

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Уральский государственный педагогический университет»
Институт математики, информатики и информационных технологий
Кафедра информатики, информационных технологий
и методики обучения информатике

ОБУЧЕНИЕ СТУДЕНТОВ ОСНОВАМ ТВЕРДОТЕЛЬНОГО МОДЕЛИРОВАНИЯ С ПОМОЩЬЮ ПРОГРАММЫ OPENSCAD

*Выпускная квалификационная работа
бакалавра по направлению подготовки
44.03.01 – Педагогическое образование (Профиль: Информатика)*

Исполнитель: студент группы БИ-41
Института математики, информатики и ИТ
Пяткова А.Н.

Руководитель: ассистент кафедры ИИТиМОИ
Алексеевский П.И.

Работа допущена к защите
« ____ » _____ 2017 г.
Зав. кафедрой _____

Екатеринбург – 2017

Реферат

Пяткова А.Н. ОБУЧЕНИЕ СТУДЕНТОВ ОСНОВАМ ТВЕРДОТЕЛЬНОГО МОДЕЛИРОВАНИЯ С ПОМОЩЬЮ ПРОГРАММЫ OPENSCAD, выпускная квалификационная работа: 36 стр. рис. 10, библи. 32 назв., приложений 1.

Ключевые слова: моделирование, средства автоматизации проектирования, OpenSCAD, комплекс лабораторных заданий, обучение студентов моделированию.

Объектом данного исследования является процесс обучения работе с CAD-системами.

Предметом исследования является обучение студентов основам твердотельного моделирования с использованием свободного ПО.

Цель работы – разработать комплекс лабораторных заданий для обучения студентов основам твердотельного моделирования при помощи свободного средства автоматизации проектирования.

В работе произведен анализ федеральных государственных образовательных стандартов для выявления потребности изучения основных возможностей пакета OpenSCAD студентами высших образовательных учреждений двух ступеней образования различных направлений подготовки. Также, на основе сравнительного анализа наиболее популярных на рынке программных средств автоматизации проектирования, имеется обоснование выбора свободного программного обеспечения OpenSCAD в рамках изучения основ твердотельного трехмерного моделирования.

Оглавление

ВВЕДЕНИЕ.....	4
ГЛАВА 1.....	6
1.1 ПРИМЕНЕНИЕ МОДЕЛИРОВАНИЯ В ОБРАЗОВАНИИ	6
1.2 ИЗУЧЕНИЕ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЯ НА РАЗЛИЧНЫХ СТУПЕНЯХ ОБРАЗОВАНИЯ	14
1.3 АНАЛИЗ ПРОГРАММНЫХ СРЕДСТВ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ.....	19
ГЛАВА 2.....	33
2.1 ОПИСАНИЕ РАЗРАБОТАННОГО КОМПЛЕКСА ЛАБОРАТОРНЫХ ЗАДАНИЙ	33
2.2 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К РАЗРАБОТАННОМУ КОМПЛЕКСУ ЛАБОРАТОРНЫХ ЗАДАНИЙ.....	36
2.3 РЕЗУЛЬТАТЫ АПРОБАЦИИ	45
ЗАКЛЮЧЕНИЕ	48
СПИСОК ЛИТЕРАТУРЫ	49
ПРИЛОЖЕНИЯ	53
ПРИЛОЖЕНИЕ 1.	53
ПРИЛОЖЕНИЕ 2.	85

Введение

3D-печать - это перспективная сфера развития современных технологий. Её внедряют и применяют повсеместно: начиная с создания деталей для станков, заканчивая печатью донорских органов в медицине. Для того, чтобы получить конечный напечатанный продукт, необходимо уметь проектировать его модель в электронном виде. Данный процесс организуется при помощи работы в программах, относящихся к САД-системам. Возникает необходимость обучения эксплуатации данного программного обеспечения специалистов разных сфер. Но современные САД-системы, используемые и востребованные на рынке, имеют высокую стоимость. Государственные учреждения, занимающиеся подготовкой квалифицированных кадров, не обладают достаточным количеством финансовых средств для обеспечения учащихся дорогостоящим программным обеспечением. Поэтому появляется потребность в поиске такой программы, которая, во-первых, обладала бы всеми основными инструментами САД-систем, во-вторых, имела минимальную цену или находилась в свободном доступе. На данный момент имеющийся методический материал направлен на обучения работе с дорогими коммерческими программными продуктами, а информация по свободным аналогам разрознена. Поэтому, помимо поиска дешевой САД-системы, необходимо разработать методику изучения данного программного обеспечения.

Объектом данного исследования является процесс обучения работе с САД-системами.

Предметом исследования является обучение студентов основам твердотельного моделирования с использованием свободного ПО.

Цель работы – разработать комплекс лабораторных заданий для обучения студентов основам твердотельного моделирования при помощи свободного средства автоматизации проектирования.

Для достижения цели поставлены следующие задачи:

1. Произвести анализ федеральных государственных образовательных стандартов на выявление потребности в изучении моделирования.
2. Произвести анализ федеральных государственных образовательных стандартов различных ступеней образования, направленный на выявления актуальности изучения трехмерного моделирования.
3. Произвести анализ представленных на рынке программных средств автоматизации проектирования и выбрать программное средство для обучения студентов основам твердотельного моделирования.
4. Разработать комплекс лабораторных заданий, направленный на изучение основ твердотельного моделирования при помощи выбранного средства автоматизации проектирования.
5. Подготовить методические рекомендации по использованию разработанного комплекса лабораторных заданий в учебном процессе.

Глава 1.

1.1 Применение моделирования в образовании

Моделирование используется уже многие годы и является эффективным инструментом для науки и техники. В учебном процессе обучающиеся должны изучить основы науки, но приоритетным является не усвоение стандартного набора фактов и закономерностей, а овладение методами и идеями, которые выделяют ее. К сожалению, заученная информация остается в памяти человека не долго, а вот методы, которые он применял не раз и в различных областях сохраняются в течении длительного времени. Одним из таких методов является моделирование.

Рассмотрим подробнее понятие моделирование.

Модель – это объект, созданный искусственно, который заменяет объект реального мира, воспроизводя необходимое число свойств. Процесс создания модели называется моделированием [4].

Моделирование – это метод познания окружающей действительности, который используют разные науки [9].

Метод моделирования обладает большой эвристической силой, так как благодаря ему становится возможным сведение сложного к простому, чтобы сделать объект доступным для понимания и более глубокого исследования. Таким образом, уже на данном этапе можно говорить об актуальности моделирования в обучении [5].

На данный момент моделирование чаще используется в виде компьютерного моделирования. В учебном процессе получили развитие три направления:

- * компьютер как объект изучения;
- * компьютер как инструмент обучения;
- * компьютер как инструмент познания.

В рамках первого направления рассматриваются собственные задачи информатики, такие как кодирование и обработка информации. Второе направле-

ние ориентировано на изучение компьютера как инструмента обработки информации. Последнее направление связано с использованием компьютерного моделирования и это самое важное направление, которое имеет большой потенциал. Стоит отметить это то, что наиболее эффективно является применение информационных технологий в учебном процессе, когда компьютер реализует возможности, которые недоступны в стандартных условиях.

В обучении информационные технологии имеют все больший вес, ведь выполняют интегрирующую функцию. Практическая реализация данной функции требует наличие достаточно глубоких межпредметных связей. Компьютерное моделирование применяется в совершенно различных областях, поэтому каждый студент и школьник имеет возможность получить позитивный, а главное полезный именно для него опыт в использовании информационных технологий. Преподаватель, в свою очередь, имеет большой арсенал в изучении своего предмета. Но достигнуть это возможно только при регулярном практическом применении компьютерного моделирования [5]. Развивая межпредметные связи, обучающийся увеличивает свою информационную базу в виде различных содержательных задач, при решении которых появляется новый стимул в дальнейшем освоении информационных технологий, а также создает условия для развития творческих навыков. Интеграция информатики с другими предметными областями реализуется путем решения прикладных задач, которые развивают навык использования компьютерного моделирования, а далее обучающийся применяет этот навык в освоении других дисциплин.

Включение данной связи в обучение с абсолютно новыми возможностями дает не только положительный эффект, но и является хорошим стимулом к дальнейшему развитию. Построение и анализ моделей, проведение экспериментов формирует глубокое и устойчивое понимание законов протекания процессов. Также можно говорить о расширении знаний в различных предметных областях, развитии познавательной и творческой активности.

Благодаря моделированию возможно получение замены реального объекта его моделью. Причем модель при исследовании является более наглядной и доступной, чем реальный объект. В модели устранены все не существенные свойства, которые могут отвлекать студента от главного, что упрощает усвоение необходимой информации о свойствах моделируемого объекта.

Само понятие модели включает несколько компонентов:

- * объект моделирования;
- * решаемая задача;
- * способ построения модели;
- * реализация модели [9].

В данном комплексе задача определяет характер модели, то есть определяет все существенные свойства моделируемого объекта, которые необходимо отобразить в создаваемой модели. При отсутствии конкретной задачи создание модели не имеет смысла.

Одному объекту могут соответствовать несколько моделей, связанных с разными задачами. При решении задачи лучше создавать несколько моделей, которые будут отличаться формами представления или воспроизведением свойств реального объекта [31].

Любой объект многогранен, поэтому необходимо построение нескольких моделей для более тщательного изучения всех групп свойств. Однако возможно построение и различных моделей по одному набору заданных свойств, но с условием разной степени детализации.

Таким образом, моделирование как технология имеет вариативность в построении и выборе типа модели.

Выделим основные функции моделей в науке и практической деятельности:

- * функция познания, то есть получение совершенно новых знаний об объекте, также о его функционировании;
- * передача информации;

- * выявление свойств и закономерностей;
- * решение задач оптимизации;
- * решение задач управления состоянием объектов, протеканием процессов;
- * построение объектов с заданными свойствами;
- * диагностика состояния объекта;
- * прогнозирование поведения объекта или развития процесса;
- * имитация объектов;
- * создание тренажеров;
- * создание игровых и когнитивных моделей обучения [31].

Каждая наука отводит особую роль для концептуальных моделей, то есть для представлений, сложившихся в сознании человека, о моделируемом объекте. Создание таких моделей производится на основе наблюдений, а также теоретической базе разработчика. Для создания концептуальной модели необходимо применение не только текущих знаний, но выявление действительно существенных для исследования свойств. Концептуальные модели являются простейшими, отражающими заданный набор свойств реального объекта, который интересует данную науку. Такие модели рождаются в ходе практической деятельности и связаны с ней.

Научные законы формируются как описание взаимосвязанных концептуальных моделей. Таким образом, можно сделать вывод, что научные законы также являются в определенном смысле моделями. На основе концептуальных моделей строятся модели классов явлений и образуют научные теории. Понятие гипотезы тоже можно считать моделью, но в условиях ограниченной изученности реального объекта. Поэтому, наука допускает существование одновременно нескольких гипотез, ведь одно и тоже наблюдение люди могут аргументированно объяснять с разных точек зрения.

Моделирование позволяет существенно сократить затраты на проведение эксперимента, в рамках проекта или курсовой работы студента,

что является не маловажным фактом в ограниченном бюджете образовательных учреждений.

Обучение управлению и регулированию сложных технических систем происходит на компьютерных моделях. В рамках учебного процесса сложно и слишком дорого реализовать реальную техническую систему, но возможно управление компьютерной системой, построенной на основе реально существующей [4]. Такой подход позволяет обучить студентов учету возможных ограничений, установлению оптимальных режимов функционирования системы, то есть полностью погрузить обучающегося в более приближенную к реальной жизни ситуацию. Для своевременного распознавания приближения аварийной ситуации, также применяется моделирование. Подготовка будущего специалиста включает в себя всестороннюю подготовку, поэтому для решения задач диагностики аварийных ситуаций компьютерное моделирование подходит лучше всего, так как реальную аварийную ситуацию целенаправленно никто не допустит. Компьютерная модель предоставляет информацию о некоторых признаках или состояниях, которые предшествуют аварии, то есть обучающийся может самостоятельно проанализировать всю динамику развития аварийной ситуации [8]. В итоге, будущий специалист получает опыт в определении состояний, которые предшествуют аварии, и может своевременно прекратить эксплуатацию системы и провести с ней ремонтные работы.

Все перечисленное выше показывает, что моделирование играет одну из самых важных ролей в науке и подготовке качественных кадров.

В сфере информационных технологий модель имеет две составляющие – аппаратную и программную. Интерпретация программной составляющей происходит процессором компьютера. Только в данном случае компьютерная модель отражает свойства моделируемого объекта.

Выделим особенности компьютерного моделирования [9]:

* Компьютер является мощным инструментом проведения экспериментов в области моделирования, так как у него есть возможность хранить и обрабатывать большие объемы информации;

* Компьютерное моделирование создает все условия для исследования сложных моделей и помогает анализировать множество факторов;

* Компьютер, как инструмент в моделировании, дал толчок к формированию новых направлений в моделировании;

* Многие науки, которые ранее могли развиваться только как описательные, теперь имеют компьютерные модели, которые стали основой математизации для этих наук;

* Компьютерное моделирование позволяет визуализировать как промежуточные, так и окончательные результаты исследования, благодаря виртуальной реальности;

* Компьютер предоставил возможность полной автоматизации построения, анализа и преобразования моделей.

Построение компьютерной модели в рамках учебного процесса реализуется несколькими шагами [4]:

1. Описание объекта. Преподаватель должен показать, как правильно выбирать существенные свойства реального объекта исходя из заданных условий задачи.

2. Создание описания объекта средствами языка математики.

3. Преобразование математической модели в вычислительную.

4. Разработка программ для реализации решения задачи моделирования.

К сожалению, эта схема моделирования очень громоздка и инертна. Значит можно сделать вывод, что существует проблема приближения компьютера к специалисту в данной предметной области [31]. Данную проблему можно решить путем создания инструментальных программных комплексов моделирования.

Очень долгое время компьютерное моделирование не имело широкого применения в образовательном процессе из-за необходимости создания моделей средствами профессиональных языков программирования. В современном мире программирование – это самостоятельная, специфичная и очень сложная дисциплина, которая требует больших затрат как времени, так и сил. Благодаря применению инструментальных программных комплексов визуального моделирования стало возможным внедрение моделирования в учебный процесс и на уровень школьного обучения, так как в данных условиях появилась возможность быстрой разработки моделей, а также проведения модельного эксперимента, при этом программирование, как написание кода, не требуется совсем. Программные комплексы визуального моделирования имеют ряд существенных преимуществ [8]:

- * конструирование модели за минимальное количество времени;
- * наглядное представление результатов моделирования;
- * возможно варьирование значение некоторых параметров модели в ходе эксперимента.

Построение моделей становится значительно проще, поэтому отводится больше учебного времени на творческую деятельность, то есть основа таких занятий – это проведение модельного эксперимента [31]. Данный факт имеет особую ценность в образовании. Такие программные комплексы осваиваются быстро и качественно школьниками и студентами.

Тема моделирование обязательная для изучения школьниками и студентами соответствующего направления подготовки. Поэтому важно насколько хорошо усвоена данная тема.

К сожалению, многие учащиеся, которые уже имеют опыт в изучении информатики, математики, физики и других наук, не понимают, что все время обучения имеют опыт с моделями. Следовательно, такие учащиеся имеют очень ограниченные представления о моделировании или не имеют такого представления совсем.

Однако те обучающиеся, которые имеют представление о модельном характере наук, а также знакомы с различными понятиями моделирования, имеют сформированное правильное научное мировоззрение, а это способствует обогащению методологического аппарата, значительно меняет отношение студентов и школьников к различным дисциплинам, к обучению в общем, а также создает все условия для более осмысленной и продуктивной учебной деятельности.

Моделирование целесообразно использовать не только как обучение конкретному навыку, но при знакомстве учащихся с абстрактным понятием, то есть реализовать это понятие в виде модели, тогда учащиеся легче смогут проникнуть в суть и начать осмысленно оперировать данным понятием.

Таким образом, можно выделить основные достоинства применения моделирования в учебном процессе:

- * применение модели сокращает сроки проведения исследования;
- * применение модели значительно уменьшает затраты на проведение исследования;
- * возможность проведения эксперимента без отрицательных последствий для окружающих;
- * снижение трудоемкости;
- * повышение интеллектуальной, познавательной и творческой активности;
- * наглядное представление результатов эксперимента;
- * возможность сохранения результатов эксперимента;
- * более глубокое освоение информационных технологий;
- * возможность проведения более тщательного анализа исследования;
- * решение сложных задач, сложно реализуемых в реальных условиях;
- * возможность обучения в игровой, творческой форме.

Таким образом, инструментальные программные комплексы моделирования способствуют повышению качества преподавания, а также более эффек-

тивному усваиванию знаний учащимися. В результате такого обучения, знания будут не просто законспектированы, а получены в ходе творческой активности. Значит, моделирование является неотъемлемой частью учебного процесса [26].

1.2 Изучение трехмерного моделирования на различных ступенях образования

В государственных федеральных стандартах обучения студентов некоторых направлений подготовки, прописаны требования по изучению средств компьютерной автоматизации проектирования с целью создания моделей различного применения.

В настоящее время многие профессии связаны с созданием всевозможных моделей при помощи программ, позволяющих моделировать трёхмерные объекты. Данный процесс осуществляется при помощи программ, позволяющих создавать конструкции для различных целей и применений. Исходя из потребности в квалифицированных кадрах, соответствующих требованиям работодателей, возникает необходимость обучения эксплуатации данного программного обеспечения специалистов разных сфер. Проведённый анализ федеральных государственных образовательных стандартов высшего профессионального образования двух уровней подготовки по каждой дисциплине выявил, что умение студента создавать модели при помощи программного обеспечения является одной из основных профессиональных задач, относящихся к проектно-конструкторской деятельности, по нескольким направлениям [6].

Системы автоматизированного проектирования, основывающиеся на трехмерном моделировании, в настоящее время становятся стандартом для создания конструкторской и технологической документации. Это обуславливает специальные требования к подготовке инженеров в техническом вузе. В процессе организации профессиональной подготовки как необходимое условие выступают: формирование заданных уровней компетентности, профессиональная культура специалиста, развитие его потребностей в постоянном профессио-

нальном самосовершенствовании. Данные условия являются базовыми для эффективной деятельности в обстановке конкурентно способной среды [5].

На фоне этого наблюдается высокий уровень мотивации будущих специалистов к изучению методов компьютерной графики, возрастает роль графической подготовки в современном техническом образовании.

Несмотря на значительное количество исследований по методике преподавания графических дисциплин, технология обучения на основе 3D-моделирования требует особого внимания и отдельного научного исследования. Развитие и применение современных графических пакетов при изучении графического цикла дисциплин обусловлены спецификой предмета, требующей развитого пространственного мышления, умений воспринимать и производить графическую информацию.

Пространственное мышление, как и любую другую способность человека, нужно и можно развивать. С помощью трехмерного моделирования в среде графических пакетов задача визуального представления геометрических объектов значительно упрощается [29].

Сегодня трехмерное моделирование (или 3D-моделирование) уже не является новинкой, но еще 5-7 лет назад бурно шли споры среди педагогов-теоретиков и практиков: нужна ли эта технология, и как её применять. Время решило всё само и показало нужность и эффективность данного направления визуализации. Теперь педагогам-методистам приходится разрабатывать и придумывать различные технологии применения 3D-моделей и объектов в образовательном процессе.

Современная концепция высшего профессионального образования требует качественной подготовки специалистов, соответствовавших новым требованиям, предъявляемым к современной профессиональной деятельности. Информационный характер системы образования создает предпосылки для универсализации средств и ресурсов межпредметных связей и средств их реализации, предлагая свои средства, методы и формы. Целью этого процесса является

формирование у студентов системного научного мышления, экологической и информационной культуры, творческой активности и высокой нравственности – качеств, которые позволят им успешно адаптироваться, жить и работать в глобальном информационном обществе [19]. Для повышения профессиональной компетенции будущих специалистов необходимо реализовать главные идеи реформирования высшего образования, одним из направлений которого, является внедрение в образовательный процесс компьютерных технологий. Информационные технологии оказывают большое влияние на подготовку к будущей профессиональной деятельности. В результате их использования в вузовском учебном процессе повышаются эмоциональный отклик студентов на процесс познания, мотивация учения, интерес к овладению новыми знаниями, умениями и практическое их применение, способствует развитию способностей студентов, активизирует мышление. Именно трехмерное моделирование в ходе создания чертежей позволяет, начиная с первого курса обучения, студентам сформировать образ изучаемого объекта.

