



Проектная деятельность

Разработка ИЭТР на базе
графического ядра C3D

Участники - студенты 2-го курса:

- Фролов А. М. группы 181-326
- Серяков А. В. группы 181-326
- Волобуев Р. А. группы 181-326
- Новохатский М. Б. группы 181-325



C3D Labs

Products

Applications

Customers

For Developers

Blog

Company

Contact

Develop Stunning 3D

Цель проекта:

создание примеров использования
функционала ядра C3D



Распределение ролей в проекте:

- 1)Серяков А.В. - разработка примеров, создание аннотаций;
- 2)Волобуев Р.А. - разработка примеров, создание аннотаций;
- 3)Новохатский М.Б. - создание аннотаций, создание видео;
- 4)Фролов А.М. - создание аннотаций, создание презентации.



C3D Labs

Products

Applications

Customers

For Developers

Blog

Company

Contact

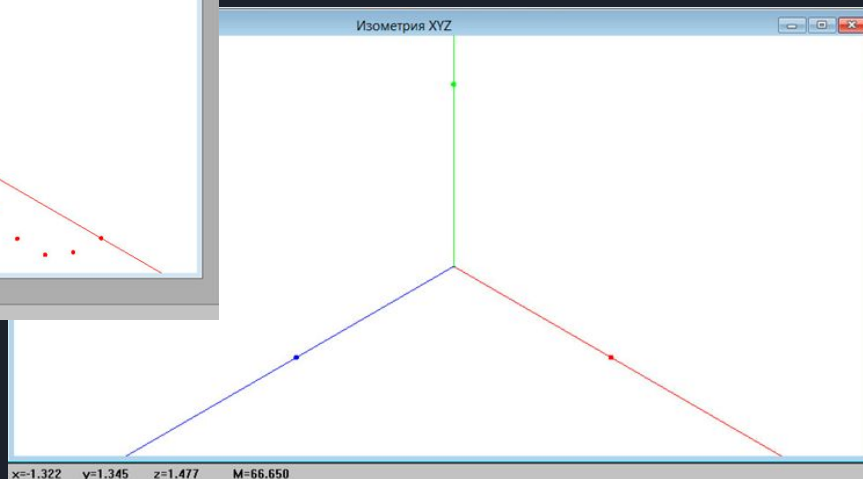
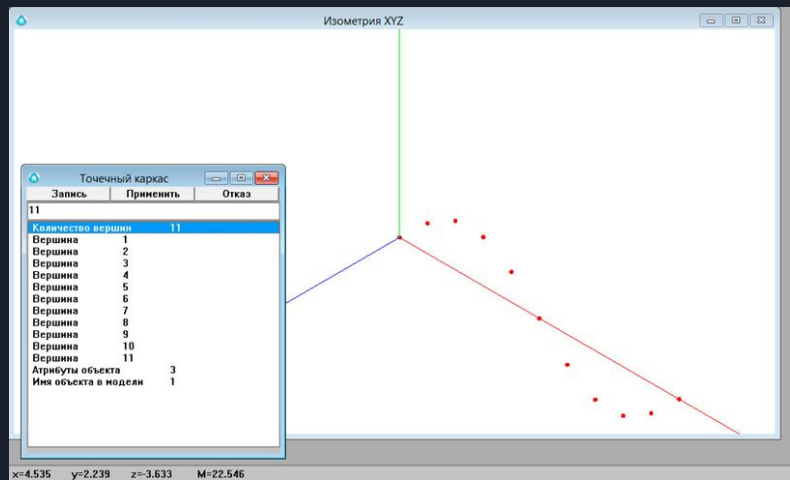
Develop Stunning 3D

