**ПО «GMESH Generator»**

**Руководство оператора**

**Аннотация**

Данное программное обеспечение предназначено для построения регулярных сеток для выпуклых n-угольников.

Оглавление

[1. Назначение программы 3](#_Toc473559916)

[2. Условия выполнения программы 3](#_Toc473559917)

[3. Выполнение программы 4](#_Toc473559918)

[3.1. Запуск основного окна ПО «GMESH Generator» 4](#_Toc473559919)

[3.2. Используемые форматы 6](#_Toc473559921)

[3.3. Завершение программы 10](#_Toc473559922)

## 

## 1. Назначение программы

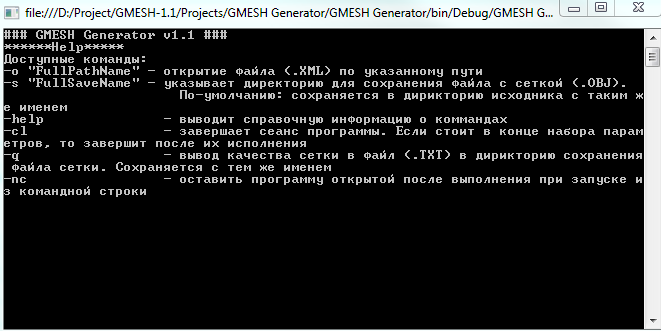
ПО «GMESH Generator» решает задачу построения регулярной сетки для выпуклых n-угольников.

## 2. Условия выполнения программы

Для функционирования программного изделия ПЭВМ должны удовлетворять следующим требованиям: оперативная память не менее 1ГБ, доступная дисковая память не менее 10ГБ, процессор с PR-рейтингом не менее 2000, манипуляторы мышь и клавиатура. Программное обеспечение должно функционировать под управлением операционных систем MS WINDOWS 7 SP1 или более поздних версий c установленным ПО .Net Framework 4.5+

## 3. Выполнение программы

Выполнение программы генерации сетки для плоских n-угольников требует файлы установленного формата XML или OBJ (см. пункт 3.2)



окно доступных команд генератора

## 3.1. Запуск основного окна ПО «GMESH Generator»

## Режим запуска ПО «GMESH Generator» выбирается в зависимости от следующих параметров командной строки:

3.1.1. Параметр –o “путь до файла” открывает по указанному пути файл с исходными данными формата XML или OBJ(пример структуры файла см. в п. 3.2) (пример: -o "C:\Users\klim2\Desktop\newtest\4.xml");

3.1.2. Параметр –s “путь до файла” сохраняет по указанному пути сгенерированную сетку в формате OBJ (пример структуры файла см. в п. 3.2) ( пример: -s "C:\Users\klim2\Desktop\TestResults\5674.obj");

***По умолчанию*** *файл со сгенерированной сеткой сохраняется по пути расположения файла с исходными данными. Берется имя файла с исходными данными и к названию файла прибавляется строка “MESH”. Пример:*

*Исходный файл: "C:\Users\klim2\Desktop\newtest\4.xml"*

*Сохраняемый файл: "C:\Users\klim2\Desktop\newtest\4MESH.obj"*

3.1.3. Параметр –help выводит в окне справочную по доступным командам;

3.1.4. Параметр –q выводит качество сетки в формате .txt (пример структуры файла см. в п. 3.2) с тем же именем файла, что и задается в пункте 3.1.2;

3.1.5. Параметр –nc оставляет командную строку в ждущем режиме;

3.2. Далее в папке указанной с помощью команды 3.1.2 находим файл формата «название файла.obj». Такой формат файла можно открыть с помощью любого стороннего ПО, поддерживающего открытие данного формата (Blender, ,[Modo](https://ru.wikipedia.org/wiki/Modo_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0)" \o "Modo (программа)), [Cinema 4D](https://ru.wikipedia.org/wiki/Cinema_4D" \o "Cinema 4D)), в том числе и программой визуализации GMESH Visualizer 1.0.

