоритм

Алгоритм написания черновика задания для руководителя:

1. Выбрать тему работы для предложения руководителю.
2. Выделить входные и выходные данные. Выходными данными программы по итогу курсовой работы обычно являются либо кадр, либо видео.
3. Начав задание со слов «разработать программу для...», описать суть работы в одном предложении.
4. Перечислить входные данные, ограничения на оные, при необходимости – формат и этап получения данных.
5. Добавить упоминание графического интерфейса.
6. Добавить упоминание исследования – на усмотрение руководителя либо конкретное исследование, либо *исследование характеристик программного обеспечения*.

**Пример полного задания №1** Разработать программу для построения трёхмерного изображения мыльных пузырей вещества. Количество, мощность и расположение белых точечных источников света задаются пользователем. Предусмотреть наличие источника света на бесконечности. Предусмотреть возможность изменения геометрии мыльных пузырей в ходе расчёта. Оптические свойства поверхностей объектов – коэффициенты рассеянного, диффузного и зеркального отражения, коэффициент пропускания, коэффициент преломления – задаются пользователем. Физические свойства вещества задаются пользователем на этапе выполнения. Реализовать графический интерфейс пользователя. Исследовать характеристики разработанного программного обеспечения.

**Пример полного задания №2** Разработать программу для построения трёхмерного изображения планетарной системы со звездой в центре. Звезду считать идеальной сферой с массой и радиусом, задаваемыми пользователем. Планеты считать выпуклыми объектами, геометрия планет задаётся пользователем в виде сеток-описателей. Начальные координаты центров масс и начальные скорости задаются пользователем. Для расчёта траектории движения системы использовать явную схему Эйлера решения системы дифференциальных уравнений, при расчёте движения сводить планеты к их



центрам масс. Предусмотреть наличие источника света на бесконечности. Реализовать графический интерфейс пользователя. Исследовать зависимость скорости синтеза кадра от количества полигонов.

**Пример полного задания №3** Разработать программу для построения трёхмерного изображения озера с растительностью и птицами фламинго. Предусмотреть наличие источника света на бесконечности. Предусмотреть наличие точечных статичных источников света. Поверхность озера считать плоской. Оптические свойства поверхностей объектов задаются пользователем на этапе выполнения. Количество и расположение птиц, плотность растительности, состоящей из множества моделей одной формы вокруг озера, число и расположение источников света задаются пользователем на этапе выполнения. Реализовать графический интерфейс пользователя. Исследовать зависимость скорости генерации кадра от плотности растительности.

**Пример полного задания №4** Разработать программу для построения трёхмерного изображения качелей и двигающего по поверхности качелей металлического шара. Для расчёта траектории движения системы использовать явную схему Эйлера решения системы дифференциальных уравнений. Количество, мощность и расположение белых точечных источников света задаются пользователем. Предусмотреть наличие источника света на бесконечности. Предусмотреть возможность изменения геометрии в ходе расчёта. Оптические свойства поверхностей объектов задаются на этапе трансляции. Физические свойства вещества задаются пользователем на этапе выполнения. Реализовать графический интерфейс пользователя. Исследовать характеристики разработанного программного обеспечения.

### 3.3 Самые частые ошибки

Перед чистовой вёрсткой проверить:

1. Нигде не упомянут **курсовой проект** – уже несколько лет как по компьютерной графике в учебном плане стоит курсовая работа.
2. Лица, подписывающие документ, указаны в конце документа в нужном порядке.
3. Нигде не упомянут **эксперимент** – в курсовой работе проводится исследование. Если студент предлагает руководителю провести эксперимент, требуется представить план эксперимента.
4. Нигде не упомянута **апробация** – апробация учебной курсовой работы проводиться не может.