C3D T

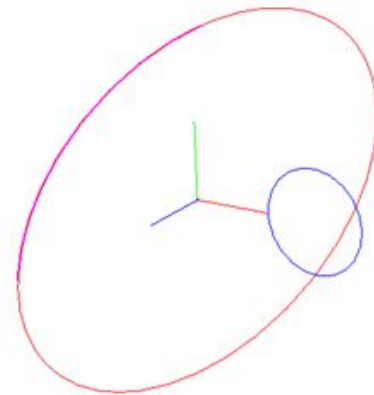
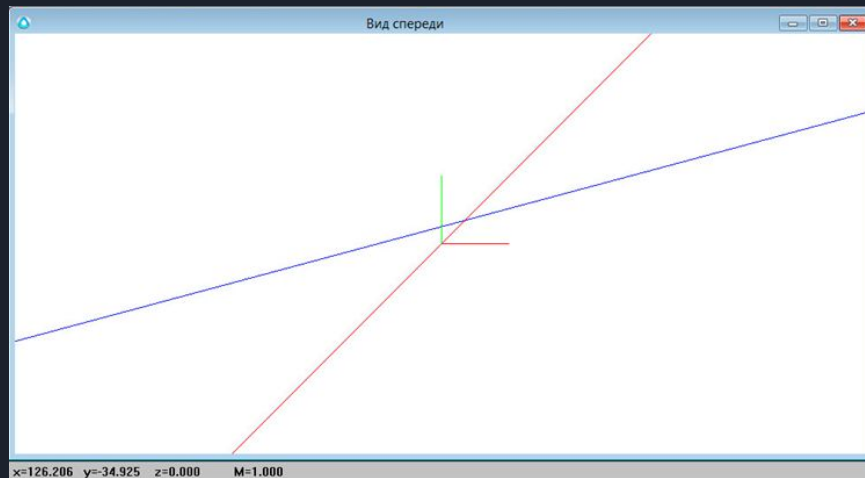
Этапы работы
1.

Определение базовых функций
ядра

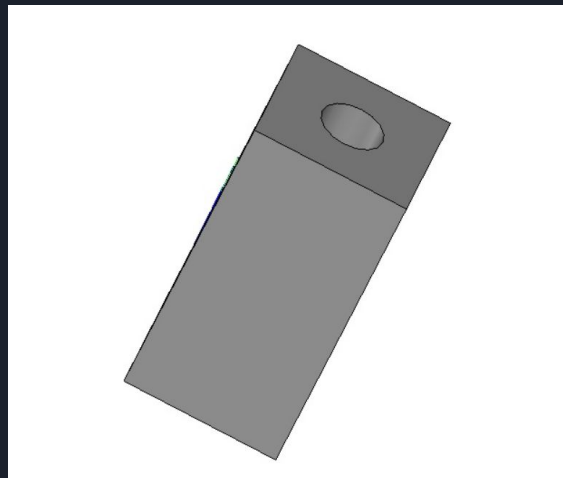
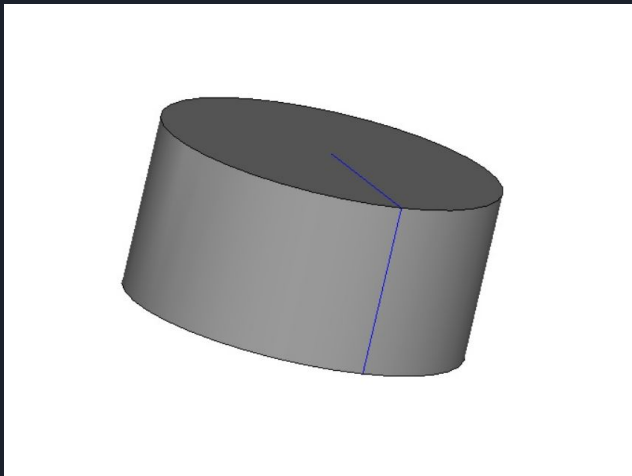
1. Построение точек в различных системах координат