Использование трехмерного моделирования позволяет создать визуальный образ объекта, использовать цвет, анимацию, но, тем не менее, не должно отвлекать внимание обучающихся от решения поставленных задач. Умение анализировать ортогональный чертеж геометрического объекта, расчленить его сложную форму на простые составляющие геометрические тела – позволит легко переходить от 3D-моделей к плоским чертежам, при этом значительно упрощая процесс редактирования чертежей.

Подробный анализ образовательных стандартов позволил выделить ряд направлений подготовки бакалавров, профессиональная деятельность которых напрямую связана с созданием компьютерных твердотельных моделей:

«Техническая физика» [23]. Студент должен уметь проектировать детали на элементном уровне с использованием средств автоматизации проектирования.

«Информационные системы и технологии». Обучающиеся обязаны изучать создание деталей при помощи средств компьютерного проектирования [17].

«Приборостроение». Бакалаврам данного направления необходимо уметь работать с программным обеспечением, позволяющим конструировать типовые детали в электронном виде [16].

«Конструирование и технология электронных средств». Программа данной специальности должна научить бакалавров проектированию деталей электронных средств с помощью средств автоматизации проектирования [18].

Разработанный комплекс лабораторных заданий направленный на изучение студентов твердотельному моделированию получит широкое применение в подготовке магистрантов по следующим направлениям:

«Техническая физика». Студентам необходимо уметь работать со средствами компьютерного проектирования необходимых для разработки рабочих проектов изделий [19].

«Ракетные комплексы и космонавтика». В стандарте данного направления указано, что обучающиеся должны уметь использовать средства автоматизации проектирования в моделировании деталей аппаратуры, необходимые для проведения экспериментов [20].

«Информационные системы и технологии». По окончании данного курса магистрант обязан обладать умениями проектирования сложных изделий с использованием программного обеспечения [21].

Таким образом, федеральные государственные образовательные стандарты высшего профессионального образования по вышеперечисленным направлениям подготовки обуславливают целесообразность создания программы обучения, применимой для обучения студентов бакалавриата и магистратуры твердотельному моделированию при помощи средств автоматизации компьютерного проектирования [26].

Исходя из того, что изучение твердотельного моделирования в большей части ориентировано на студентов высших учебных заведений, делаем вывод, что формат предоставления учебного материала должен опираться на федеральные государственные образовательные стандарты высшего профессионального образования, в которых указано, что формат организации образовательной программы должен содержать в себе лабораторные задания по главным дисциплинам обучения [22].

Проведя анализ федеральных государственных образовательных стандартов основного общего, среднего общего и среднего специального образования приходим к выводу, что комплекс лабораторных заданий, направленный на изучение программного обеспечения основ твердотельного моделирования, применим для школьников и для учащихся получающих среднее специальное образование. Данный комплекс поможет обучающимся сформировать представление о роли информатики и информационно-коммуникационных технологий в современном мире, о тенденциях развития компьютерных технологий, о влиянии информационных технологий на жизнь современного человека и о вкладе информатики в развитие науки, что является полным соответствием требований федеральных государственных образовательных стандартов [24]. Наиболее полезным является применение комплекса лабораторных заданий в учебных заведениях, имеющих и реализующих работу с 3D принтерами. Ученики смогут увидеть результат своей деятельности, распечатав свою собственную созданную трехмерную модель, тем самым осознать практическую полезность полученных знаний и проявить интерес к предмету, что также полностью соответствует государственным образовательным стандартам.

Также в профессиональном стандарте педагога имеется пункт, связанный с обучением школьников работе с 3D моделированием, который гласит, что педагог обязан уметь создавать самостоятельно и совместно с учениками наглядное представление неких объёмных моделей при помощи компьютера и уметь использовать и применять в своей деятельности 3D-принтеры [25]. Следова-

тельно, комплекс лабораторных работ, направленный на обучение твердотельному моделированию, применим в равной степени как для процесса образования школьников, так и для подготовки педагогов.

Таким образом, проведя анализ государственных федеральных стандартов приходим к выводу, что методика обучения твердотельному моделированию, актуально для использования в процессе обучения школьников основного общего и среднего общего образования, учителей на основе требований профессионального стандарта педагога и главным образом студентов двух уровней подготовки (магистратура и бакалавриат) высшего профессионального образования [26].

1.3 Анализ программных средств автоматизации проектирования

В государственных федеральных стандартах обучения студентов различных направлений подготовки прописаны требования по изучению средств автоматизации проектирования с целью создания моделей различного применения в том числе 3D печати [11], которая является перспективной сферой развития современных технологий. Её внедряют и применяют повсеместно: начиная с создания деталей для станков, заканчивая печатью донорских органов в медицине. Для того, чтобы получить конечный напечатанный продукт необходимо уметь проектировать его модель в электронном виде. Данный процесс организуется при помощи работы в программах, относящихся к CAD-системам. Стоит отметить, что программное обеспечение должно иметь простой язык, так как разработанная на основе данной программы методика должна быть применима для обучения пользователей с базовым уровнем знаний [15]. Большая часть программного обеспечения, представленного на рынке, имеет высокую стоимость. Государственные образовательные учреждения, занимающиеся подготовкой квалифицированных кадров, не обладают достаточным финансовым обеспечением. Следовательно, появляется потребность в поиске такой про-

граммы, которая обладала всеми основными инструментами CAD-систем, имела минимальную цену или находилась в свободном доступе и была проста в обучении. Стоит отметить, что персональные компьютеры образовательных учреждений работают под управлением различных операционных систем, поэтому важным критерием выбора ПО будет поддержка возможности работы на разнообразных программно-аппаратных платформах.

Использование компьютеров для представления трехмерных объектов, несомненно, является одним из достижений XX века. Редакторы трехмерной графики заметно облегчают и раскрашивают жизнь, как простых пользователей, так и профессионалов в разных сферах деятельности [13]. Использование 3D-редакторов ведет к росту производительности труда в той или иной области, сокращению времени работы над проектом, уменьшению их стоимости и затрат на производство [28].

Проведя анализ всех факторов, влияющих на выбор программного обеспечения, применимого для обучения твердотельному моделированию [29], и в соответствии с требованиями стандартов [22] выделяются следующие критерии: низкая стоимость, принадлежность к CAD-системам, возможность моделирования путем написания программного кода, простой язык программирования, поддержка 3D-принтеров, возможность твердотельного моделирования, работа на основе множества операционных систем. Выбранное далее программное обеспечение обязано удовлетворять основные сформулированные требования к ПО.

Для выявления самого оптимального программного обеспечения, удовлетворяющего требованиям стандартов, необходимо проанализировать основные пакеты, занимающие устойчивое положение на рынке. Рассмотрим следующие программы, позволяющие создавать трехмерные модели объектов: 3Ds max, Maya, SolidWorks, Компас-3D, AutoCad, CATIA, Blender, OpenSCAD.

Первым рассмотрим самый популярный и часто встречаемый программный пакет 3Ds max. Программа имеет удобный инструментарий, что позволяет

повысить эффективность работы в создании трехмерных объектов. При помощи пакета 3Ds max пользователям предоставляется возможность создания на профессиональном уровне трехмерной анимации, визуализации и моделирования. Приложение специализируется на создании трехмерных объектов. Интерфейс программы и процесс моделирования в данном программном обеспечении представлены на рисунке 1. Разработчиком является американская компания Autodesk, специализирующаяся на выпуске программ систем САПР. Программа разработана на языке программирования C++. Первая версия пакета появилась в 1990 году, последняя в апреле 2017. 3Ds max не является свободным программным обеспечением, но имеет студенческую версию для обучения, которая является бесплатной, но для установки лицензионной версии студенческого пакета необходима регистрация. Полная лицензионная версия 3Ds max обладает стоимостью две тысячи четыреста евро. Программа может быть установлена на операционной системе Windows. 3Ds max представляет из себя интегрированную среду для создания трехмерных моделей на профессиональном уровне. Также программа позволяет создавать анимацию для различных объектов, которые в свою очередь, обладают максимальной фотореалистичностью. Что позволяет использовать данное программное обеспечение при создании компьютерных игр, мультипликации и графики, в киноиндустрии. Чаще всего пакет применяется для моделирования планируемых объектов, не созданных в настоящем времени. Создание трехмерных объектов различной сложности реализуется при помощи всевозможных техник моделирования, начиная от моделирования при помощи полигонов, заканчивая моделированием при помощи примитивов. Все техники могут применяться как отдельно друг от друга, так и вместе. 3Ds max обладает огромным количеством различных возможностей. Очень важной является способность программы учитывать физические свойства реального мира в виртуальном, для этого существует этап визуализации. Для удобства создания трехмерных моделей существуют различные плагины, позволяющие упростить работу пользователя. Данные расширения представлены

как в платных версиях от разработчика, так и в бесплатных пользовательских версиях. Возможность использовать дополнительные инструменты значительно упрощает процессы создания некоторых моделей [12].



Рис.1. Процесс создания трехмерной модели в программе 3Ds max

Следующая по популярности - программа Maya, созданная компанией Autodesk при помощи программирования на языках C++ и Python. Данный пакет содержит огромное количество всевозможных инструментов нужных для создания трехмерных объектов и максимально реалистичной анимации, в том числе возможность визуализации анимированных 3D-сцен, что позволяет программе находиться на лидирующей позиции по созданию графики в киноиндустрии и анимационных фильмах. Пример работы с программой представлен на рисунке 2. Майя включают в себя естественные законы физики чтобы контролировать поведение виртуальных объектов в компьютерной анимации. Maya может производить видео, которые более реалистичны, относительно других программ. Первый релиз Maya вышел в 1998 году, последний в июле 2016. Данная программа может быть установлена на платформах Mac OS X, Windows, Linux. Программа Maya имеет три версии: полную, базовую и студенческую. Стоимость полного пакета три тысячи триста восемьдесят евро. Студенческая версия предоставляется бесплатно, но имеет ряд ограничений и каждый итоговый продукт помечается водяным знаком для того, чтобы версия не использовалась в коммерческих целях. Программный интерфейс Maya пол-

ностью настраивается, при помощи редактирования кода программы, для тех пользователей, которым требуется максимальная производительность. Maya позволяет пользователям расширять свои функциональные возможности, предоставляя доступ к встроенному языку Maya MEL. С помощью MEL вы можете настроить пользовательский интерфейс, написать сценарии и макросы. Также Maya имеет возможность переводить готовые проекты в программный код. Кроме того, для повышения мощности и функциональности Maya, доступен полный интерфейс прикладных программ (API). Пользователи имеют возможность использовать предоставленные Майя API на основе Python. Из-за огромного количества инструментов и различных возможностей программного пакета, Maya становится более сложной в изучении относительно других программ [30].

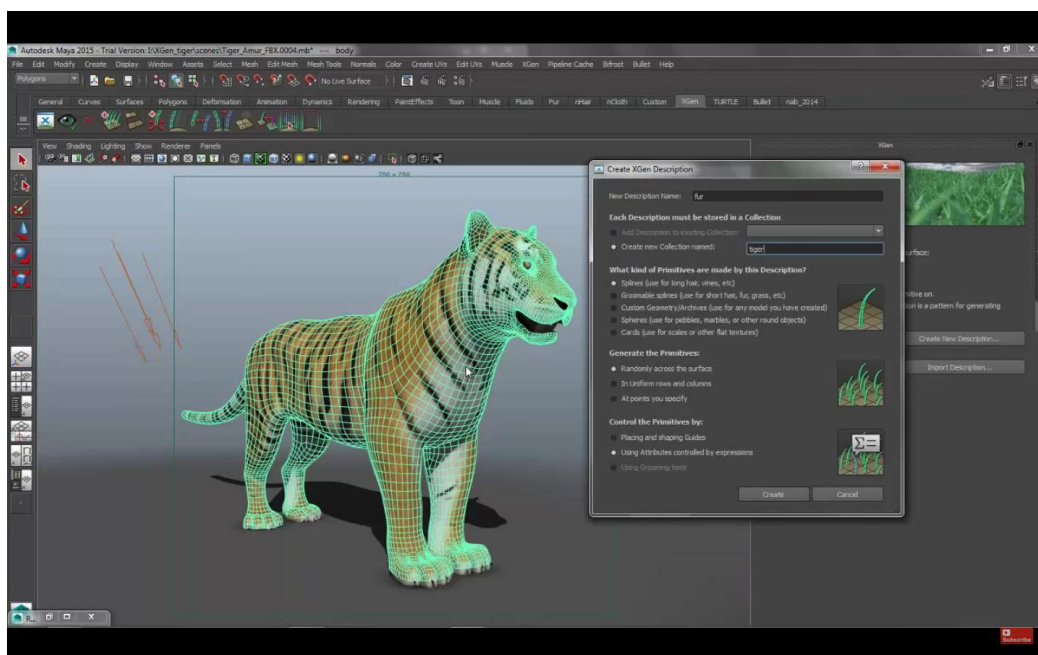


Рис.2. Процесс создания трехмерной модели в программе Maya

Программное обеспечение SolidWorks предназначено для конструирования производственных деталей инженерами промышленных предприятий. Программа получила широкое распространение в космонавтике, оборонной промышленности, автомобилестроении, медицине. Основное преимущество данного пакета, это простота в использовании и удобный понятный графический

пользовательский интерфейс (рис.3.). Возможности программы затрагивают все этапы построения модели начиная от чертежей заканчивая трехмерным объектом. Стоит отметить что SolidWorks первая программа по созданию твердотельных моделей CAD системы работающей на платформе Windows, на которой работает и по сей день. Использование пакета на других операционных системах не предусмотрено. Программа принадлежит французской компании SolidWorks Corporation, выпущена в 1995 году, последняя версия- апрель этого года. Программа имеет множество различных вариантов, также имеются несколько пакетов предназначенных для обучения студентов. Версии для обучения различны по функционалу, по времени использования, по стоимости и по количеству учебных мест. Пакет SolidWorks имеет обширный функционал и огромный спектр различных возможностей, удобных для работы инженеров и создания производственных деталей. Solidworks выбран в качестве самого производительного пакета САПР (до 95% по сравнению с другими). Большинство международных инженерных университетов и колледжей используют данный пакет для изучения CAD систем при помощи Solidworks. Вузы Канады, США, Великобритании, Индии, Австралии, Германии и многих других стран для обучения специалистов в области инженерии и промышленного дизайна используют программное обеспечение Solidworks [11].

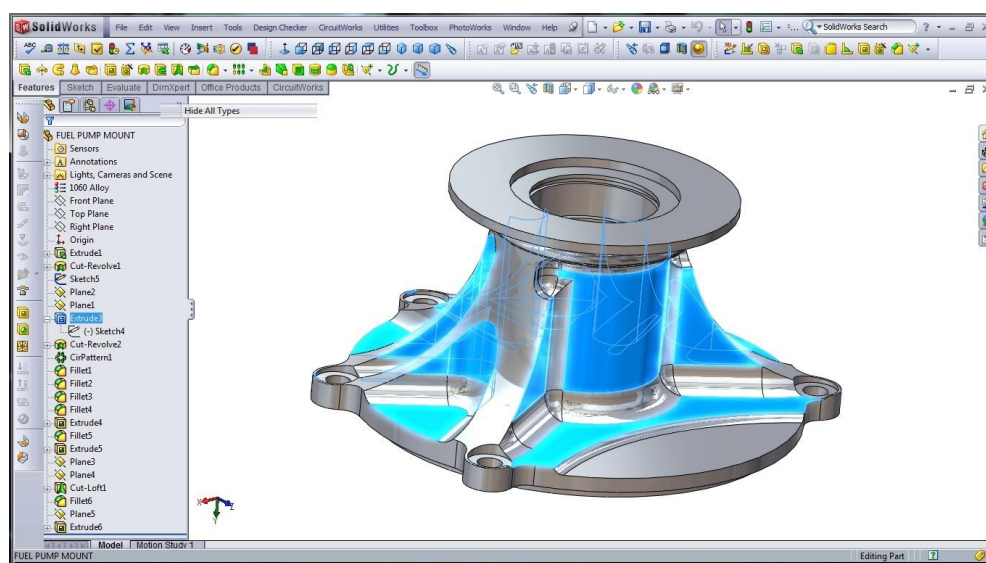


Рис.3. Процесс создания трехмерной модели в программе Solidworks

Пакет Компас-3D является одним из нескольких программ семейства Компас, созданных российской компанией Аскон. Программы предназначены для работы только на платформе Windows. Стоимость лицензионного пакета составляет от восьмидесяти тысяч рублей. Система КОМПАС-3D имеет мощные функции для управления проектами из тысячи узлов, деталей и стандартных библиотечных продуктов. Программа поддерживает все возможности трехмерных твердотельных, а также поверхностных моделей, которые стали стандартом среди программ САД среднего уровня, имеет возможность работы с облаками точек, что позволяет осуществлять 3D-сканирование. Также имеет возможность создания объектов предназначенных для реализации деталей, получаемых путем сгибания листового материала. КОМПАС-3D прост в изучении. Стоит отметить что программа основана на своем геометрическом ядре С3D и использует параметрические технологии. В состав КОМПАС-3D входит компонент Компас-График, обладающий возможностью создания чертежей. Существует несколько версий данной программы, некоторые из них предназначены не для коммерческого применения и служат средством обучения. Бесплатный пакет "КОМПАС LT" используется в образовательных учреждениях и имеет ограниченный спектр возможностей, которых достаточно для освоения программного обеспечения. "КОМПАС Home" рассчитан на самостоятельное изучение пакета и включает в себя только основные библиотеки и приложения, также имеет встроенную обучающую программу под названием "АЗБУКА КОМПАС". Также компания Аскон предоставляет возможность установки бесплатной полной версии "Учебная версия "КОМПАС-3D "" для домашнего обучения студентов, но не предназначенную для учебных заведений. Пакет может быть доступен только после регистрации на официальном сайте компании [7].

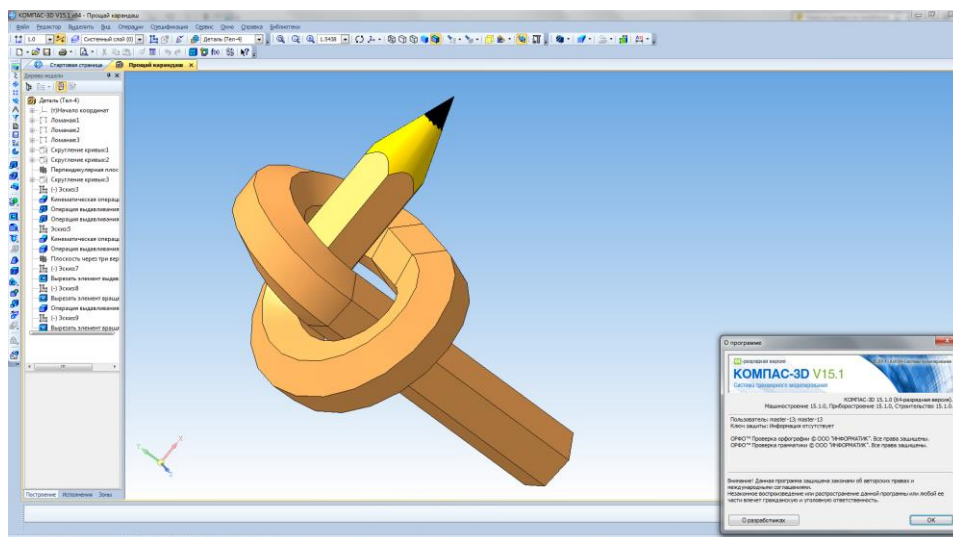


Рис.4. Процесс создания трехмерной модели в программе КОМПАС-3D

Программное обеспечение AutoCad принадлежит системе САПР, что позволяет создавать детали в двух или трех мерном формате. Программа предназначена для создания моделей объектов, применимых в промышленной отрасли. Пакет является одним из продуктов компании Autodesk, взаимодействующая с платформами Windows, OS X, iOS, Android, Windows Phone, то есть помимо работы с программой на персональном компьютере пользователь имеет возможность использовать AutoCad на своем мобильном устройстве. Программный пакет появился на рынке в 1982 году и продолжает выпускаться в новых версиях в настоящее время. Программа частично или полностью переведена на 18 различных языков, в том числе русский, который имеет полный перевод, в том числе интерфейс командной строки. Начиная с 2014 года, программа имеет возможность проектировать трехмерные объекты, поддерживая при этом твердотельное, поверхностное и полигонное моделирование. В AutoCad включена технология прямого моделирования и технология редеринга, что позволяет создавать трехмерные объекты при помощи программного кода, но отсутствует трехмерная параметризация. Также стоит отметить, что программный пакет имеет возможность взаимодействовать с 3D-принтерами и 3D-сканерами. Для расширения возможностей программы компания Autodesk и сторонние разработчики создают прикладные приложения, которые увеличи-

вают функционал работы AutoCad. Пример работы с программным обеспечением представлен на рисунке 5. Компания Autodesk предоставляет свободную студенческую версию программы для использования в образовательной сфере. Продукты этой лицензии не пригодны для коммерческих применений, для предотвращения незаконного использования студенческой версии Autodesk на каждом конечном объекте имеется пометка [7].