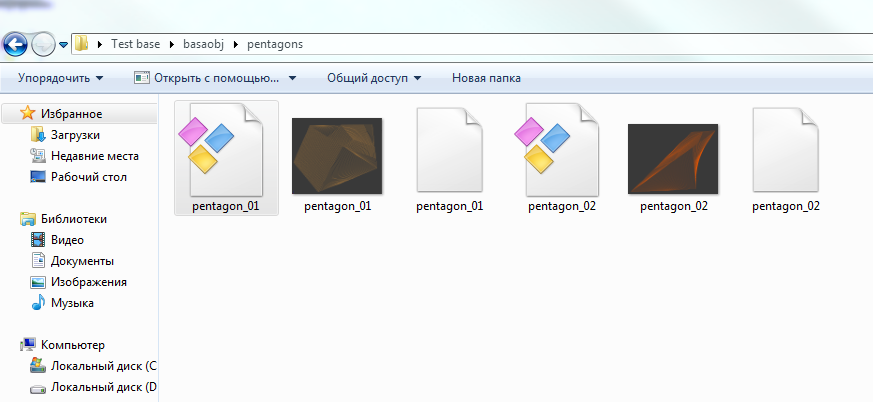


Рис.1 Ищем файл со сгенерированной сеткой.

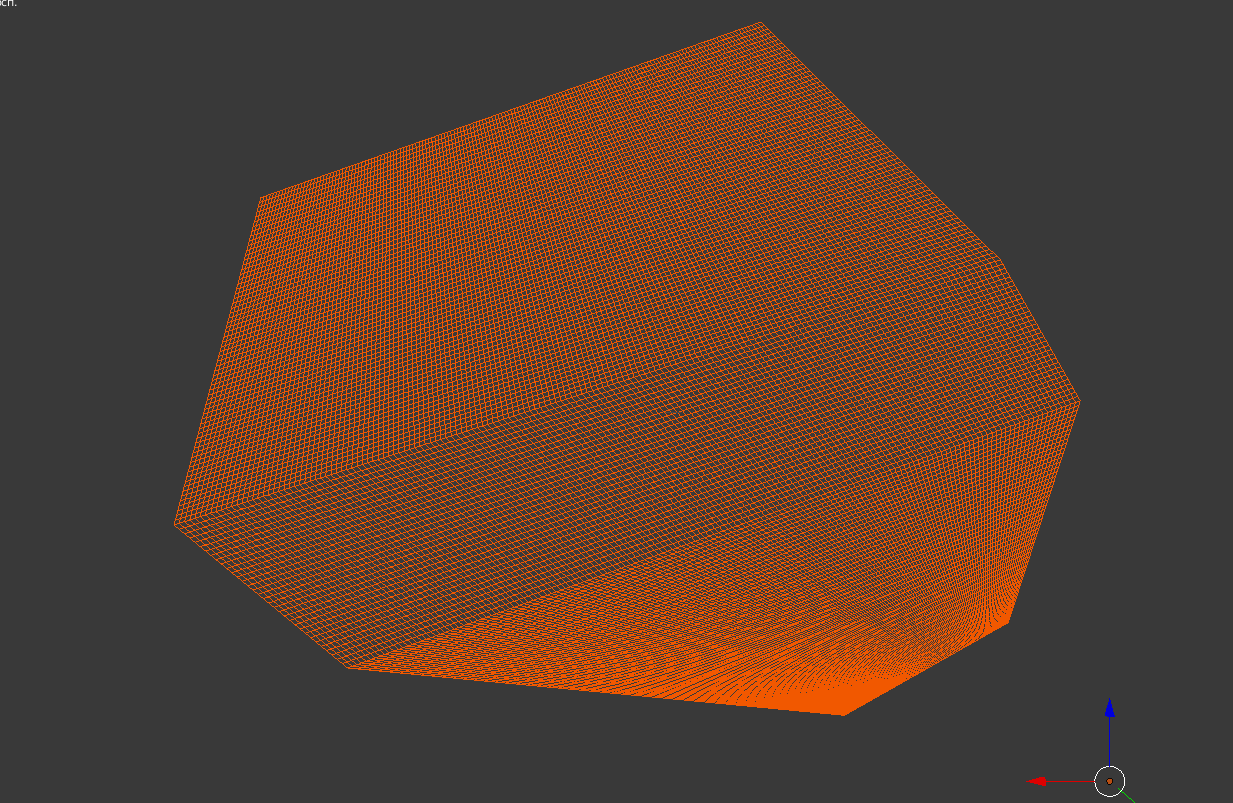


Рис 2. Сгенерированная сетка, сохраненная в формате .obj и открытая с в ПО Blender.