2. Построение прямых линий и кривых



3. Построение поверхностей и тел вращения





C3D Labs

Products

Applications

Customers

For Developers

Blog

Company

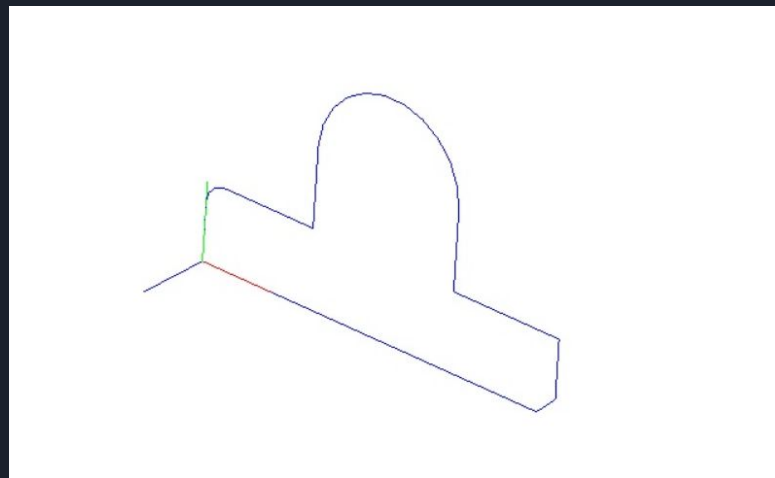
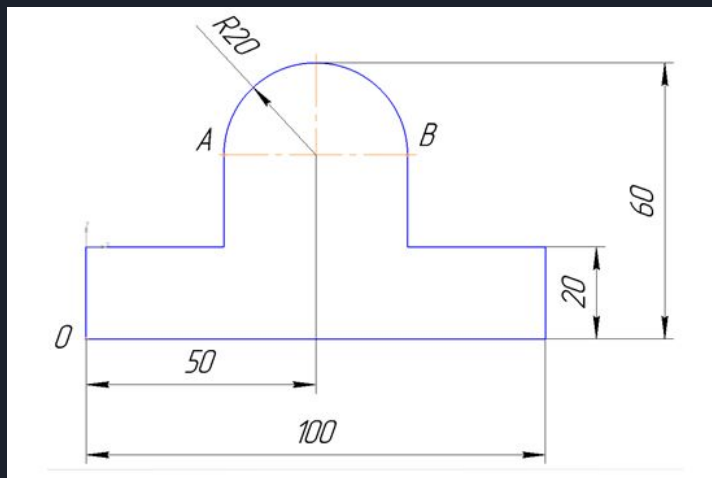
Contact

Develop Stunning 3D

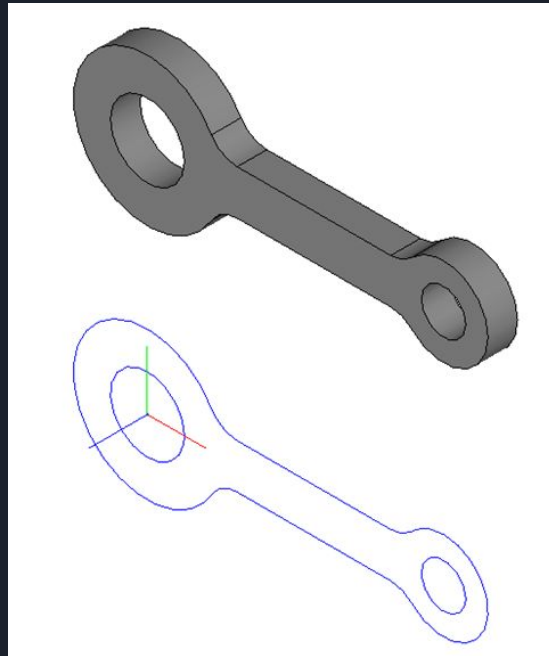
2.