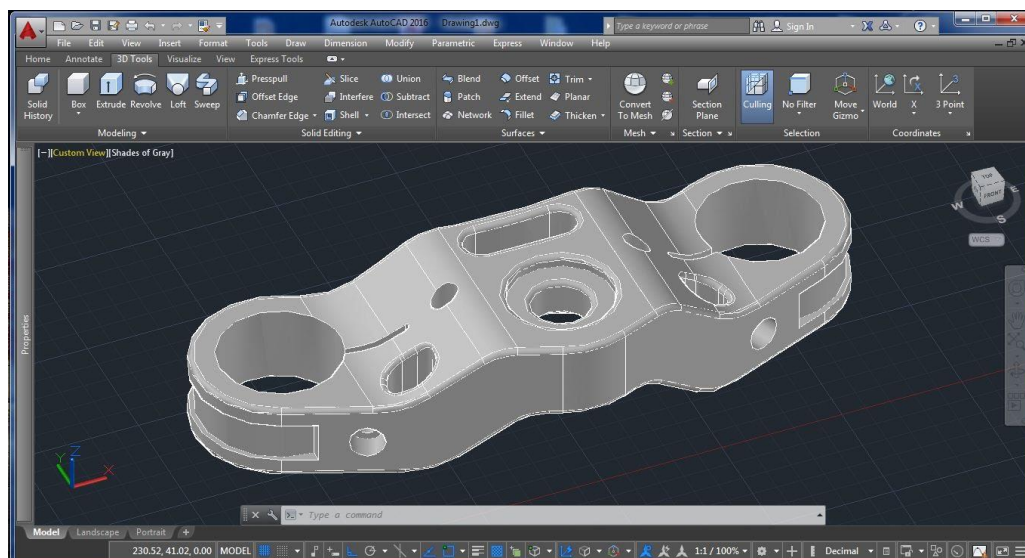


Рис.5. Процесс создания трехмерной модели в программе AutoCad

Программное обеспечение CATIA относится к типу CAD и PLM (управления жизненного цикла продукта) и позволяет создавать любой тип 3D модели применимых для широкого спектра технологических процессов. Программа имеет широкий спектр применения: машиностроение, дизайн, электрические системы, гидравлические системы, промышленность. Разработано программное обеспечение французской компанией Dassault Systèmes на языке C++. Само название CATIA расшифровывается как Computer Aided Three-dimensional Interactive Application - трехмерное интерактивное приложение с компьютерным управлением. Первый релиз CATIA осуществился в 1977, последняя же вышла в феврале прошлого года. Первоначально пакет предназначался для использования в разработке истребителя Dassault Mirage. Лицензия программы проприетарная, что позволяет владеющей компании устанавливать цену на продукт по собственному желанию. CATIA поддерживается на Unix и

Windows. Для каждой операционной системы существует индивидуальная версия программы. Пакет имеет специфические возможности создания трехмерных объектов, осуществления контроля на всех этапах построения модели и организации коллективной работы в реальном времени. Последняя функция удобна в организации связи между инженерами, находящимися на значительных расстояниях друг от друга. Пример работы с программным обеспечением продемонстрирован на рисунке 6. Также стоит отметить что программа CATIA имеет студенческую версию, которая предоставляется за девятьсот девять долларов, что является дешевле основного пакета [12].

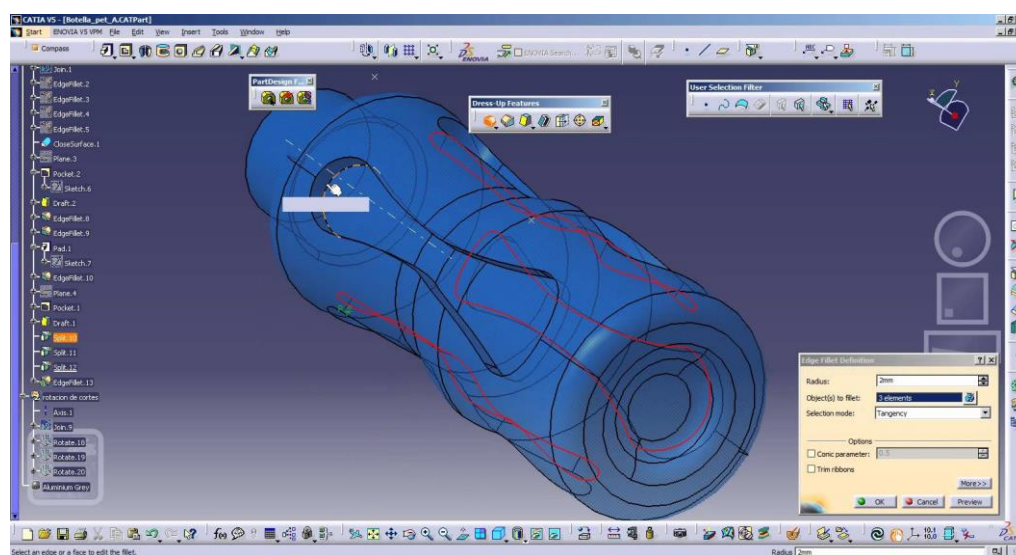


Рис.6. Процесс создания трехмерной модели в программе CATIA

Программное обеспечение Blender это система свободного автоматизированного проектирования, применяемая для моделирования трехмерных объектов. Разработана компанией Blender Foundation на языках программирования Си, C++ и Python. Blender поддерживается на платформах Linux, Microsoft Windows, macOS, Solaris, BSD и OpenBSD. Язык интерфейса переведен на несколько языков. Анонсирована в 1995 году. Последняя версия вышла в февраль 2017 года. Программное обеспечение включает в себя трехмерное моделирование, анимацию, рендеринг, постобработку и монтаж видео, и возможность создания интерактивных игр. Blender не является простым в изучении, по причине сложного объемного интерфейса. Но широкий интерфейс дает ряд воз-

возможностей: наличие двух режимов редактирования - объектный и редактирование, большое количество горячих клавиш, возможность управления рабочим пространством. Blender в первую очередь свободный, профессиональный пакет для создания трёхмерной компьютерной графики, включающий в себя средства моделирования, анимации, рендеринга, постобработки и монтажа видео со звуком, компоновки с помощью «узлов» (Node Compositing), а также для создания интерактивных игр. Интерфейс программного обеспечения представлен на рисунке 7. В настоящее время пользуется наибольшей популярностью среди бесплатных 3D редакторов в связи с его быстрым и стабильным развитием, которому способствует профессиональная команда разработчиков [10].

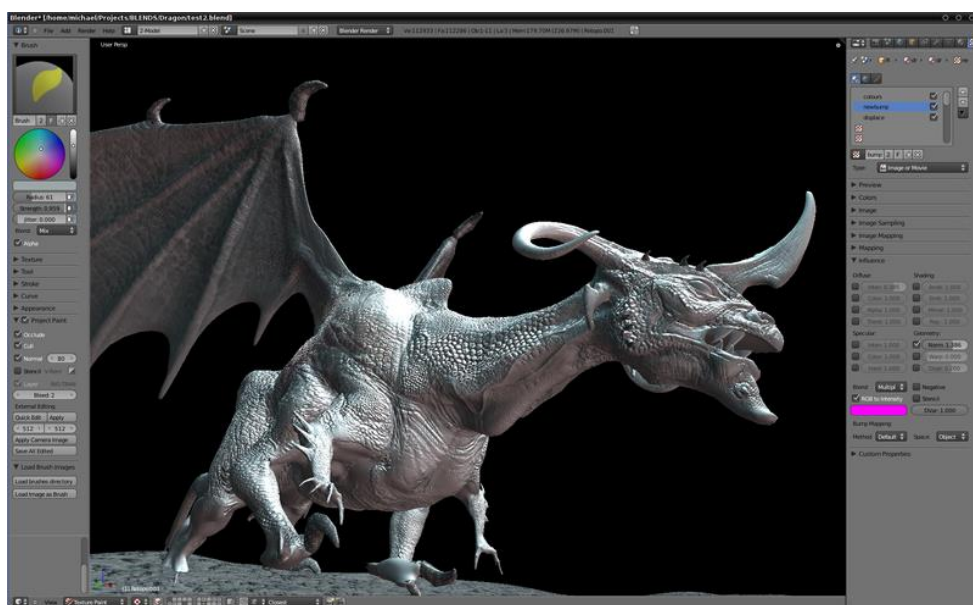


Рис.7. Процесс создания трехмерной модели в программе Blender

В заключении рассмотрим малоизвестный, но очень удобный в применении пакет OpenSCAD. Программа представляет собой бесплатное программное обеспечение, позволяющее создавать сложные твердотельные трехмерные модели, которые могут быть распечатаны в дальнейшем. Программа представляет собой компилятор, т.е. процесс моделирование происходит путем считывания написанного пользователем скрипта и создания на его основе 3D-модели, что позволяет полностью контролировать все этапы построения детали. Также готовый код в последствии может использоваться другими пользователями и кор-

ректироваться под новые задачи. Для описания модели используется простой в обучении и воспроизведении язык. Детали модели строятся при помощи примитивов, простейших фигур таких как куб, сфера или цилиндр. Сам примитив описывается программным кодом в текстовом редакторе. Для построения детали необходимо использовать операторы трансформации, которые так же описываются вручную кодом, и отвечают за изменение размера, формы, вращение и перемещение. OpenSCAD также имеет возможность обрабатывать циклы, условные операторы, математическое уравнение и вычисления, а также другую алгоритмическую информацию, которая позволяет легко создавать сложные проекты. OpenSCAD отлично подходит для создания параметрических моделей - моделей, представленных числами, которые можно легко настроить для изменения размера, формы или любого другого аспекта конструкции [27]. Пример создания трехмерных моделей представлен на рисунке 8. Если вы хотите сделать модели с точными измерениями, определенными допусками, движущимися частями или закрытыми петлями, или те, которые используют математическую или процедурную информацию, тогда OpenSCAD является одним из лучших вариантов свободного проектирования. OpenSCAD легко освоить, даже если вы никогда не писали код раньше, вы можете научиться использовать OpenSCAD довольно быстро. Каждая модель, которая экспортирует OpenSCAD, будет высококачественным, прочным, водонепроницаемым STL-файлом, который может быть понят с помощью программного обеспечения 3D-принтера. В отличие от многих программ, OpenSCAD был специально создан для создания 3D-моделей для печати поэтому, никогда не экспортирует плохую или искаженную геометрию [2].

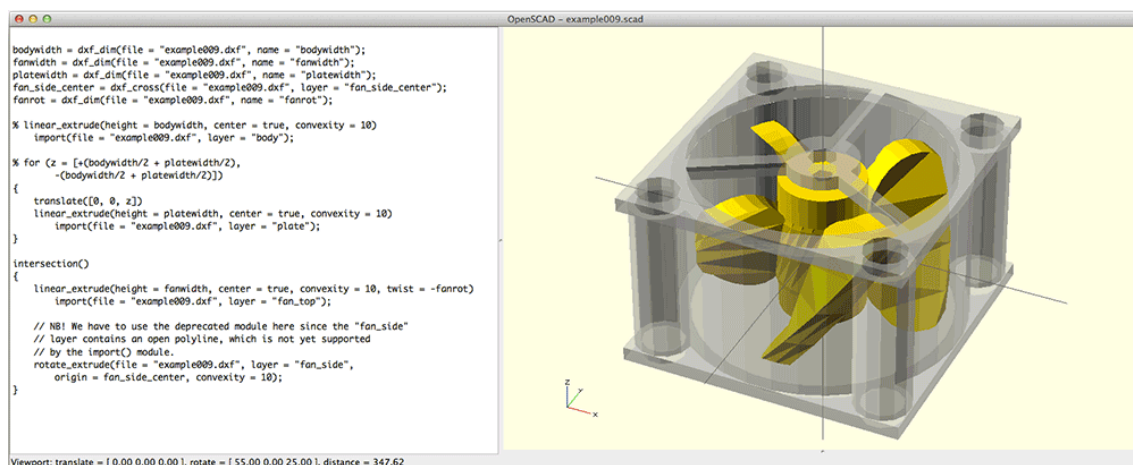


Рис.8. Процесс создания трехмерной модели в программе OpenSCAD

Рассмотрев основные представленные на рынке программные средства автоматизации проектирования, производим сравнение характеристик пакетов и приходим к выводу, что лидеры рынка являются удобными в изучении, имеют большое количество различных возможностей, но стоимость таких пакетов слишком высокая. Проанализировав свободные и дешевые программы делаем вывод, что для обучения студентов твердотельному моделированию, с учетом всех сформулированных требований к программному обеспечению являются пакеты OpenSCAD и Blender. Blender имеет больше возможностей, чем OpenSCAD, но данный фактор также затрудняет процесс обучения, делая его более затратным по времени и методически объемным. Программа OpenSCAD имеет меньше инструментов, но выполняет достаточное количество основных функций необходимых для изучения основ твердотельного трехмерного моделирования. Также стоит отметить, что программное обеспечение Blender не имеет возможности создавать твердотельные модели, что не соответствует сформулированным требованиям к программе. Необходимой для обучения студентов основам трехмерного моделирования.

Таким образом, анализ наиболее популярных программных обеспечений специализированных на создании трехмерных объектов выявил, что наиболее подходящим пакетом для использования в обучении студентов, удовлетворяю-

щим все основные требования государственных стандартов, является программа OpenSCAD [26].

Глава 2.

2.1 Описание разработанного комплекса лабораторных заданий

Целью данной выпускной квалификационной работы является разработка комплекса лабораторных заданий, направленных на изучение студентами основ твердотельного моделирования при помощи программы OpenSCAD. Данный комплекс разработан и представлен в приложении 1. Обучающая программа рассчитана на широкий круг пользователей, обладающих основными умениями работы с персональным компьютером. Методика обучения твердотельному моделированию применима, в большей степени, к студентам высшего профессионального образования различных направлений подготовки, для дополнительного образования школьников основного общего и среднего общего образования, а также учащихся, получающих среднее профессиональное образование некоторых направлений подготовки, связанных с созданием трехмерных объектов.

Программа курса направлена на освоение основных возможностей программного обеспечения OpenSCAD и самостоятельное решение, по изучению комплекса, профессиональных задач при помощи данного средства автоматизации проектирования. После прохождения обучающей программы пользователь сможет самостоятельно проектировать разнообразные трехмерные объекты различной сложности используя средство автоматизации проектирования OpenSCAD.

При помощи комплекса лабораторных заданий, направленного на изучение твердотельного моделирования, пользователь познакомится с интерфейсом программы, узнает, как работать с основными инструментами данного пакета, научится создавать модели написанием программного кода в стандартном редакторе программы.

Обучающий комплекс включает в себя введение и пять лабораторных работ, рассчитанных на пошаговое освоение программы.

Ведение описывает основные особенности программы автоматизации проектирования OpenSCAD, содержит описание актуальности изучения пакета, указания на основных пользователей, для которых создан комплекс, и цели, которые достигаются в процессе прохождения данного курса.

Каждая из пяти лабораторных работ содержит в себе постановку цели, пункты пошагового изучения пакета, задания для самостоятельного изучения и контрольные вопросы необходимые для проверки освоенного материала. Также лабораторные работы содержат иллюстрации, демонстрирующие результат выполнения некоторых заданий. Для удобства ознакомления с материалом, заголовки каждой лабораторной, цели, пункты и задания выделяются таким образом, чтобы ученик мог легко ориентироваться в тексте.

Проведем более точный анализ каждой лабораторной работы в отдельности.

Лабораторная работа номер один. Цель работы: Познакомиться с интерфейсом программы OpenSCAD. Первым пунктом описывается процесс установки пакета на персональный компьютер с подробным порядком поиска и скачивания программы с интернета. Во втором пункте пользователю пошагово объясняется процесс запуска программы, со все возможными вариантами открытия файлов, начиная с создания нового проекта, заканчивая открытием готового разработанного ранее проекта сторонним разработчиком. Последний пункт содержит в себе знакомство с интерфейсом программы, описывается назначение всех окон, инструментов, использование горячих клавиш с их комбинациями на клавиатуре. В данной лабораторной нет заданий для самостоятельной работы, но имеется указания к работе с программой и три контрольных вопроса, которые помогут самостоятельно изучить некоторые возможности программы [27].

В лабораторной работе номер два целью является: знакомство со структурой простейшей программы и создание стандартных геометрических фигур. Начиная с этой лабораторной работы, пользователь учится создавать трёхмер-

ные объекты в программе OpenSCAD. Для начала происходит описание структуры программного кода. Остальные пункты лабораторной работы знакомят с примитивами, отвечающими за создание основных фигур, из которых строятся все детали в программе OpenSCAD. Для всех примитивов имеется описание набора аргументов, отвечающих за изменения параметров объекта. Далее в работе имеются четыре задания, направленных на самостоятельное изучение, и три контрольных вопроса, позволяющие изучить аспекты темы лабораторной не упомянутые в основных пунктах работы, либо углубить полученные знания путем размышления и поиска решения поставленной задачи [2].

В третьей лабораторной работе пользователь учится изменять стандартные фигуры при помощи основных операторов. Цель работы сформулирована следующим образом: научиться изменять цвет фигур и положение моделей в пространстве. Первым пунктом идет общее описание правил, применимых для всех операторов трансформации. В остальных пунктах рассказывается о видах стандартных операторов, их аргументах, отвечающих за различные параметры, и их применении в процессе создания трехмерных моделей. Также имеется одно объемное задание на самостоятельное выполнение, необходимое для закрепления полученных знаний в данной лабораторной работе. В заключении представлено три задания, которые подходят как для самоконтроля, так и для проверки полученных знаний учителем [14].

Цель лабораторной работы номер четыре научить пользователя основным булевым операциям и познакомить с простейшими модификаторами. В пунктах содержится описание трех булевых операций: объединение, пересечение и вычитание, для наглядности используются примитивы сфера и куб, измененные операторами и представленные в виде иллюстраций в тексте. Также, имеются пункты с описанием модификаторов, позволяющих упростить процесс моделирования для пользователя. Далее имеются два задания для самостоятельной работы и четыре контрольных вопроса, содержащих два тестовых задания [27].

Последняя лабораторная работа необходима для ознакомления обучающихся с основными алгоритмами и для обучения использованию модулей при построении трехмерного объекта и закрепления всего изученного материала. Пункты содержат описания создания и применения цикла, условия и модуля, проиллюстрированных отрывком программного кода. В заключении имеются несколько заданий для самостоятельного выполнения, ориентированные на повторение всего изученного и закрепления нового материала. Контрольные вопросы содержат в себе задания различных уровней сложности [3].

В заключении комплекса представлена шпаргалка, в которой систематизированы все изученные примитивы, операции, модификаторы, алгоритмы и операторы. Также имеется библиография.

По прохождению данного комплекса, у обучающегося формируется представление о работе с программой автоматизации проектирования OpenSCAD. Пользователь сможет самостоятельно работать с данным программным пакетом, без труда моделировать трехмерные твердотельные объекты различных уровней сложности. Комплекс лабораторных работ прост в изучении и поможет овладеть искусством моделирования при помощи программы OpenSCAD обучающихся различных уровней подготовки, обладающих базовыми знаниями в работе с персональным компьютером.