## 3.2. Используемые форматы

**TXT -** в файл формата .txt выводится вещественное число из диапазона [0, 1], чем ближе число к 1 тем качественнее сетка ( ячейки более квадратные).

**XML** – заполнение файла на примере треугольника c вершинами (539, 99) (757, 308) (506, 436)

<?xml version="1.0"?>

<Gmesh xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">

<Poligons>

<Poligon>

<Curves>

<Curve id="0" type="line">

<points>

<ID>0</ID>

<ID>1</ID>

</points>

</Curve>

<Curve id="1" type="line">

<points>

<ID>1</ID>

<ID>2</ID>

</points>

</Curve>

<Curve id="2" type="line">

<points>

<ID>2</ID>

<ID>0</ID>

</points>

</Curve>

</Curves>

<Points>

<Point id="0" x="539" y="99" />

<Point id="1" x="757" y="308" />

<Point id="2" x="506" y="436" />

</Points>

</Poligon>

</Poligons>

</Gmesh>

Ссылка на спецификацию https://www.w3.org/TR/xml11/

**OBJ**

* пример структуры obj файла, в котором хранится 4х-угольник с вершинами (0,0), (10,0)(10,10)(0,10), соединенными отрезками.

Инициализация вершины начинается с символа "v" далее записываются координаты вершины х у z через пробел, в случае если координата вещественная она записывается через ".", к примеру "v 10 50.25 0".

Инициализация стороны n-угольника начинается с символа "l" далее записываются через пробел порядковые номера вершин , которые образуют сторону n-угольника ( отрезок).

v 0 0 0 #вершина №1

v 10 0 0 # вершина №2

v 10 10 0 # вершина №3

v 0 10 0 # вершина №4

g all

s 1

l 1 2

l 2 3

l 3 4

l 4 1

* пример сетки сохраненной в формате .obj. Сперва инициализируется список узлов ( послойно), затем инициализируются отрезки (см. Рис.1)

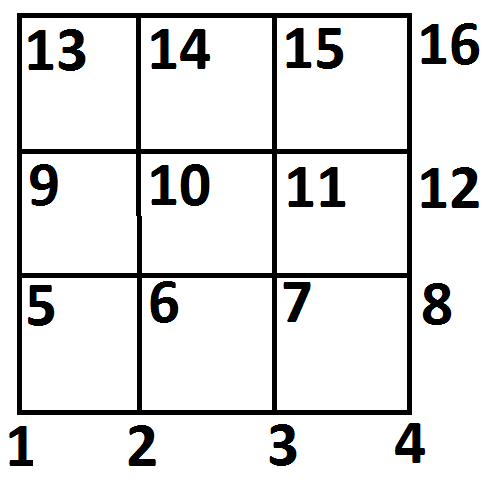


Рис. 1

# прямоугольник с сеткой. Вершины прямоугольника (0,0) (9,0) (9,4) (0, 4)

v 0 0 0# 1ая вершина 1ого слоя

v 3 0 0 # 2ая вершина 1ого слоя

v 6 0 0

v 9 0 0

v 0 2 0 # 1ая вершина 2ого слоя

v 3 2 0

v 6 2 0

v 9 2 0

v 0 4 0

v 3 4 0

v 6 4 0

v 9 4 0 # 4ая вершина 3ого слоя

g all

s 1

l 1 2

l 1 5

l 2 3

l 2 6

l 3 4

l 3 7

l 4 8

l 5 6

l 5 9

l 6 7

l 6 10

l 7 8

l 7 11

l 8 12

l 9 10

l 10 11

l 11 12

Ссылка на спецификацию http://www.martinreddy.net/gfx/3d/OBJ.spec

## 3.3. Завершение программы

3.3.1. При вызове программы из командной строки, после генерации окно программы ( командная строка) автоматически закрывается, если не указан соответствующий ключ(“-nc”);

3.3.2. Можно закрыть стандартным образом (нажав на крестик в верхнем правом углу командной строки) или введя соответствующий ключ (“-с”).