Разработка программ для
демонстрации функций ядра

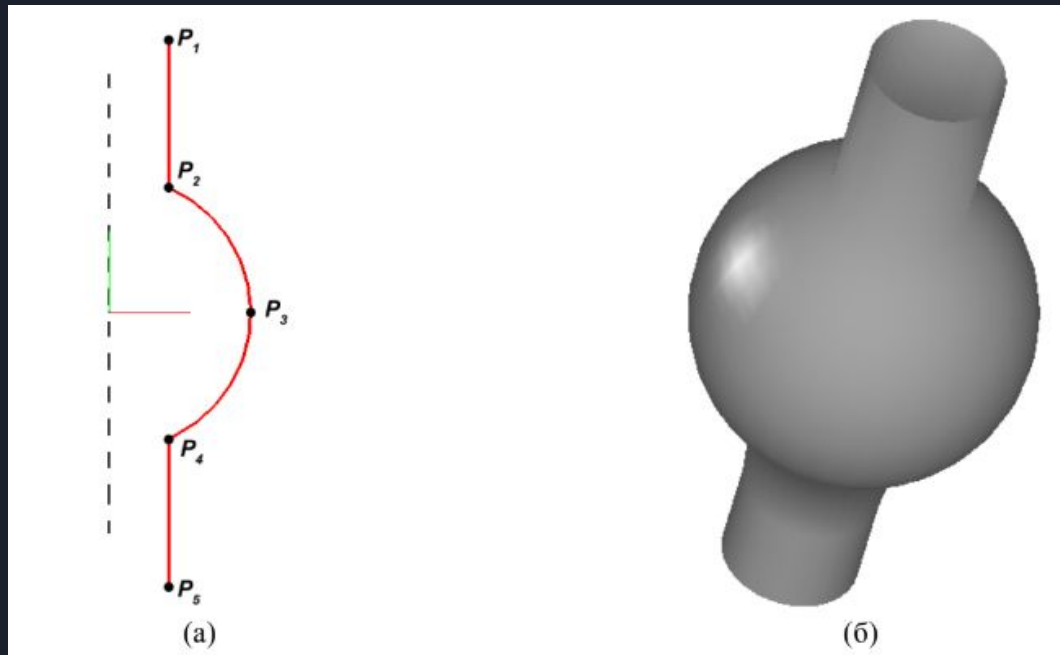
1.Пример алгоритма по созданию эскиза



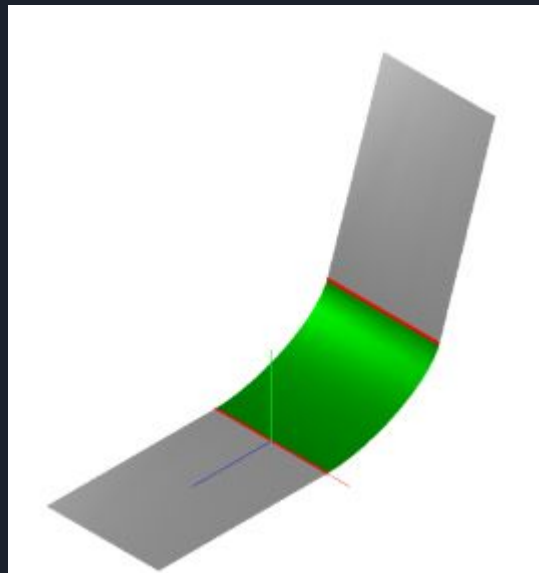
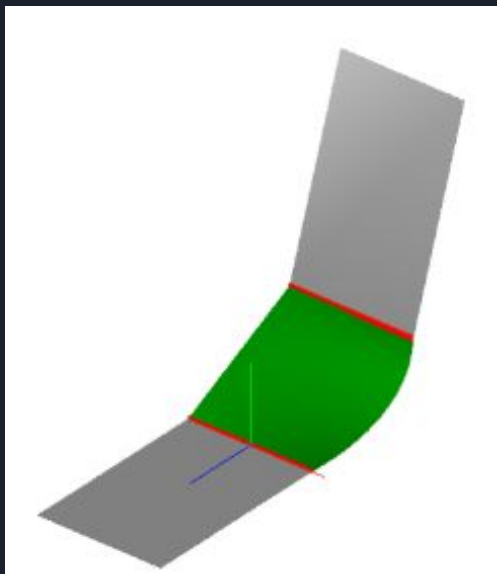
2.Пример алгоритма по созданию поверхностей путем выдавливания эскиза