Таким образом, разработанный комплекс лабораторных работ, посвященный изучению студентов основ твердотельного моделирования при помощи программного средства автоматизации проектирования пакета OpenSCAD, является удачным для обучения [26].

2.2 Методические рекомендации к разработанному комплексу лабораторных заданий

Комплекс заданий выполнен в формате самостоятельных лабораторных работ. Следовательно, обучающие выполняют работу сами без объяснения материала педагогом. Деятельность преподавателя направлена на мониторинг по-

следовательного выполнения комплекса заданий учащимися, проверку выполненных самостоятельных работ и ответов учеников на контрольные вопросы, представленных в конце каждой лабораторной, и помощь в затруднительных моментах выполнения заданий. Вопросы в процессе выполнения работ встречаются редко, так как задания имеют полное подробное описание всех этапов похождения работ. Затруднения могут вызывать выполнение самостоятельных заданий и поиск ответов на контрольные вопросы. Для облегчения деятельности педагога, ниже представлены полный список ответов и комментариев к возможным затруднительным ситуациям по каждой лабораторной работе в отдельности.

Перед началом выполнения студентами с комплекса лабораторных заданий убедитесь, что данное программное обеспечение подходит для установки на операционную систему учебных компьютеров. Если программа уже имеется на учебных компьютерах, то пункты, связанные со скачиванием и установкой программы, рекомендуются к прочтению для ознакомления с установкой данного программного комплекса для последующей работы с программой на других компьютерах. Если программа не установлена на рабочие компьютеры, то перед началом урока стоит проверить возможность доступа к глобальной сети Интернет. Если на компьютерах не установлен пакет и не имеется доступа в интернет, то заранее рекомендуется перенести на каждый рабочий компьютер приложение запускающее программное обеспечение.

Далее приведен перечень ответов на контрольные задания по каждой лабораторной работе.

Ответы на контрольные задания:

Лабораторная работа №1:

1. Произойдет перемещение камеры.
2. Произойдет изменение угла зрения вокруг центральной оси.
3. Произойдет изменение масштаба.
4. Использовать инструмент «Сбросить вид»

Лабораторная работа №2:

1. Программа выдаст ошибку и укажет на строчку, в которой пропущен символ, обозначающий закрытие примитива.
2. Радиус сферы равен пяти.
3. Произойдет наложение фигур

Лабораторная работа № 3:

1. Оператор `resize` не целесообразно применять для куба. Данный оператор изменяет стороны фигуры относительно осей *x,y,z*, но примитив куб имеет аргументы отвечающие за данные параметры.
2. Цвет фигуры при всех значениях аргумента равному 1 - белый, при 0 – черный, при 0,5 – серый.
3. Фигура будет иметь двухмерный вид.

Лабораторная работа №4:

1. Результатом кода будет деталь под буквой Г
2. Необходимо добавить оператор объединения. После исправления программа будет иметь следующий вид:

```
difference(){  
  union(){  
    sphere(r=25);  
    cylinder(h=60,r1=20,r2=0);}  
  cube([30,30,30]);}
```

3. Модификатор `#` выделяет объект не изменяя его параметры, модификатор `%` изменяет свойства объекта, делая его прозрачным.
4. Правильный ответ (а). Это происходит потому, что модификатор находится перед самым примитивом, опуская все операторы трансформации, отвечающие за изменение положения модели в пространстве.

Лабораторная работа №5:

1. Исправленная программа выглядит следующим образом:

```
difference(){
```

```

color("green")
rotate([0,90,0])
cylinder(h=6,d=70);
rotate([22,0,0]) resize([6,32,32]) test();
translate([0,17,17]) test();
translate([0,-17,17]) test();
translate([0,17,-17]) test();
translate([0,-17,-17])test();
translate([0,-24,0]) test();
translate([0,0,-24]) test();
translate([0,24,0]) test();
translate([0,0,24]) test();}
module test(){
for(i=[0:1:7]){
rotate([i*45,0,0]){
cylinder(h=10,d1=6,d2=0);
sphere(3);}}}}

```

2. После правильного исправления программного кода программы должна получиться следующая модель:

Далее приведен перечень вариантов ответов на самостоятельные задания по каждой лабораторной и некоторые указания на которые стоит обратить внимание при проверке выполненных работ.

Ответы на самостоятельные задания:

Лабораторная работа №2

1. Создайте куб сторонами [20, 30, 50]

Решение: (рис.9)

Код программы:

```
cube([20, 30, 50]);
```

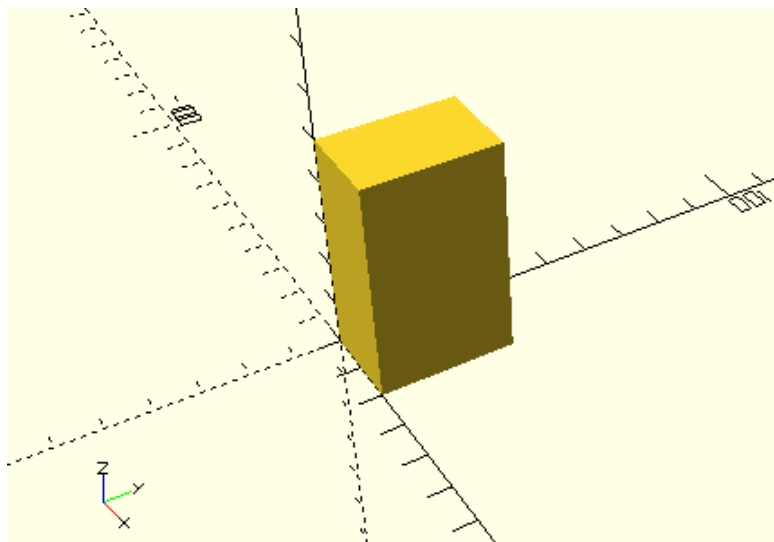


Рис.9. Результат программы. Лабораторная №1 Задание №1

2. Создайте модель цилиндра, имеющую высоту 10 и диаметр 6 и расположение центром в начале осей координат.

Решение: (рис.10)

Код программы:

```
cylinder(h=10,d=6,center=true);
```

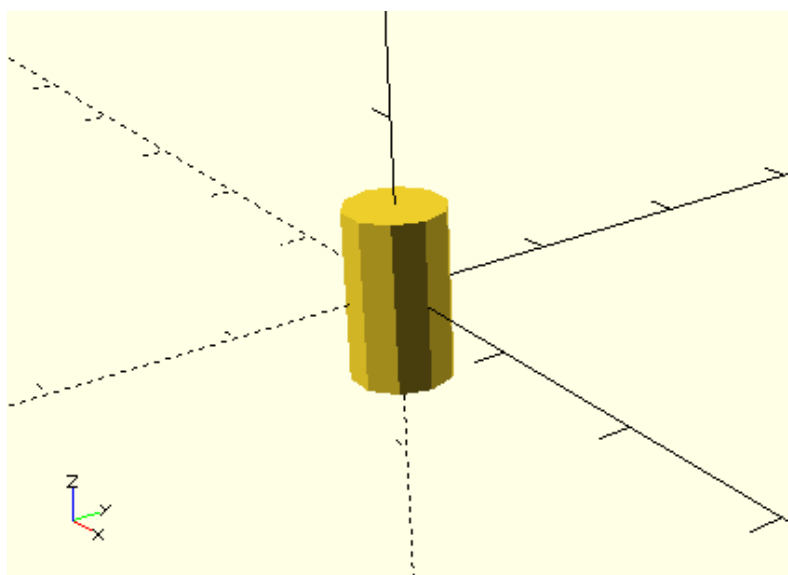


Рис.10. Результат программы. Лабораторная №1 Задание №2

3. Изменяя количество полигонов, высоты и радиуса в цилиндре, измените фигуру из задания 2 таким образом, чтобы получился куб со сторонами 10

Решение: (рис.11)

Для того чтобы определить каким диаметром должен быть куб воспользуйтесь теоремой

Пифагора. При правильных расчетах диаметр будет равен 30.

Код программы:

```
$fn=4;
```

```
cylinder(h=20,d=30,center=true);
```

Проверить правильность построенной учеником фигур возможно при повороте куба относительно координаты Z на 45 градусов: rotate([0,0,45]) Добавляется оператор между двумя строчками программы без точки с запятой в конце.

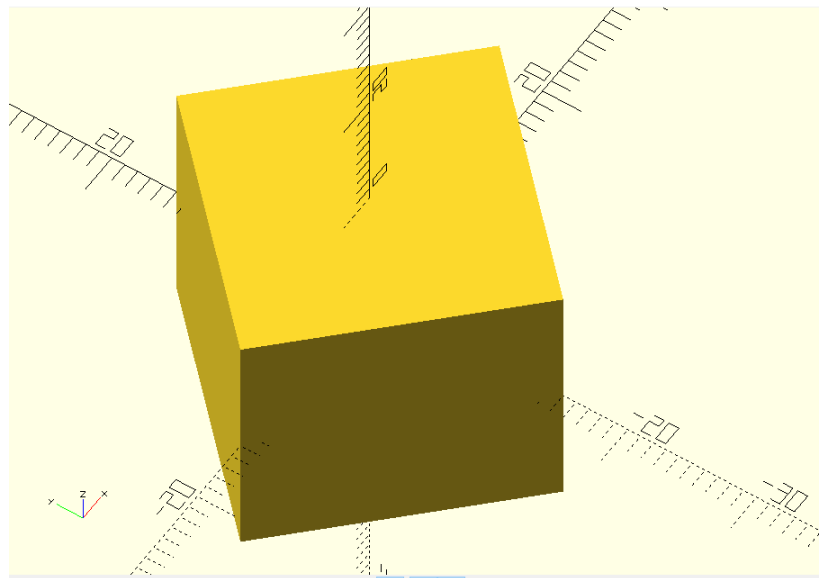


Рис.11. Результат программы. Лабораторная №1 Задание №3

4. Постройте шестиугольную перевернутую пирамиду высотой 8 и радиусом основания 7

Решение: (рис.12)

Код программы:

```
$fn=6;
```

```
cylinder(h=8,r0=7,r1=0);
```

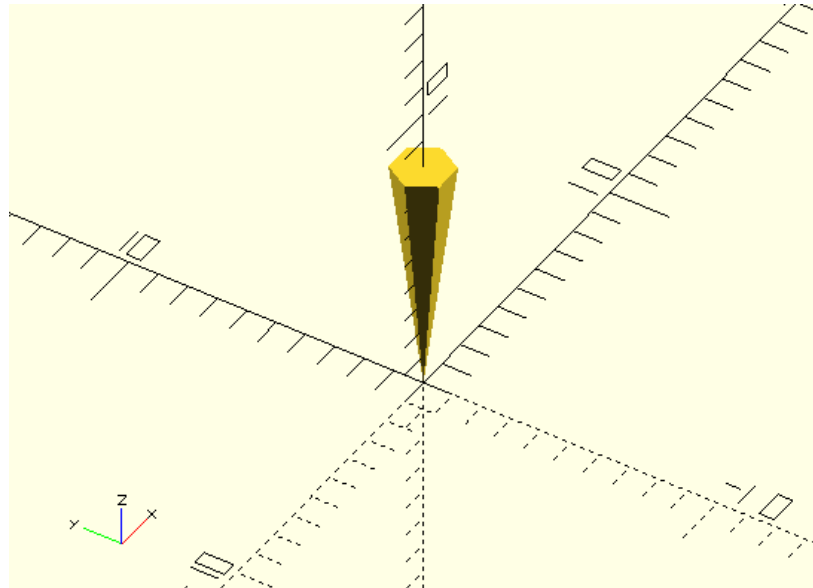


Рис.12. Результат программы. Лабораторная №1 Задание №4

Лабораторная работа №3

1. Используя полученные знания создайте 3-D модель новогодней елки

Части кода программы с примера:

Данный оператор позволит контролировать размер всей модели для того чтобы фигурка вошла в коробку:

```
resize([30,30,30]){
```

Пример создания одного яруса елки

```
$fn=3;
```

```
translate([0,0,3.2]){
```

```
cylinder(h=2, r1=1, d2=0);
```

```
rotate([0,0,180])
```

```
cylinder(h=2, r1=1, d2=0);}
```

Пример создания елочной игрушки:

```
color("white")
```

```
scale([20,20,0.5])
```

```
sphere($fn=64);
```

Подставку рекомендуется создать через сферу и поменять размер. Пример создания подставки:

```
color("white")
scale([20,20,0.5])
sphere($fn=64);
```

Пример создания звезды:

Создаем три цилиндра, и поворачиваем каждый на равное расстояние друг к другу (рис.13)

```
rotate([0,0,100]){
cylinder($fn=64, h=3, d1=2, d2=0);
rotate([0,72,0])
cylinder($fn=64, h=3, d1=2, d2=0);
rotate([0,144,0])
cylinder($fn=64, h=3, d1=2, d2=0);
rotate([0,216,0])
cylinder($fn=64, h=3, d1=2, d2=0);
rotate([0,288,0])
cylinder($fn=64, h=3, d1=2, d2=0);}
```

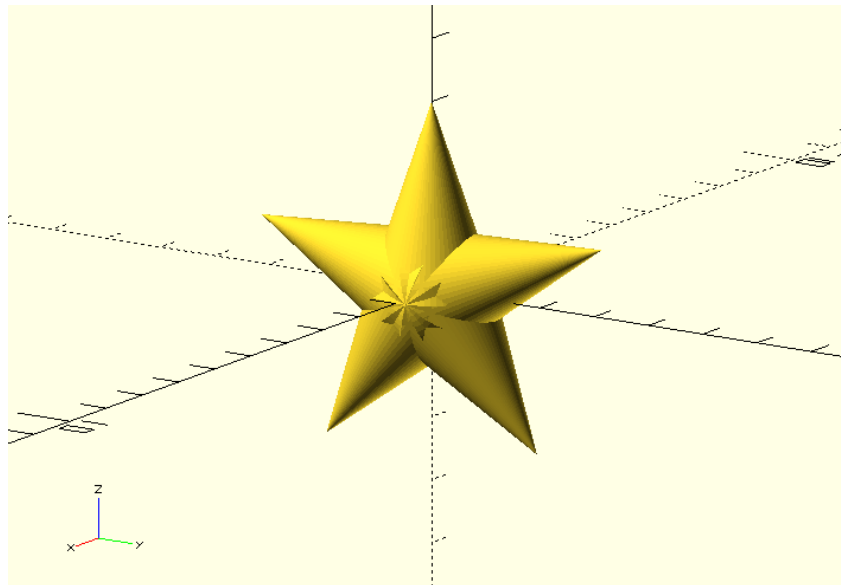


Рис.13. Моделирование звезды тремя цилиндрами

Лабораторная работа №4

1. Постройте деталь со значка OpenSCAD

Для создания данной детали понадобятся примитивы, отвечающие за сферу и цилиндры. Для начала необходимо сгладить края фигур увеличением количества полигонов $fn=64$. Далее вычитаем из сферы три цилиндра. Один из цилиндров выделяем модификатором #

2. Построить модель игрального кубика

Строим куб, сглаживаем края при помощи оператора пересечения, как указано в лабораторной работе. Далее вычитаем сферы из каждой стороны куба.

Пример части кода программы: (рис.14)

```
difference(){  
  intersection (){cube([50,50,50],center=true);  
    sphere(r=34);}  
  translate([25,0,0])  
    sphere(r=3);}  
}
```

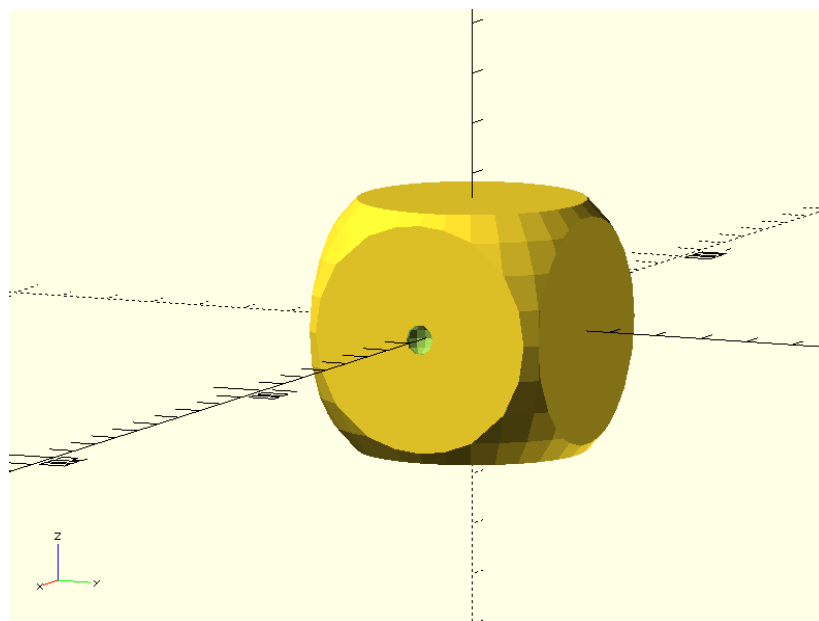


Рис.14. Модель игрального кубика

Лабораторная работа №5

1. Смоделируйте цилиндр с градиентом

Идея решения заключается в создании нескольких цилиндров расположенных друг за другом и меняющих цвет на более темный тон.

Программный код:

```
for (i=[0:5:100]){  
  translate([20+i,0,0])  
  rotate([0,90,0])  
  color([i/100,0,0])  
  cylinder(h=20,d=30);}
```

2. Постройте модель решета

Обучающимся предстоит построить модель решета, состоящую из ручки и самого решета. Одним из вариантов создания решетчатой части является удаление из полусферы цилиндрической формы отверстий. В данной программе применяется операция вычитания для создания полусферы и для удаления отверстий.

Таким образом, при помощи данных рекомендаций педагог сможет контролировать и помогать обучающимся в процессе освоения данного комплекса лабораторных работ, посвященных изучению студентов основ твердотельного моделирования при помощи программного средства автоматизации проектирования пакета OpenSCAD.

2.3 Результаты апробации

Комплекс лабораторных заданий был опробован студентами третьего курса направлений подготовки педагогическое образование профиль «информатика» и информационные системы и технологии в рамках предмета «Компьютерный дизайн». Студенты выполнили комплекс заданий за шесть академических часа. Все задания студентами пройдены успешно. Самостоятельные задания выполнены, что иллюстрирует понимание и успешное освоение программы комплекса лабораторных работ. По завершению прохождения программы обучающимися заполнена анкета, позволяющая сформировать мнение студентов о пройденном комплексе. На основе полученных результатов опроса формируется следующее мнение о программном обеспечении: Все опрошенные отзыва-

лись о программе положительно. Программное обеспечение вызывало положительные эмоции тем, что содержит в себе необходимость создания трехмерной модели путем написания программного кода и является простатой в изучении. Установка программы не вызвала затруднения. На начальном этапе работы с программой у студентов возникли трудности с расположением и перемещением камеры, но в процессе дальнейшего прохождения лабораторных заданий данная проблема исчезла. Опрошенные поставили программному обеспечению от 7 до 10 баллов по десятибалльной шкале оценивания. Студенты обосновали снятие баллов отсутствием возможности создания моделей без использования программного кода. Одни сочли программу слишком ограниченной в возможностях, другие наоборот посчитали легкость освоения программы большим плюсом в обучении основам программных средств автоматизации проектирования.

Опрошенные оценили не только программное обеспечение, но и комплекс лабораторных заданий при помощи которого они освоили данный пакет. Студенты сочли лабораторные работы удобными и понятными в изучении. Текст лабораторных, по мнению опрошенных, является читабельным по причине того, что состоит из удобных для чтения шрифтов, имеет благоприятный для чтения размер и присутствуют заголовки, отделяющие между собой пункты лабораторных работ. Опрос выявил, что у студентов не возникло трудностей в освоении программного пакета, а если появлялись сложности, то по причине невнимательного прочтения текста лабораторных заданий. Студенты сочли комплекс лабораторных заданий грамотно составленным, обосновав это тем, что изучение материала осуществлено пошагово и имеет как легкие, так и трудные задания. Всем опрошенным больше всего понравилось выполнять самостоятельные задания. Студенты отметили, что выполнять лабораторные работы было интересно и просили увеличить количество заданий. Все опрошенные считают, что по прохождению данного курса смогут самостоятельно моделировать трехмерные объекты в программе OpenSCAD на базовом и среднем уровнях.

Таким образом, по прохождении апробации формируется вывод, что программное обеспечение OpenSCAD является простым в обучении средством автоматизации проектирования пригодным для обучения студентов твердотельному моделированию. Разработанный комплекс лабораторных работ, позволяющий познакомиться с основами программного пакета OpenSCAD, удобен, понятен и интересен в процессе обучения всем студентам, прошедшим данную программу обучения.

Заключение

В ходе реализации данной выпускной квалифицированной работы на основе сформулированной цели и задач выполнены следующие этапы:

1. В результате анализа федеральных государственных образовательных стандартов выявлена потребность в изучении моделирования.
2. Анализ федеральных государственных образовательных стандартов различных ступеней образования позволил выявить актуальность изучения трехмерного моделирования.
3. Анализ представленных на рынке программных средств автоматизации проектирования позволил выявить программное средство OpenSCAD как наиболее пригодную для обучения студентов основам твердотельного моделирования.
4. Разработан комплекс лабораторных заданий, направленный на изучение основ твердотельного моделирования при помощи выбранного средства автоматизации проектирования.
5. Подготовлены методические рекомендации по использованию разработанного комплекса лабораторных заданий в учебном процессе.

Успешная апробация, на студентах высшего учебного заведения, подтвердила правильность выбора программы OpenSCAD в качестве средства автоматизации проектирования, предназначенного для обучения основам твердотельного моделирования и выявила, что составленный комплекс лабораторных заданий является интересным, познавательным и полным для обучения студентов основным возможностям пакета OpenSCAD.

На основе выше изложенного формируется вывод, что все поставленные задачи выполнены; цель квалификационной работы достигнута. Следовательно, работа носит законченный характер.

Список литературы

1. 3D печать [Электронный ресурс] // sites.google.com
URL: <https://sites.google.com/site/ifizmat/3d-print/lab-01> (дата обращения: 16.04.2017).
2. OpenSCAD [Электронный ресурс] // reprap.org
URL: <http://www.reprap.org/wiki/OpenSCAD> (дата обращения: 20.04.2017).
3. OpenSCAD CheatSheet [Электронный ресурс] // www.openscad.org
URL: <http://www.openscad.org/cheatsheet/>
4. Абдурахманова З. К. Компьютерное моделирование в системе образования // Вестник социально-педагогического института. 2011.
5. Борисенко И. Г. Информация в образовательной системе: особенности социально-философского исследования. // Вестник ИрГТУ. – 2012. – №4(63). – 347 с. Стр. 298–302.
6. Борисенко И.Г., Головина Л.Н., Володина Д.Н. Проблемы инженерного образования. Повышение эффективности самостоятельной работы // Вестник Иркутского государственного технического университета. – 2014. – № 1 (84). – С. 171- 175.
7. Камалидинова Э.Р., Рожина И.В. Программы построения трехмерных графических изображений // Информационно-коммуникационные технологии в образовании: межвузовский сборник научных работ. Екатеринбург: Урал. гос. пед. ун-т, 2015.
8. Карпов Ю. Г. Имитационное моделирование систем. Введение в моделирование с AnyLogic 5 / Ю. Г. Карпов. СПб. : БХВ Петербург, 2006.
9. Королев А. Л. Компьютерное моделирование. Лабораторный практикум. Москва: БИНОМ. Лаборатория знаний, 2012. Королев А. Л. Компьютерное моделирование в образовании (год не нашла, но вроде 2011)
10. Кронистер Дж. Blender Basics [электронное издание]. 4 изд. 2010. 153 с.
11. Ли Дж., Уэр. Б., Трёхмерная графика и анимация. 2-е изд. М.: Вильямс, 2002. 640 с.

- 12.Мэрдок К. Autodesk 3Ds Max 2013. Библия пользователя. М.: «Диалектика», 2013. 816 с.
- 13.Никулин Е. А., Компьютерная геометрия и алгоритмы машинной графики. СПб: БХВ-Петербург, 2003. 560 с.
- 14.Официальный сайт OpenSCAD [Электронный ресурс] // www.openscad.org
URL:<http://www.openscad.org/documentation.html> (дата обращения: 19.04.2017).
- 15.Прахов А.А. Blender: 3D-моделирование и анимация. Руководство для начинающих. СПб.: БХВ – Петербург, 2009. 254 с.
- 16.Приказ Минобрнауки РФ "Об утверждении и введении в действие федерального государственного образовательного стандарта высшего профессионального образования по направлениям подготовки" от 22.12.2009 № 788
- 17.Приказ Минобрнауки РФ "Об утверждении профессионального стандарта "Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)" от 18.10.2013г № 544
- 18.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 16.03.01 техническая физика (уровень бакалавриата) " от 12.03.2015 г № 204
- 19.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 27.03.04 управление в технических системах (уровень бакалавриата) " от 20.10.2015 г № 1171
- 20.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 12.03.01 приборостроение (уровень бакалавриата) " от 03.09.2015 г № 959

- 21.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 11.03.03 конструирование и технология электронных средств (уровень бакалавриата) " от 06.03.2015 г № 178
- 22.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 16.04.01 техническая физика (уровень магистратуры) " от 21.11.14 г № 1486
- 23.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 24.04.01 ракетные комплексы и космонавтика (уровень магистратуры) " от 06.03.2015 г № 164
- 24.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.04.02 информационные системы и технологии (уровень магистратуры) " от 30.10.2014 г № 1402
- 25.Приказ Минобрнауки РФ "Об утверждении федерального государственного образовательного стандарта основного общего образования" от 29.12.2014 г № 1644
- 26.Пяткова А.Н., Алексеевский П.И. Обучение студентов твердотельному моделированию с использованием свободного ПО // Информационно-коммуникационные технологии в образовании: межвузовский сборник научных работ. Екатеринбург: Урал. гос. пед. ун-т, 2017.
- 27.Руководство пользователя по OpenSCAD [Электронный ресурс] // ru.wikibooks.org
URL:ru.wikibooks.org/wiki/Руководство_пользователя_по_OpenSCA (дата обращения: 10.05.2017).

- 28.Соловьева В.В., Черенков П.С., Черкез Г.Б. Компьютерная графика для художников и дизайнеров. История развития компьютерной графики. // Учебно-методическое пособие. Нальчик; «Нальчик», 2001. 39 с.
- 29.Столбова И.Д. Формирование профессионально-ориентированных компетенций при инновационных технологиях предметного обучения в высшей школе / И.Д. Столбова, В.А. Лалетин, Е.С. Дударь // Информационные технологии в науке, образовании, телекоммуникации, бизнесе: труды 34 Межд. конф. / Приложение к журналу «Открытое образование». – Украина, Крым, Ялта-Гурзуф, 2007. – С. 256–257.
- 30.Учебник Инженерная и компьютерная графика: // Б.Г. Миронов, Р.С. Миронова, Д.А. Пяткина, А.А. Пузиков. – 5-е изд., стер. – М.: Высш. шк., 2006. – 334 с.
- 31.Фридман Л.М. Наглядность и моделирование в обучении // Новое в жизни, науке, технике. Сер. "Педагогика и психология". 2008. №6.

Приложения

Приложение 1.

Введение

3D-печати является перспективной сферой развития современных технологий. Её внедряют и применяют повсеместно: начиная с создания деталей для станков, заканчивая печатью донорских органов в медицине. Для того, чтобы получить конечный напечатанный продукт необходимо уметь проектировать его модель в электронном виде. В настоящее время на рынке представлено множество различных программ для создания трехмерных моделей. Проведя анализ всех факторов, влияющих на выбор программного обеспечения, применимого для обучения твердотельному моделированию, наиболее подходящей является программа OpenSCAD.

OpenSCAD – бесплатное программное обеспечение, позволяющее создавать твердотельные трехмерные модели. Программа представляет собой компилятор, т.е. процесс моделирование происходит путем считывания написанного пользователем скрипта и создания на его основе 3D-модели, что позволяет полностью контролировать все этапы построения детали. Также готовый код в последствии может использоваться другими пользователями и корректироваться под новые задачи. Для описания модели используется простой в обучении и воспроизведении язык.

Программа курса направлена на освоение основных возможностей программного обеспечения OpenSCAD и решение профессиональных задач при помощи данного средства автоматизации проектирования.

Данный комплекс лабораторных заданий рассчитан на широкий круг пользователей ПК. Методика обучения твердотельному моделированию применима к студентам высшего профессионального образования различных направлений подготовки, также для дополнительного образования школьников основного общего и среднего общего образования.

В процессе освоения данного комплекса лабораторных заданий вы научитесь пользоваться основными инструментами OpenSCAD и писать код, позволяющий создавать различные трехмерные модели.

Желаем удачи!

ЛАБОРАТОРНАЯ РАБОТА № 1

ЗНАКОМСТВО С OpenSCAD

Цель: Познакомиться с интерфейсом программы OpenSCAD

Установка OpenSCAD

Для начала самостоятельно установим программу на компьютер. Скачать свободное программное обеспечение OpenSCAD можно с официального сайта *openscad.org* во вкладке downloads (рис. 1) Программа подходит для всех часто встречающихся платформ. Перед скачиванием убедитесь, что программа предназначена именно для вашей операционной системы.

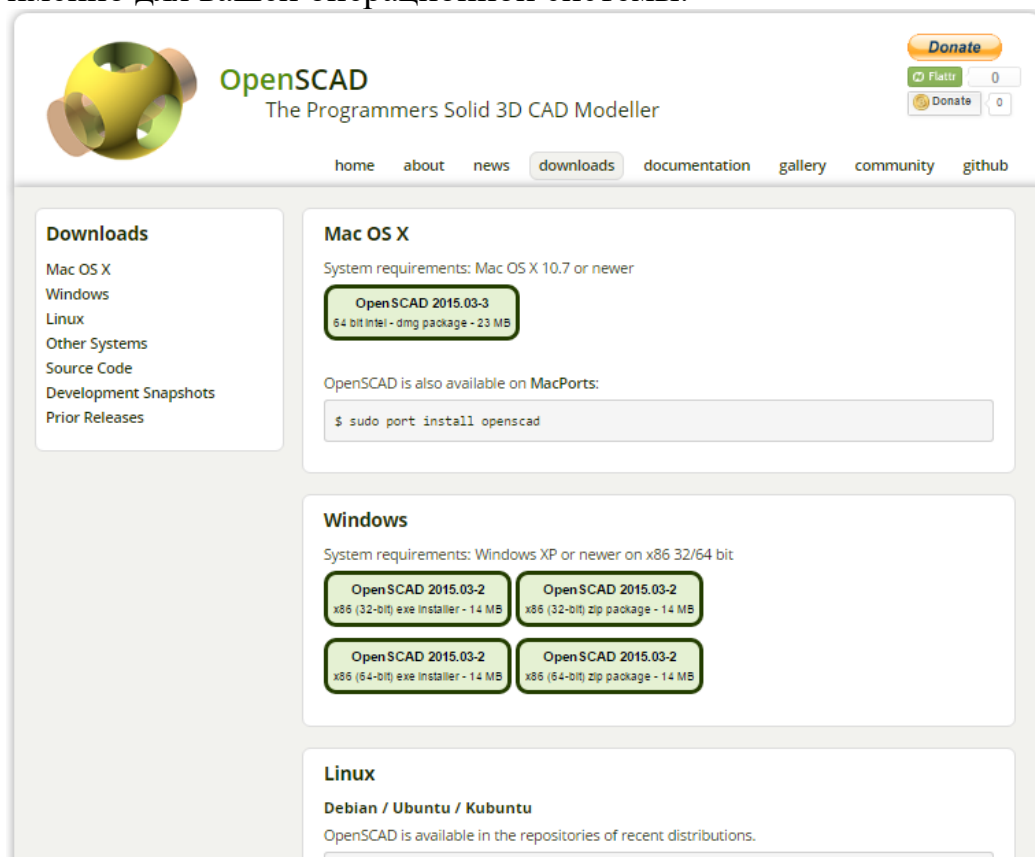



Рис.1. Официальный сайт

Разархивируйте скаченный файл и запустите приложение  **openscad.exe**. Обратите внимание, что для работы программы не требуется установка!!! Последующие запуски программы будут производиться через это файл.

Запуск OpenSCAD

После запуска программы открывается начальное окно «Добро пожаловать в OpenSCAD» (рис.2), с помощью которого мы можем создавать новый проект, открыть созданную ранее сторонним программистом или недавно использованную в работе модель. Также имеется возможность получить справочный материал или посмотреть пример программного кода какой-либо детали из архива. Попробуйте открыть различные готовые проекты из пункта «Примеры».

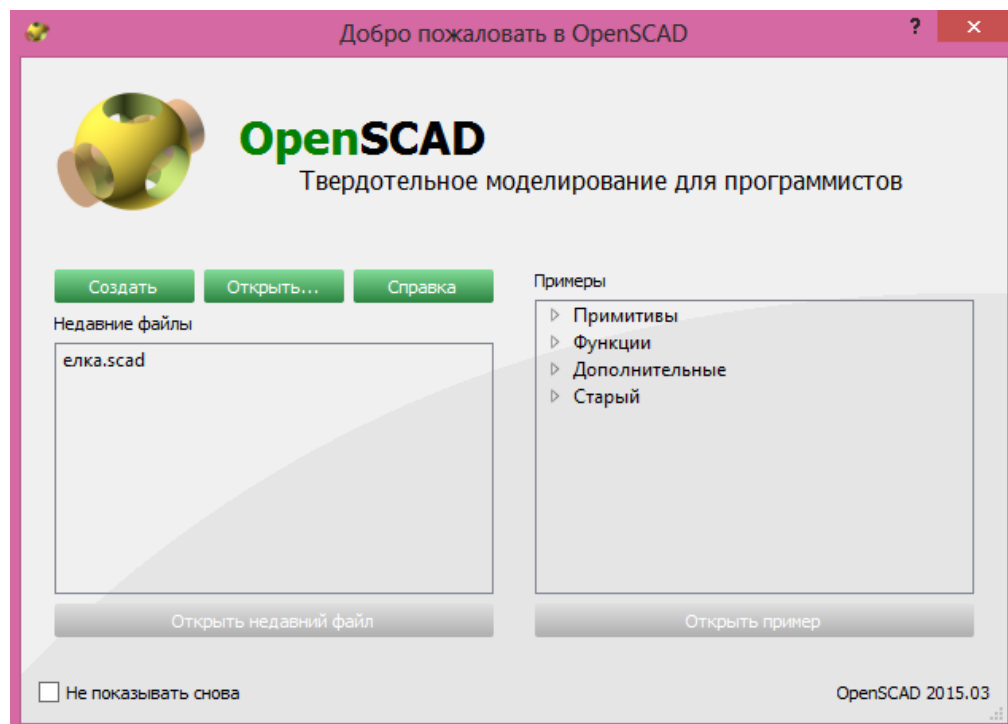


Рис.2. Окно «Добро пожаловать»

Интерфейс программы OpenSCAD имеет следующий вид (рис.3): Слева располагается Текстовый редактор, в котором описывается создаваемая модель. Справа Окно привью модели. Шкалы осей координат измеряются в миллиметрах. Снизу имеется Консоль, в которую выводятся ошибки или предупреждения. На редакторе и окне привью располагаются инструменты, необходимые для удобства пользования программы.

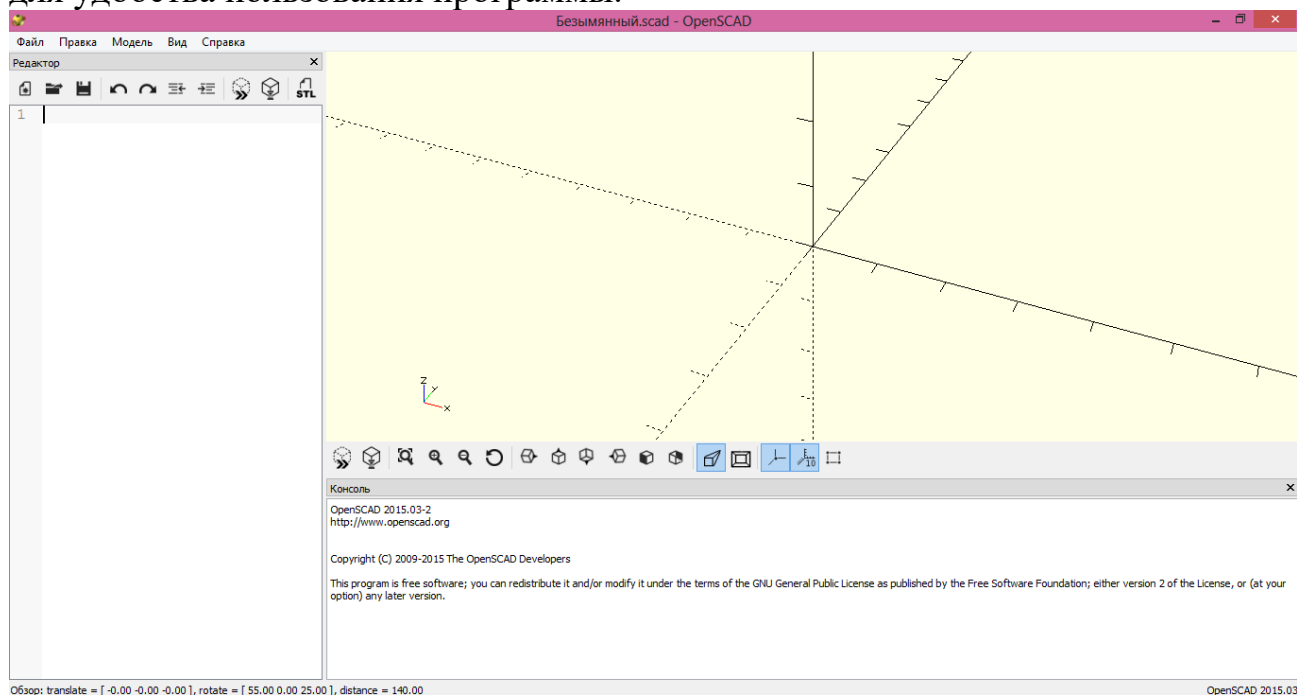




















Рис.3. Рабочее окно программы

На редакторе и окне привью располагаются инструменты, необходимые для удобства пользования программы.

	Создать новый проект Ctrl+N
	Открыть существующий проект Ctrl+O
	Сохранить проект Ctrl+S
	Отменить действие Ctrl+Z
	Повторить отмененное действие Ctrl+Shift+Z
	Убрать отступ Ctrl+Shift+I
	Добавить отступ Ctrl+I
	Предпросмотр F5
	Рендеринг F6
	Экспортировать в StL
	Показать всё
	Увеличить Ctrl+] Уменьшить Ctrl+[масштаб
	Сбросить вид
	Стороны просмотра модели Справа Ctrl+7 Сверху Ctrl+4 Снизу Ctrl+5 Слева Ctrl+6 Спереди Ctrl+8 Сзади Ctrl+9
	Вид просмотра модели Перспектива Ортодоксальная проекция
	Показать оси
	Показать метки масштаба Ctrl+2
	Показать ребра модели Ctrl+1

Для того, чтобы посмотреть на созданную после описания кода модель **НЕ ЗАБУДЬТЕ** запустить предварительный просмотр нажав кнопку предпросмотр или клавишу F5!!!

Контрольные вопросы:

1. Что произойдет в окне привью если зажать правую клавишу мыши и переместить её?

2. Что произойдет в окне привью если зажать левую клавишу мыши и переместить её?
3. Что произойдет в окне привью если менять положение колесика мыши?
4. Как вернуть камеру в исходное положение?

ЛАБОРАТОРНАЯ РАБОТА № 2

ПРОСТЕЙШАЯ ПРОГРАММА В OpenSCAD

Цель: Познакомиться с структурой простейшей программы. Научиться моделировать геометрические фигуры.

Структура программы в OpenSCAD

Детали в OpenSCAD создаются при помощи примитивов, которые отвечают за создание трехмерных геометрических фигур. Стандартные примитивы отвечают за моделирование куба, сферы и цилиндра. Используя аргументы примитивов можно менять размер модели и получать новые фигуры: параллелепипед, пирамиду, многоугольники и конус. Созданная фигура строится в начале координат.