3. Пример алгоритма по созданию поверхностей путем вращения эскиза вокруг оси



4.Алгоритм создания кривых, фасок, сопряжений





C3D Labs

Products

Applications

Customers

For Developers

Blog

Company

Contact

Develop Stunning 3D

C3D T

3.

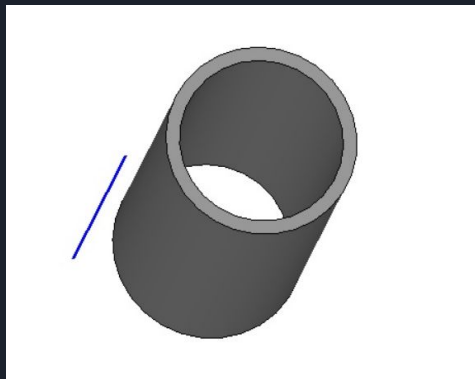
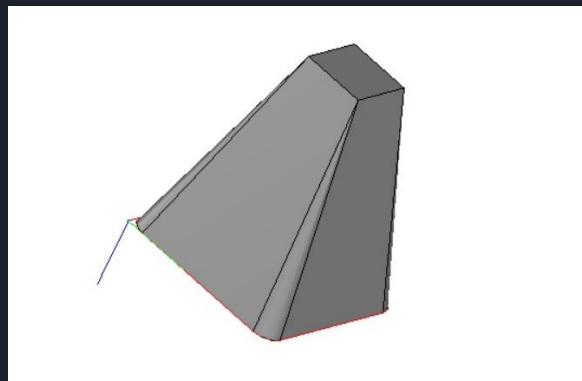
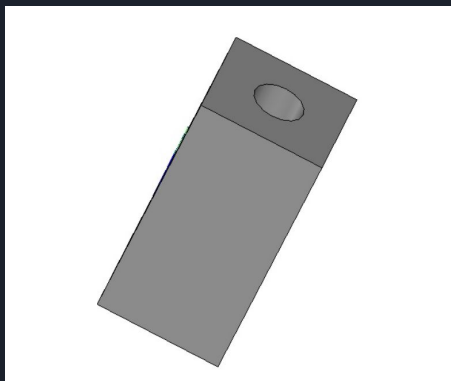
Разработка структуры
аннотации для программы



Каждая аннотация имеет следующее содержание:

- 1) Задача (постановка задачи, которую решает программа);
- 2) Поэтапный разбор кода программы с пояснениями;
- 3) Код программы, собранный воедино;
- 4) Список используемых в программе функций с пояснениями;
- 5) Список используемых в программе классов с пояснениями.

Примеры результатов работы



Примеры результатов работы

Пример №1

Задача: реализация 2D эскиза. Создание контура с использованием отрезков, дуг, скругления и фаски.

Этапное построение эскиза.

1. Создаем локальную СК:

```
MbPlacement3Dp1;
```

2. Затем создаем массив точек, вершин ломаной, и объявляем их.
Создаем массив точек:

```
SArray<MbCartPoint>arrPnts(19);  
Добавляем в массив точки:
```

```
arrPnts.Add(MbCartPoint(30, 40));  
arrPnts.Add(MbCartPoint(30, 20));  
arrPnts.Add(MbCartPoint(0, 20));  
arrPnts.Add(MbCartPoint(0, 0));  
arrPnts.Add(MbCartPoint(100, 0));  
arrPnts.Add(MbCartPoint(100, 20));  
arrPnts.Add(MbCartPoint(70, 20));  
arrPnts.Add(MbCartPoint(70, 40));
```