Создание куба в OpenSCAD

Попробуем построить модель куба стандартного размера. По умолчанию аргументы фигуры равны одному миллиметру. Для того чтобы создать куб в редакторе прописываем примитив, отвечающий за создания куба: `cube()`; Обратите внимание, что после примитива ставится `;`. Это необходимо для того, чтобы объявить о завершении операции. Для просмотра получившейся модели нажмите кнопку предпросмотр или клавишу F5. Если все сделано правильно, то в окне привью появится модель куба стандартных размеров (рис.4).

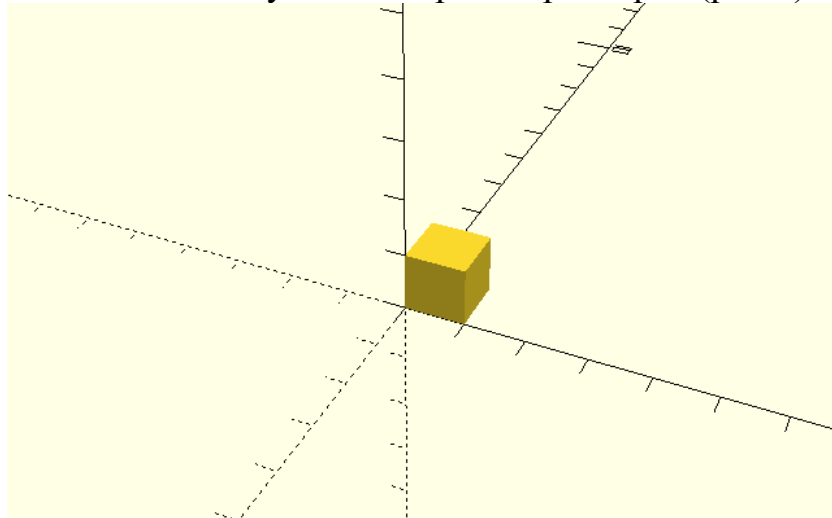


Рис.4. Модель куба стандартного размера

Примитивы имеют аргументы, отвечающие за размер фигуры. Попробуем изменить наш куб. Данную операцию можно выполнить двумя способами либо задать общий размер стороны куба, либо для каждой стороны в отдельности. Полученный результат будет одинаковый, но запись аргументов примитивов разной. Установим размер сторон 10мм каждым из способов.

1. Для первого случая имеется аргумент `size`, к которому присваивается размер общей стороны, отвечающий за размер общей стороны.
`cube(size=10);`

2. Для второго случая существуют аргументы x, y, z отвечающие за размер стороны находящийся относительно координат X, Y, Z соответственно. Вместо данных аргументов записывается размер.
`cube([10,10,10]);`

Если вы запишите оба примитива в одну программу, то создастся два куба, и они между собой совпадут. Поэтому для наглядности создайте сначала куб первым способом. Проверьте результат. Закомментируйте примитив с помощью двойного слеша `//`. Запустите программу, чтобы проверить что ничего не моделируется в окне привью. Создайте модель вторым способом. Проверьте результат.

Как вы уже заметили, по умолчанию фигура строится одним из углов в начале осей координат. Для того, чтобы совместить центр куба с началом осей (рис.5) существует аргумент `center`. По умолчанию аргумент имеет значение `false`, поэтому для решения данной задачи присваиваем `center` значение `true`. Аргументы внутри примитивов записываются через запятую. Выглядеть это будет следующим образом:

```
cube(size=10, center=true);
```

```
cube([10,10,10], center=true); или cube([10,10,10], true);
```

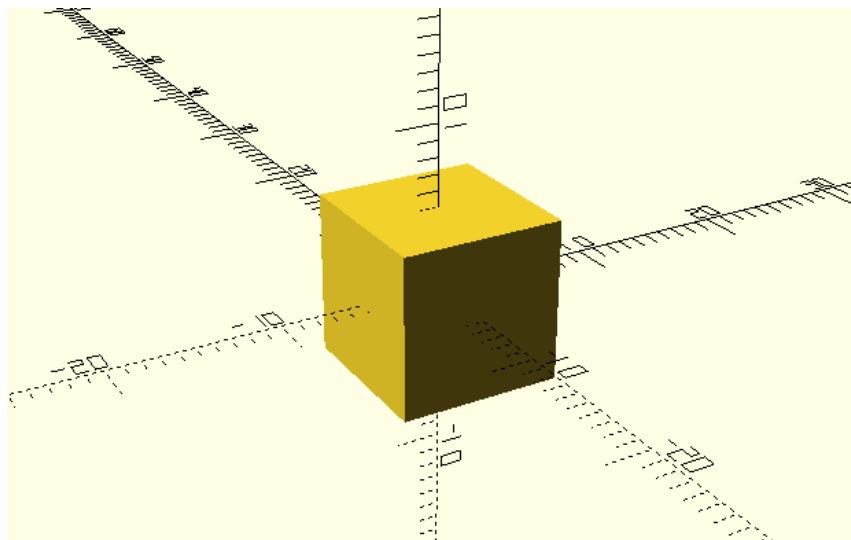


Рис.5. Модель куба, расположенная в центре осей координат

Создание цилиндра и конуса в OpenSCAD

За создание трехмерной модели цилиндра в программе OpenSCAD отвечает примитив `cylinder()`; аргументами которого будут h -высота- радиус, d -диаметр и `center`-расположение фигуры в центре осей. Если вы решили задавать

параметры через радиус, то диаметр прописывать не нужно. Аналогично наоборот, если вы задаете диаметр, то размер радиуса указывать не нужно.

Для создания конуса используется известный нам примитив `cylinder`, аргументы которого будут отвечать за h -высоту $r1$, $r2$ - радиус нижнего и верхнего основания соответственно, $d1, d2$ -диаметр нижнего и верхнего основания и `center`-расположение фигуры в центре осей. Чтобы фигура имела вид конуса (рис.6) задаем аргументу, отвечающему за верхний радиус или диаметр значение 0.

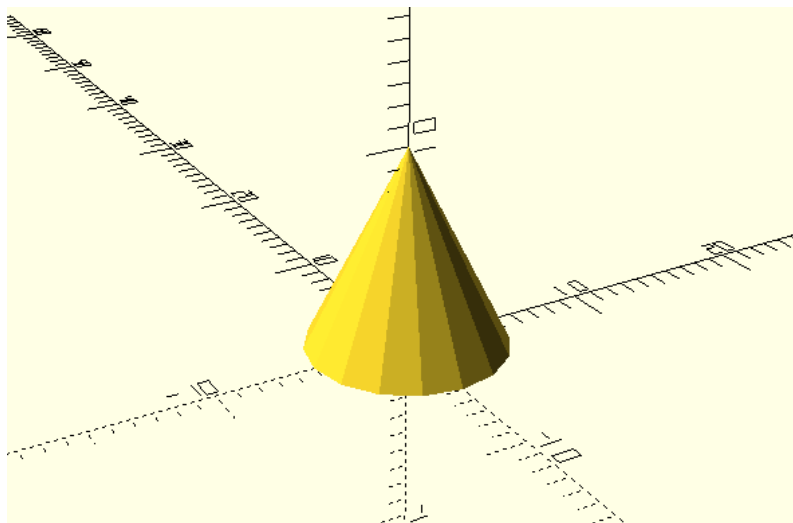


Рис.5. Модель конуса

Создание сферы в OpenSCAD

Для создания сферы используется примитив `sphere()`; . К данному примитиву относятся аргументы, отвечающие за заданию r -радиуса или d -диаметра. При построении стандартной сферы моделируется фигура больше похожая на многогранник (рис.6). Как это исправить? За сглаживание боковых поверхностей моделей отвечает аргумент `$fn`. К нему присваивается необходимое количество полигонов для фигуры. Чтобы цилиндр или сфера имели более ровную поверхность принято присваивать значение `$fn=64` (рис.7). Данный аргумент может быть применен к одной фигуре или к фигурам всей программы. В первом случае аргумент записывается в круглые скобки через запятую с другими аргументами, во втором случае в начале программы отдельно от всех примитивов `$fn=64`;

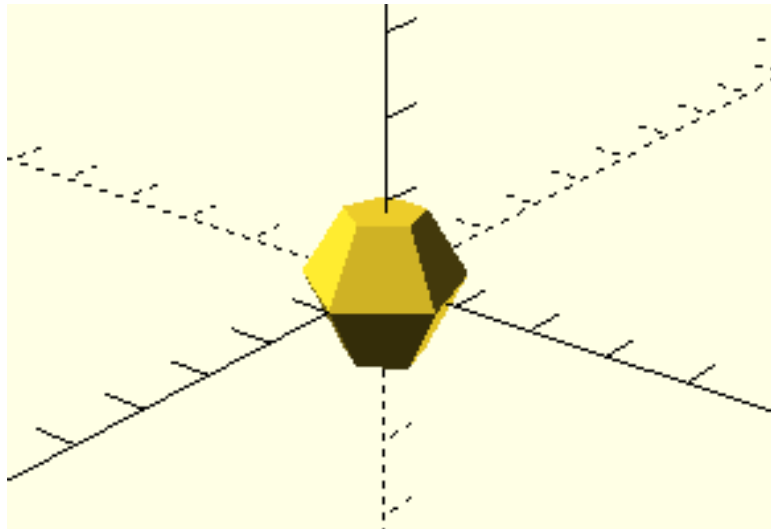


Рис.6. Модель стандартной сферы

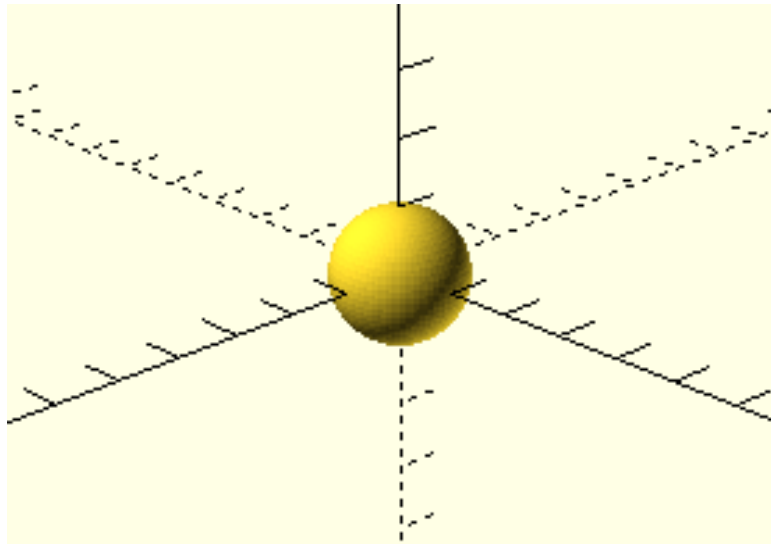


Рис.7. Модель сферы со сглаженной поверхностью

Задания для самостоятельной работы:

1. Создайте куб сторонами [20, 30, 50]
2. Создайте модель цилиндра, имеющую высоту 10 и диаметр 6 и расположение центром в начале осей координат.
3. Изменяя количество полигонов, высоты и радиуса в цилиндре, измените фигуру из задания 2 таким образом, чтобы получился куб со сторонами 10
4. Постройте шестиугольную перевернутую пирамиду высотой 8 и радиусом основания 7

Контрольные вопросы:

1. Что будет если не закрыть примитив?
2. Какого радиуса будет сфера если ее код выглядит следующим образом: `sphere(10);`?

3. Что будет если построить одновременно несколько моделей фигур, например, куб стороной 10 и сферу радиуса 7?

ЛАБОРАТОРНАЯ РАБОТА № 3

ОПЕРАТОРЫ В OpenSCAD

Цель: Научиться изменять цвет фигур и положение моделей в пространстве

Простейшие операторы в OpenSCAD

Для изменения моделей фигур существуют операторы трансформации. При помощи этих операторов модели меняют положение в пространстве. Существуют операторы, отвечающие за цвет создаваемой фигуры. Для каждого из операторов трансформации справедливы следующие правила: Оператор обязательно относится к одному или нескольким примитивам. После описания оператора не ставится точка с запятой. Это происходит потому, что оператор автоматически закрывается вместе с примитивом, к которому относится. Стоит отметить, что если примитивов несколько, то все они записываются внутри фигурных скобок после оператора. Если же операторов, относящихся к примитиву несколько, то они записываются перед примитивом один за другим без каких-либо разделяющих знаков.

Существуют следующие простейшие операторы:

- Translate –перемещает фигуры в пространстве относительно осей координат
- Rotate – поворачивает фигуру относительно оси
- Scale – изменяет исходный масштаб фигуры в несколько раз
- Resize – изменяет исходный размер фигуры
- Color – задает цвет фигуре

У каждого оператора есть свои аргументы.

Аргументы операторов в OpenSCAD

Операторы `translate`, `rotate`, `scale`, `resize` имеют стандартно по три аргумента отвечающих за перемещение и изменение модели относительно осей координат `x`, `y`, `z` соответственно. Эти аргументы записываются в квадратных скобках без указания имени.

Оператор перемещения в OpenSCAD

Для перемещения фигуры в пространстве используется оператор `translate()`. Оператору задаются координаты с новым местоположением модели. Например, попробуем переместить фигуру по координатам: `x=10,y=10,z=0` (рис.8). Код будет выглядеть следующим образом:
`translate([10,10,0])`

`cylinder(h=10, d1=6, d2=0);`

Сначала описывается оператор перемещения с нужными координатами в аргументах, затем пишется примитив. Напоминаем, что оператор и примитив не разделены между собой точкой с запятой!

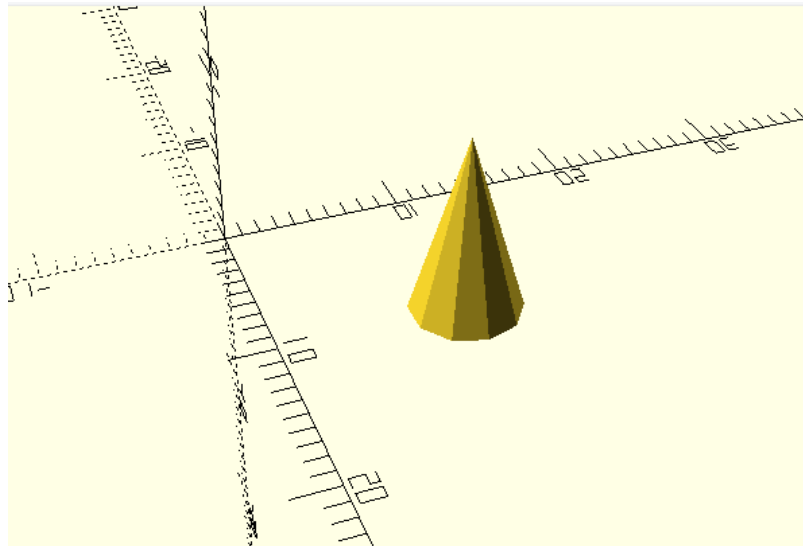


Рис.8. Фигура, перемещенная по осям координат

Оператор поворота в OpenSCAD

Следующий оператор отвечает за поворот фигуры относительно оси `rotate()`. В аргументах задается ось, вокруг которой происходит вращение.

Попробуем повернуть параллелепипед относительно оси z влево на 15 градусов (рис.9). Программа будет выглядеть следующим образом:

```
rotate([0,0,15])
```

```
cube([4,6,2]);
```

В данном случае ось вращения осуществляется только вокруг оси z. Встречаются случаи, когда вращение происходит сразу относительно двух или всех осей сразу (рис.10).

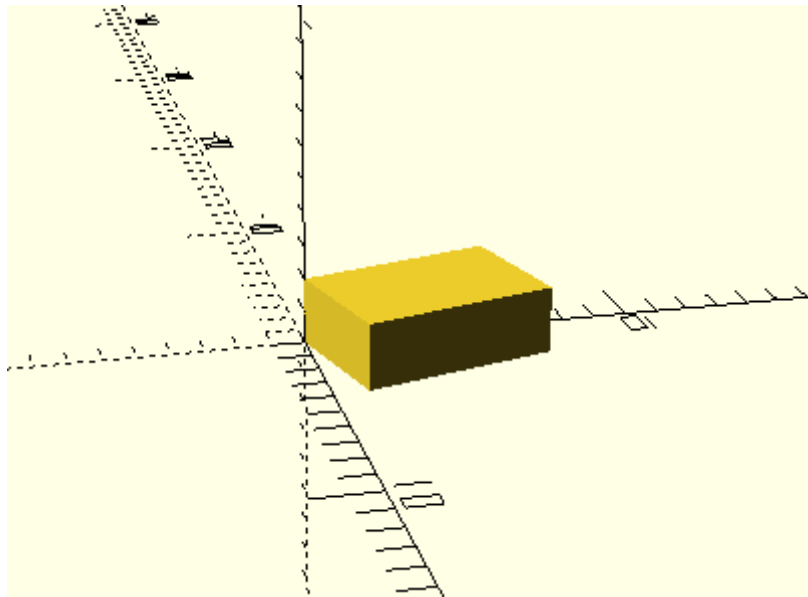


Рис.9. Вращение фигуры относительно одной оси

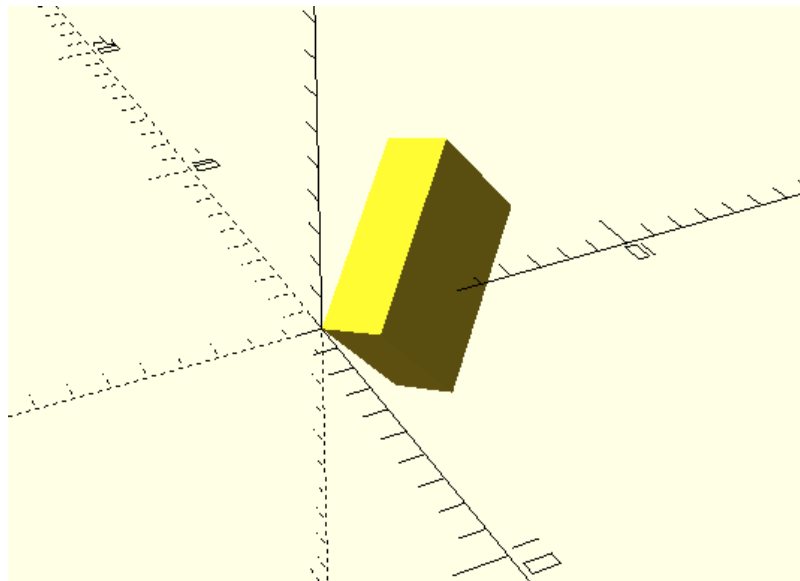


Рис.10. Вращение фигуры относительно нескольких осей

Оператор масштаба в OpenSCAD

Оператор `scale()` отвечает за масштабирование сторон модели. То есть данный оператор может изменить фигуру увеличив или уменьшив ее исходные размеры в несколько раз. Попробуем применить `scale` для цилиндра. Сожмем фигуру по оси `x` и вытянем по оси `y` в два раза, а высоту оставим прежней (рис.11). Для начала оператору задается аргумент, отвечающий за ось `x`. По условию необходимо уменьшить в два раза, значит первый аргумент будет равен 0,5. Десятичные дроби в OpenSCAD записываются через точку! Далее нам необходимо увеличить масштаб по оси `y` в два раза, поэтому задаем второму аргументу значение 2. За высоту цилиндра отвечает третий аргумент `z`. Его не нужно изменять, следовательно, необходимо указать значение 1. Программа будет выглядеть следующим образом:

```
scale([0.5,2,1])  
cylinder(h=5, r=5);
```

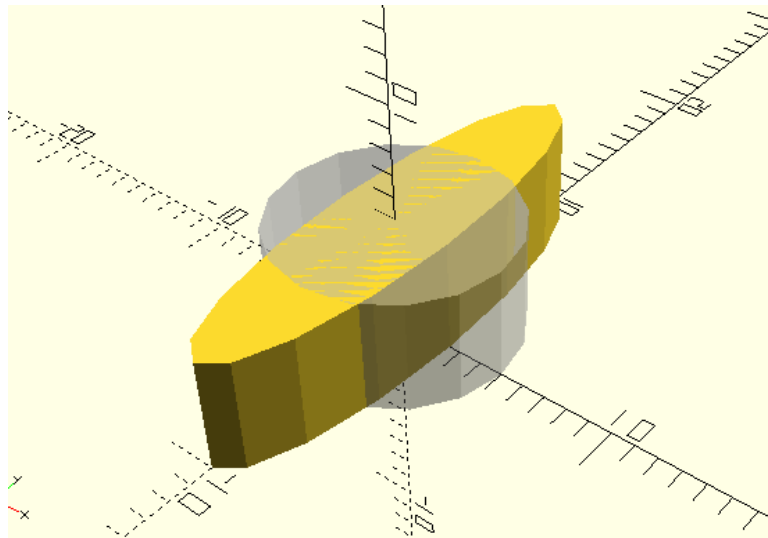


Рис.11. Изменение фигуры в пространстве при установлении масштаба

Оператор размера границ в OpenSCAD

Оператор `resize()` устанавливает границы для существования фигур. Фигура не должна выходить за пределы этого диапазона, поэтому сжимается так, чтобы уместиться в нужный интервал. Попробуем ограничить фигуры в координатах 2мм по каждой оси (рис.12). Код программы:

```
resize([2,2,2]){
```

```

cube([2,2,2]);
translate([3,3,0])
cylinder(h=3,d1=2,                                     d2=0);}

```

По условию оператор относится к нескольким примитивам, поэтому появляются фигурные скобки, следовательно, фигуры рассматриваются как одно целое. Оператор считывает крайние точки фигур и ограничивает их размещение в заданный интервал.

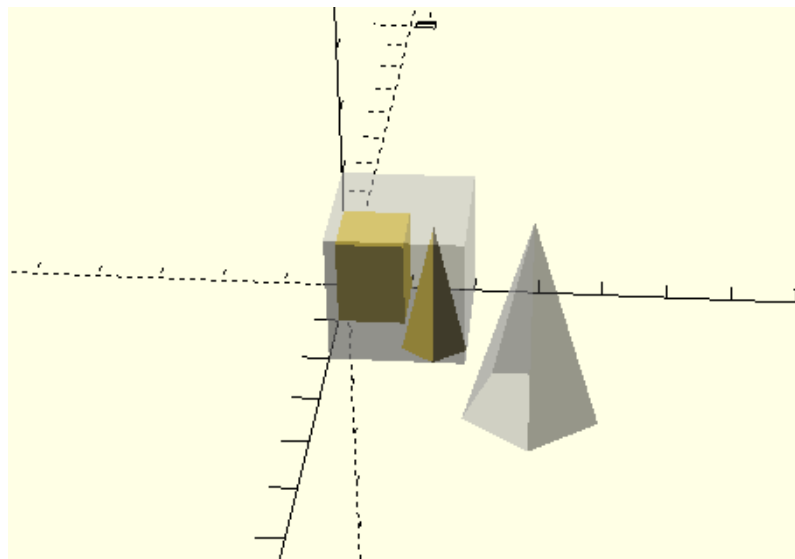


Рис.12. Изменение фигур в пространстве при ограничении диапазона

Оператор цвета в OpenSCAD

Оператор `color()` может изменять цвет и прозрачность фигуры. Задание цвета может выполняться двумя способами: В первом случае, пользователь вписывает название цвета на английском языке в двойные кавычки, а прозрачность присваивается к аргументу `alpha`, во втором, цвет задается RGB форматом, то есть имеются аргументы отвечающие за количество красного, зеленого и синего оттенков в цвете, насыщенность которых варьируется от 0 до 1, а прозрачность задается четвертым аргументом. В обоих случаях прозрачность задается числом из интервала от 0 до 1. Раскрасим фигуры двумя способами изменения цвета и зададим прозрачность 60 процентов (рис.13). Программа имеет следующий вид: Для начала создадим синюю сферу первым способом. Сделаем ее более

круглой при помощи аргумента \$fn
 color("blue", alpha=0.6)
 sphere(\$fn=64);

Оранжевую сферу создадим вторым способом, но для начала переместим ее для того чтобы наши фигуры не совпали между собой. Оператор трансформации и оператор цвета могут менять порядок расположения.

Результат останется прежним
 translate([3,0,3])
 color([0.9,0.5,0.2,0.6])
 sphere(\$fn=64);

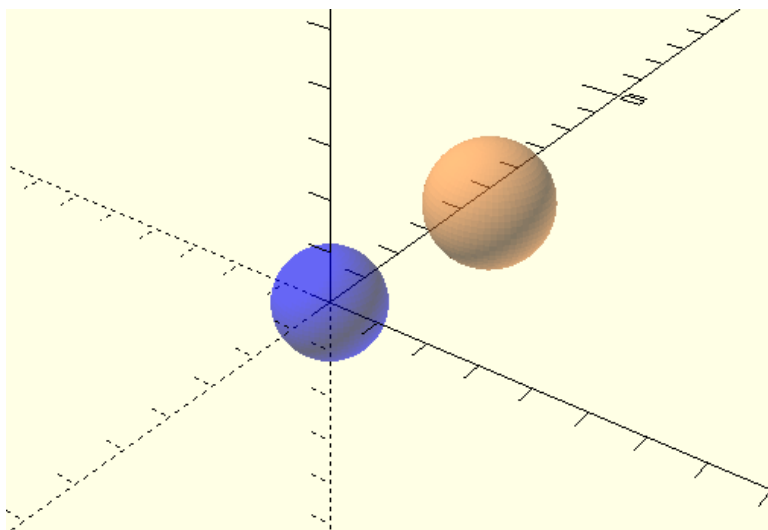


Рис.13. Полупрозрачные фигуры с установленным цветом

Задания для самостоятельной работы:

Используя полученные знания создайте 3-D модель новогодней елки по типу пример с рисунка 14. В модели обязательно должны присутствовать:

- Плоская круглая подставка
- не менее 10 разноцветных шаров
- некоторые шары должны быть стеклянные
- новогодняя звезда
- три уровня треугольных веток

Проследите чтобы модель после реализации на 3D-принтере смогла уместиться в квадратной коробке, стороны которой 3 сантиметра.

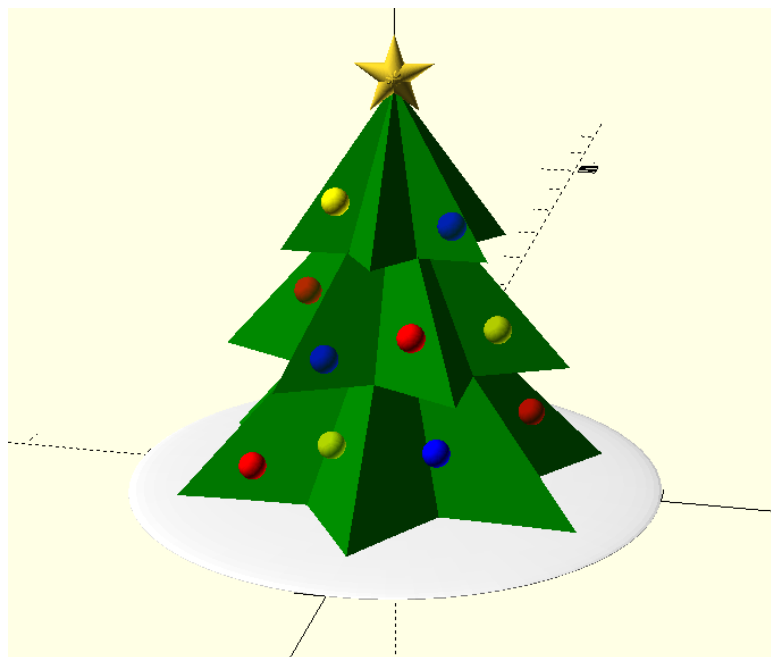


Рис.14. Пример готовой модели новогодней елки

Контрольные вопросы:

1. Имеет ли смысл применять оператор `resize` для куба? Почему?
2. В какой цвет окрасится фигура если каждый аргумент, отвечающий за цвет, будет равен 1? 0? 0,5?
3. Как изменится фигура, если в операторе `scale` один из аргументов будет равен нулю?

ЛАБОРАТОРНАЯ РАБОТА № 4

ОПЕРАТОРЫ В OpenSCAD

Цель: Научиться пользоваться основными булевыми операциями. Познакомиться с простейшими модификаторами.

Булевы операции в OpenSCAD

Булевы операции в OpenSCAD отвечают за операции изменения модели при помощи другой модели. Если пользователь знаком с основами теорий множеств, то разобраться с этой темой будет просто. Существует три стандартных булевых операции: объединение, вычитание и пересечение. Рассмотрим более подробно каждую из операций.

Операция объединения в OpenSCAD

Оператор объединения имеет вид `union(){тело оператора}`. Данный инструмент необходим для того, чтобы сгруппировать несколько фигур в одну. То есть результатом объединения является все детали, примитивы которых входят в тело операции (рис.15). Возникает очевидный вопрос: Зачем объединять фигуры, если по умолчанию все они автоматически соединены между собой? Ответ этого вопрос прост. Иногда требуется произвести действие не над всей деталью, а только над ее частью, которая содержит в себе несколько простых фигур. Также объединение может быть полезно в том случае, если некий оператор имеет структуру, при которой действие может применяться только к одному примитиву, а изменить нужно одновременно несколько частей детали. Фрагмент программы с применением объединения выглядит следующим образом:

```
union({  
cube([50,50,50],                                     center=true);  
sphere(r=33);}
```

Результатом моделирования является деталь с рисунка 15.

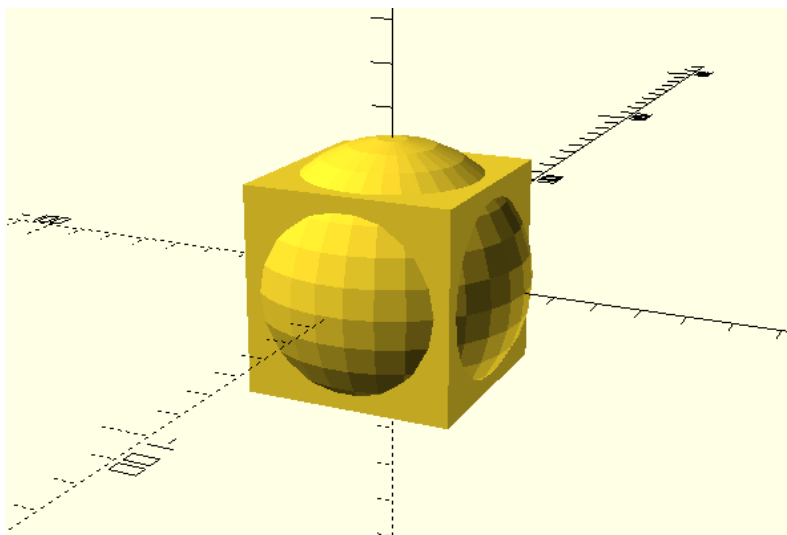


Рис.15. Результат применения оператора объединение

Операция вычитания в OpenSCAD

Иногда для создания некой модели необходимо исключить из одной фигуры другую. За это в OpenSCAD отвечает оператор вычитания *difference()*{*тело оператора*}. На первом месте в теле программы записывается та фигура из которой вычитают, на последующих местах все остальные фигуры, которые вычитаются из первой. Если вам необходимо произвести вычитание из нескольких примитивов, то воспользуйтесь оператором объединения и запишите его в тело *difference* на первой позиции. Обратите внимание, что порядок в данной операции очень важен! Для одних и тех же фигур результат вычитание при изменении порядка будет разным (рис.16,17). Примером использования оператора вычитания

служит следующая программа:

```
difference(){
cube([50,50,50],center=true);
sphere(r=33);}
```

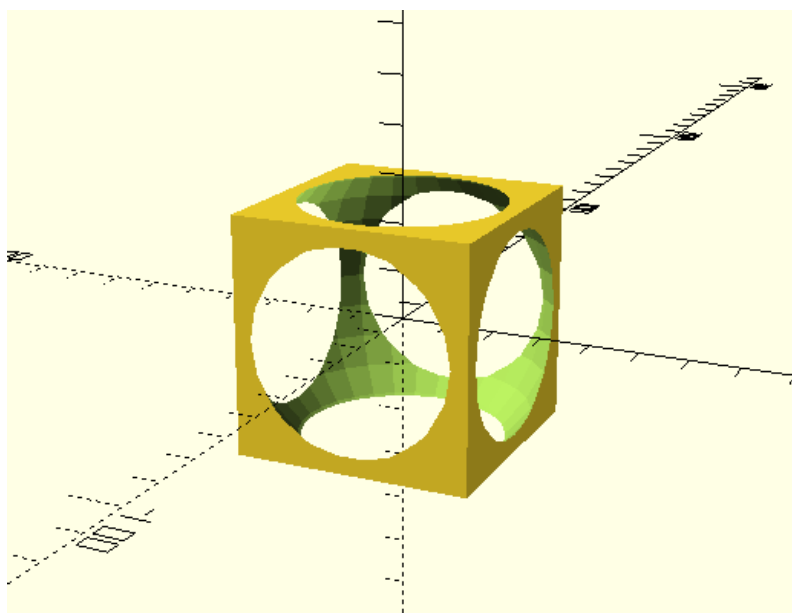


Рис.16. Результат исключения сферы из куба

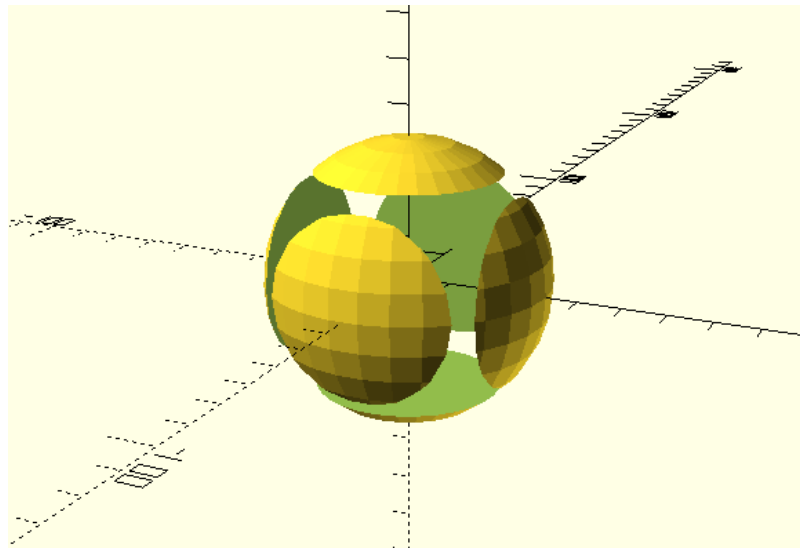


Рис.17. Результат исключения куба из сферы

Операция пересечения в OpenSCAD

Оператор пересечения отвечает за выделение области, которая принадлежит фигурам, чьи примитивы находятся в его теле (рис.18). Выглядит пересечение следующим образом: *intersection(){тело оператора}*. Данный оператор применим как для двух фигур, так и для нескольких. Все используемые примитивы записываются в тело оператора. Пример использования оператора пересечения:

```
intersection(){  
cube([50,50,50],center=true);  
sphere(r=33);}
```

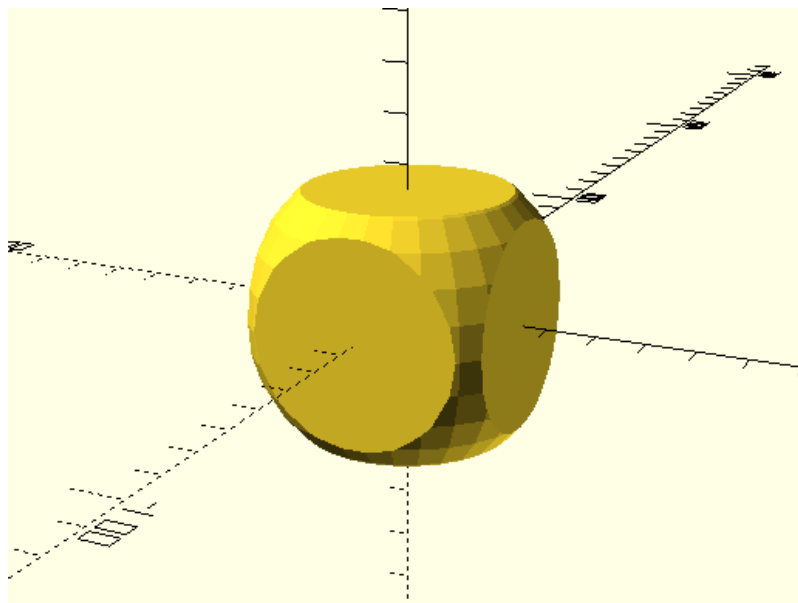



Рис.18. Результат применения оператора пересечение

Модификаторы в OpenSCAD

Для удобства моделирования в OpenSCAD имеется набор инструментов, которые называются модификаторы. С одним из них мы уже знакомы. Двойной слеш `//`- модификатор, отвечающий за комментирование одной строки кода. Сейчас мы познакомимся с остальными.

Модификатор «запретить» в OpenSCAD

В ходе программирования бывают ситуации, при которых необходимо скрыть большую часть кода не удаляя её. Для удобства принято использовать модификатор «запретить» `*`, который отвечает за комментирование не одной строки, как это делает двойной слеш, а целых блоков. Для того чтобы скрыть некую операцию с массивным телом необходимо в начале строки, в которой объявляется операция, добавить звездочку.

Модификатор «показать только» в OpenSCAD

Иногда в процессе программирования возникает необходимость просмотреть на конкретную часть детали, то есть скрыть остальные элементы модели, не удаляя их из кода. Для этого используется модификатор «показать только» `!`. Восклицательный знак ставится перед примитивом или оператором, отвечающим за нужную часть детали, которую необходимо оставить видимой, даже

если она скрыта, остальные элементы, проектированной фигуры, не отражаются на построенной модели (рис 19).

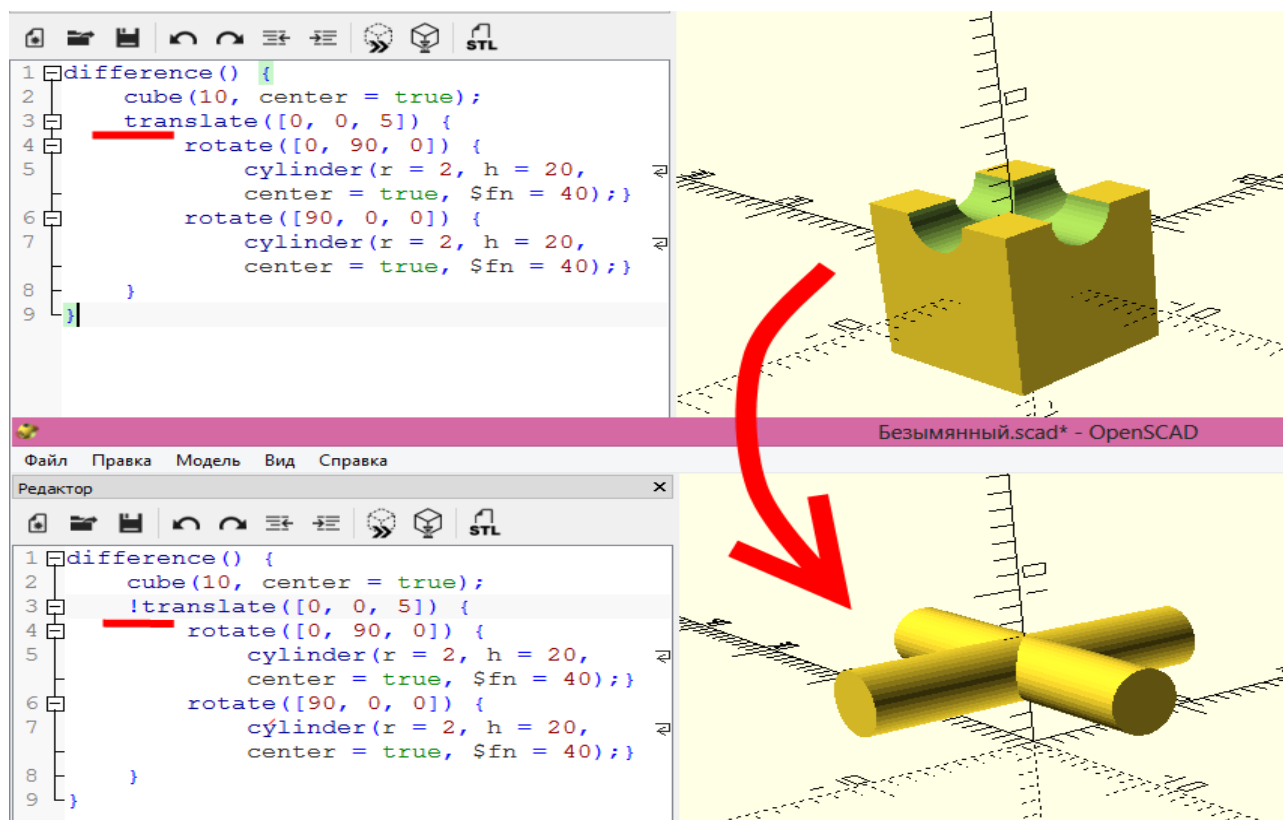


Рис.19. Исходная программа и результат применения модификатора ! в коде

Модификатор «выделить» в OpenSCAD

Также существует модификатор «выделить» #, отвечающий за отображение невидимых деталей модели. Выделенный фрагмент закрашивается в красный цвет и становится полупрозрачным (рис.20). Стоит отметить, что никаких изменений с фигурой не происходит. Данный модификатор служит помощником для конструктора. Поэтому выделение отображается только в предварительном просмотре.