3. Создаем ломаную.

```
MbPolyline* pPolyline = new MbPolyline(arrPnts, false/* Флаг незамкнутой линии  
*/);
```

4. Создаем фаску

```
ChamferPolyContour(pPolyline, 5, 5, false, false, arrPnts[4]);
```

5. Создаем контур

```
MbContour* pContour = new MbContour(*pPolyline, true);
```

6. Создаем скругление.

```
FilletPolyContour(pPolyline, 5, false, arrPnts[2], pContour);
```

7. Создаем дугу (Arc):

```
MbCartPointArcCenter(50, 40);  
constdouble RADIUS = 20;  
MbArc* pArc = new MbArc(arcCenter, RADIUS, arrPnts[7], arrPnts[0], 1  
/*initSense*/);
```

8. Добавляем арку в контур:

```
pContour->AddSegment(pArc);
```

9. Добавляем контур в сцену:

```
viewManager->AddObject(Style(1, RGB(0, 0, 255)), pContour, &p1);
```

10. Уменьшаем счетчик ссылок динамически созданных объектов ядра:

```
::DeleteItem(pPolyline);  
::DeleteItem(pArc);  
::DeleteItem(pContour);
```

Код программы

```
void MakeUserCommand0()  
{  
    MbPlacement3Dp1; // ЛокальнаяСК (по умолчанию совпадает с мировойСК)  
  
    // Вершиниломаной  
    // Создание массива точек  
    SArray<MbCartPoint>arrPnts(19);  
  
    arrPnts.Add(MbCartPoint(30, 40));  
    arrPnts.Add(MbCartPoint(30, 20));  
    arrPnts.Add(MbCartPoint(0, 20));  
    arrPnts.Add(MbCartPoint(0, 0));  
    arrPnts.Add(MbCartPoint(100, 0));  
    arrPnts.Add(MbCartPoint(100, 20));  
    arrPnts.Add(MbCartPoint(70, 20));  
    arrPnts.Add(MbCartPoint(70, 40));
```

```
arrPnts.Add(MbCartPoint(70, 40));  
  
    // ЛоманаялиниисериализацииarrPnts  
    MbPolyline* pPolyline = new MbPolyline(arrPnts, false/* Флагзамкнутойлинии */);  
  
    ChamferPolyContour(pPolyline, 5, 5, false, false, arrPnts[4]);  
  
    MbContour* pContour = new MbContour(*pPolyline, true);  
  
    FilletPolyContour(pPolyline, 5, false, arrPnts[2], pContour);  
  
    // Дуга окружности для замыкания ломаной.  
    // При построении указывается центр, радиус, начальная и конечная точки и  
    // направление обхода дуги между этими точками (значение initSense>0  
    // соответствует  
    // обходу против часовой стрелки, а initSense<0 - по часовой стрелке).  
    MbCartPointArcCenter(50, 40);  
    constdouble RADIUS = 20;  
    MbArc* pArc = new MbArc(arcCenter, RADIUS, arrPnts[7], arrPnts[0], 1  
    /*initSense*/);  
  
    // Контуриздвухсегментов  
    pContour->AddSegment(pArc);  
  
    // Отображениеконтура  
    if (pContour)  
        viewManager->AddObject(Style(1, RGB(0, 0, 255)), pContour, &p1);  
  
    // Уменьшение счетчиков ссылок динамически созданных объектов ядра  
    ::DeleteItem(pPolyline);  
    ::DeleteItem(pArc);  
    ::DeleteItem(pContour);  
}
```

Список используемых функций

MbCartPoint::Add(X, Y) - создание двумерной точки. X, Y - координаты точек



Вывод

Данный проект нацелен на обеспечения удобства программистам при освоении ядра C3D путем создания примеров использования функций ядра.



СПАСИБО ЗА ВНИМАНИЕ