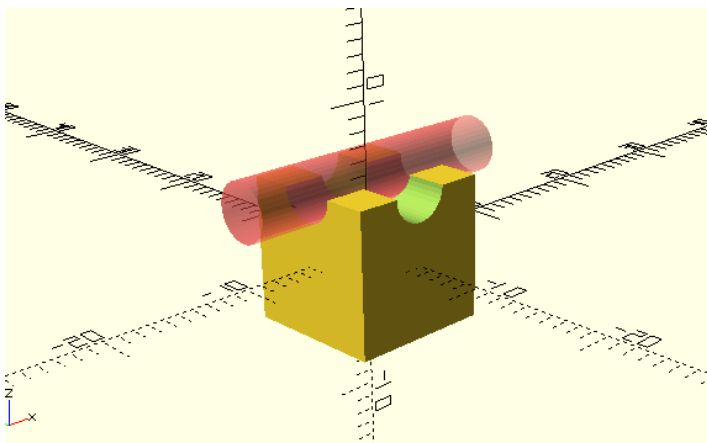


Рис.20. Пример выделения одного объекта из детали

Модификатор «прозрачный фон» в OpenSCAD

Также существует модификатор «прозрачный фон» он отвечает за исключение из фигуры материальной оболочки. Если проставить % перед кодом какой-либо фигуры, то в предварительном просмотре данная фигура будет прозрачной (рис.21), а в рендеринге не отобразится совсем. Если попробовать вычесть такую фигуру из какой-либо детали, то модель не изменится, потому что модификатор исключает фигуру из материального мира, оставляя только эффект присутствия (рис.21).

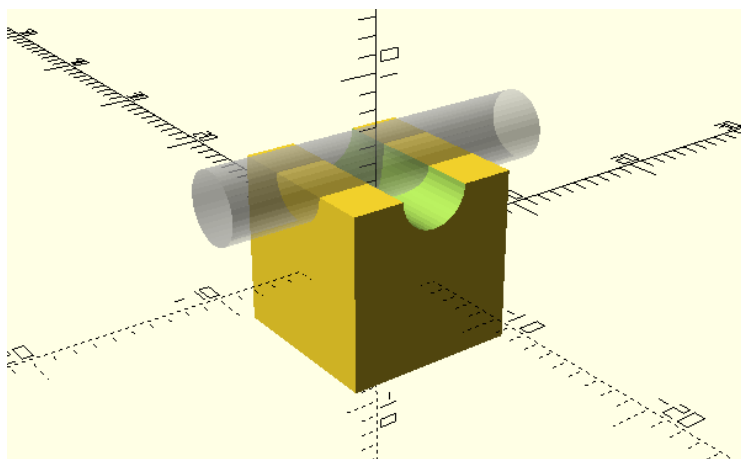


Рис.21. Пример применения модификатора прозрачный фон

Задания для самостоятельной работы:

1. Постройте деталь со значка OpenSCAD
2. Постройте модель игрального кубика со скругленными краями

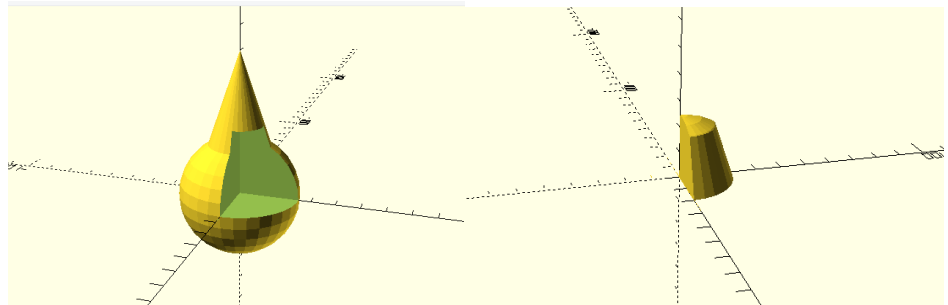
Контрольные вопросы:

1. Не строя модель, ответе на вопрос: Какая из деталей получится после выполнения следующей программы:

```

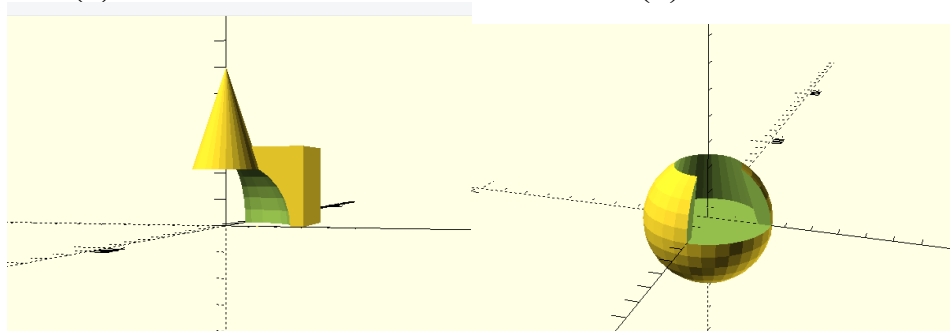
difference({
sphere(r=25);
cylinder(h=60,r1=20,r2=0);
cube([30,30,30]);}

```



(a)

(б)

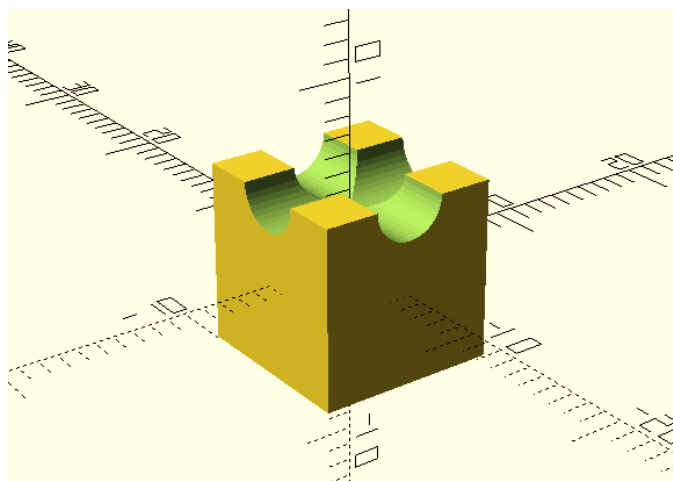


(в)

(г)

2. Каким образом необходимо изменить код чтоб результатом программы стала модель с рисунка *a*?
3. В чем отличает модификаторов # и %?
4. Не строя деталь, выберите, из приведённых вариантов, верный ответ на вопрос: Как будет выглядеть модель в окне привью после добавления модификатора «Показать только» перед примитивом, отвечающим за построения цилиндра?

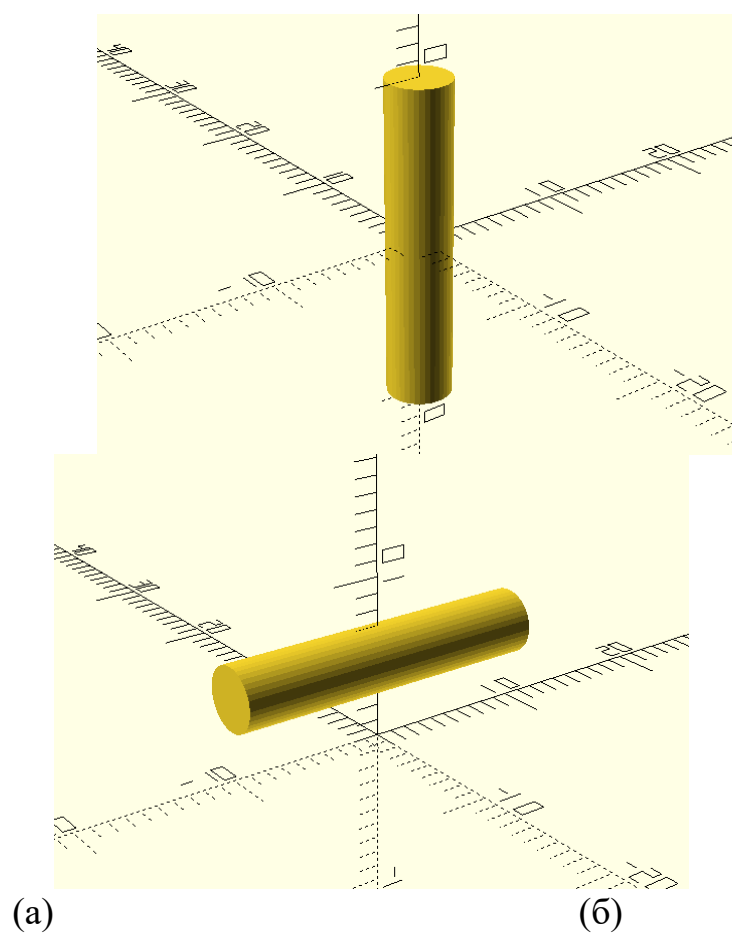
Исходная модель:

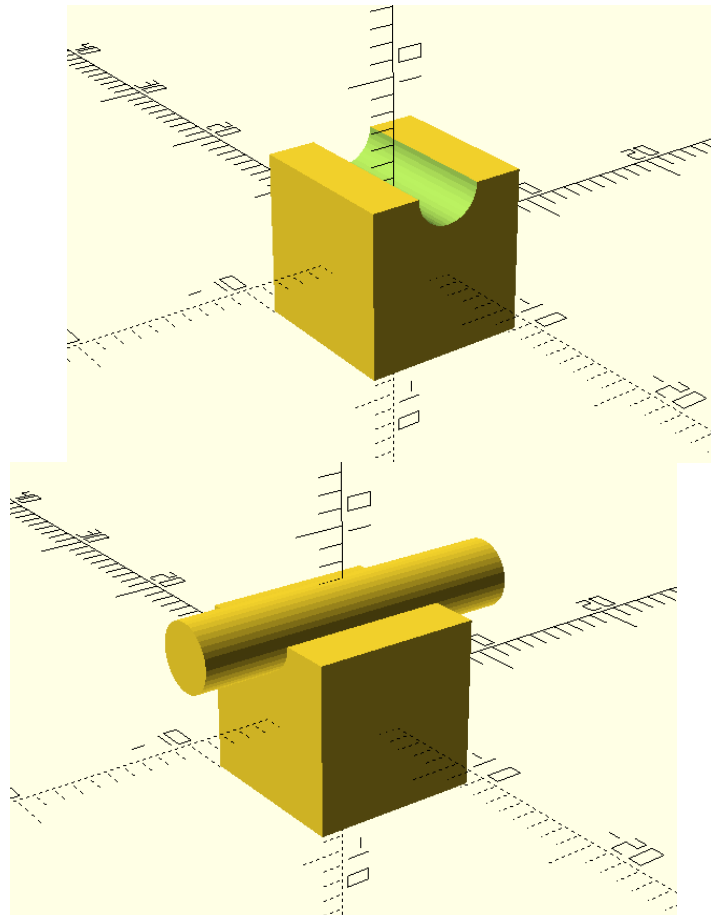


Измененный код:

```
difference() {  
  cube(10, center = true);  
  translate([0, 0, 5]) {  
    rotate([0, 90, 0]) {  
      !cylinder(r = 2, h = 20,  
        center = true, $fn = 40);}   
    rotate([90, 0, 0]) {  
      cylinder(r = 2, h = 20,  
        center = true, $fn = 40);} } }
```

Варианты измененной программы:





(B)

(Г)

ЛАБОРАТОРНАЯ РАБОТА № 5

АЛГОРИТМЫ И МОДУЛИВ OpenSCAD

Цель: Познакомиться с основными алгоритмами. Научиться использовать модули при моделировании.

Алгоритмы в OpenSCAD

Алгоритмы в любом языке программирования значительно облегчают процесс создания программы. Существуют два основных алгоритма в OpenSCAD это алгоритм цикл и условие. Рассмотрим каждый из них подробнее.

Цикл в OpenSCAD

Цикл в OpenSCAD имеет следующий вид:

for(i=[начальное значение: шаг: конечное значение]) {тело алгоритма}

В теле алгоритма прописывается то действие которое должно повторяться несколько раз. Шаг цикла может не указываться, тогда алгоритм будет иметь вид:

for(i=[начальное значение: конечное значение]) {тело алгоритма}

Примером цикла может служить следующая программа:

```
for                                                    (i=[0:5:100]){  
translate([40*i,0,0])  
cube([40,40,40]);}
```

Условие в OpenSCAD

Алгоритм условия. Применяется в том случае, если код выполняется, когда удовлетворяется некое условие. Данный алгоритм имеет следующий вид:

if(условие){тело алгоритма1}else{тело алгоритма2}

Обратите внимание, что алгоритм условия может включать в себя действия, которые выполняются, если условие не удовлетворено.

В алгоритме условия могут содержаться арифметические знаки, такие как:

==	равно
<	меньше
>	больше

&&	и
!=	не равно
%	делится без остатка
	и т.п. . .

Пример программы,

использующей условие:

```
for(i=[0:5:50]){
  if (i%10 != 0) {
    translate([5*i,0,0])
    cube(15);}
  else{
    translate([5*i,-10,0])
    sphere(10);}}
```

Модуль в OpenSCAD

Программа OpenSCAD умеет работать с модулями. Чтобы не описывать несколько раз один и тот же кусок кода мы можем присвоить ему имя и вызывать по мере надобности. Модуль имеет вид:

module ИмяМодуля(){тело модуля}

Вызов модуля происходит путем указания имени и имеет следующий вид:

ИмяМодуля();

```
test();
```

```
  translate([0,80,0])
```

```
  test();
```

```
module test(){
```

```
  intersection(){
```

```
    union(){
```

```
      color("red")
```

```
      rotate([0,60,20])
```

```

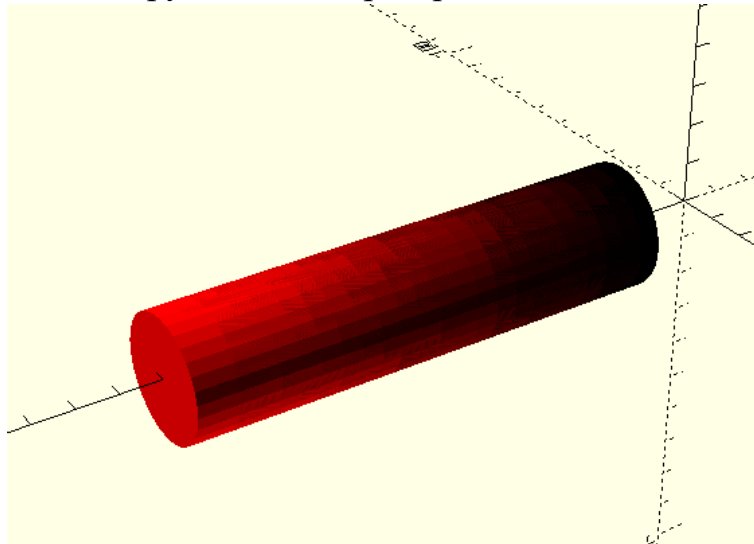
cylinder(h=140,d=5);
    difference(){
        color("blue")
        cube([100,20,30]);
        translate([20,10,0])
        sphere(r=30);}}
    translate([50,10,20])
    sphere(r=50);}}

```

Обратит внимание, что вызов и описание модуля может быть в произвольном порядке!

Задания для самостоятельной работы:

1. Смоделируйте цилиндр с градиентом



2. Постройте модель решета

Контрольные вопросы:

1. Исправьте ошибки в программе:

```

diference(){
    color('green');
    rotate([0.90.0]);
    cylinder(h=6,d=70);

    rotate([22.0.0]);
    resize([6,32,32]);
    test();

```

```

        resize([6,25,25]);
test();
        translate([0,17,17]);
test();
        translate([0,-17,17]);
test();
        translate([0,17,-17]);
test();
        translate([0,-17,-17]);
test();
        translate([0,-24,0]);
test();
        translate([0,0,-24]);
test();
        translate([0,24,0]);
test();
        translate([0,0,24]);
test();
    }

```

```

module test(){
for(i=[0:7]){
    rotate([i*45.0,0,0]){
        cylinder(h=10,d1=6,d2=0);
        spha(3);} } }

```

2. Что получится после выполнения исправленной программы?

ШПАРГАЛКА OpenSCAD

3D примитивы

[`sphere\(radius | d=diameter\)`](#)
[`cube\(size, center\)`](#)
[`cube\(\[width,depth,height\], center\)`](#)
[`cylinder\(h,r|d,center\)`](#)
[`cylinder\(h,r1|d1,r2|d2,center\)`](#)

Операторы трансформации

[`translate\(\[x,y,z\]\)`](#)
[`rotate\(\[x,y,z\]\)`](#)
[`scale\(\[x,y,z\]\)`](#)
[`resize\(\[x,y,z\],auto\)`](#)
[`mirror\(\[x,y,z\]\)`](#)
[`color\("colorname",alpha\)`](#)
[`color\(\[r,g,b,a\]\)`](#)

Операции

[`union\(\)`](#)
[`difference\(\)`](#)
[`intersection\(\)`](#)

Модификаторы

[`*`](#) запрещать
[`!`](#) покажи только
[`#`](#) выделить / отладки
[`%`](#) прозрачный фон

Алгоритмы

[`for \(i = \[start:end\]\) { ... }`](#)
[`for \(i = \[start:step:end\]\) { ... }`](#)
[`if \(...\) { ... }`](#)
[`module name\(...\) { ... }`](#)
[`name\(\);`](#)
[`import\("...stl"\)`](#)
[`include <...scad>`](#)
[`use <...scad>`](#)

Специальные переменные

[`\$fa`](#) угловое положение минимума
[`\$fs`](#) минимальный размер
[`\$fn`](#) число фрагментов
[`\$vpr`](#) окно просмотра углов поворота в градусах
[`\$vpt`](#) окно просмотра перевода
[`\$vpd`](#) окно просмотра расстояние камеры

Приложение 2.

Анкета на апробацию

1. Был ли данный комплекс лабораторных работ полезен для вас?
2. Узнали что-то новое?
3. Понравилась форма предоставления материала?
4. Удобно ли было читать текст заданий?
5. Все описания были понятные?
6. Возникли какие-либо трудности в изучении материала?
7. Что запомнилось больше всего?
8. Сможете ли вы самостоятельно моделировать в OpenSCAD после прохождения данного курса?
9. Легко ли было установить программу?
10. Чем вам понравилась программа?
11. Чем не понравилась?
12. Есть ли какие-либо пожелания в улучшении комплекса?
13. Поставьте оценку от 1 до 10 программе OpenSCAD. Обоснуйте свой выбор
14. Поставьте оценку от 1 до 10 лабораторным заданиям. Обоснуйте свой выбор