ПРОГРАММА И МЕТОДИКА КОМПЛЕКСНЫХ ИСПЫТАНИЙ

**«Программный инструмент генерации регулярных сеток для выпуклых n-угольников (ПО GMESH 1.1)»**

|  |
| --- |
| Дронин В.И.,  Горшкова Д.П.,  Киракосян А.О.,  Климов А.И.,  Кротова Е.А.,  Федотов А.Д. |

Исполнители: коллектив студентов кафедры ИАНИ ИИТММ ННГУ

2017 г.

Оглавление

[1.Объект испытаний 3](#_Toc473582647)

[2. Цель испытаний 3](#_Toc473582648)

[3. ОРГАНИЗАЦИЯ ПРОВЕДЕНИЯ ИСПЫТАНИЙ 3](#_Toc473582649)

[4. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ 4](#_Toc473582650)

[5. Средства и ПОРЯДОК ИСПЫТАНИЙ 5](#_Toc473582651)

[6. МетодИКА испытаний 5](#_Toc473582652)

[7. ОТЧЕТНОСТЬ 11](#_Toc473582653)

[ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ 11](#_Toc473582654)

Настоящая программа и методика приемочных испытаний определяют порядок проведения комплексных испытаний программного инструмента генерации регулярных сеток для выпуклых n-угольников (ПО «GMESH 1.1»), разработанных согласно техническому заданию.

# 1.Объект испытаний

Испытанию подлежит программный инструмент генерации регулярных сеток для выпуклых n-угольников (ПО «GMESH 1.1»).

# 2. Цель испытаний

Испытания проводятся с целью проверки программного обеспечения «GMESH 1.1» на соответствие требованиям технического задания и выполнение задач, использующих расчеты методом конечных элементов и требующих разбиения исследуемой области на конечные элементы.

# 3. ОРГАНИЗАЦИЯ ПРОВЕДЕНИЯ ИСПЫТАНИЙ

Испытания проводятся силами и средствами ННГУ. Порядок проведения проверок и отдельные пункты программы могут изменяться или уточняться в процессе испытаний.

# 4. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

Испытания проводятся в соответствии с пунктами методики испытаний приведенными в табл. 1

Таблица 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Наименование проверки | | Пункт ТЗ | Пункт методики |
| 1. | Функции подсистемы генерации «GMESH Generator 1.0» | Ввод исходных данных через файл формата XML и OBJ. | 2.8.2.1.1 | 6.1 |
| Сохранение файла (в формате .obj) регулярной сетки декомпозированного n-угольника в желаемую директорию. | 2.8.2.1.2 | 6.2 |
| Генерация регулярной сетки для декомпозированного многоугольника. | 2.8.2.1.3 | 6.3 |
| Анализ регулярной сетки для декомпозированного n-угольника на качество регулярной сетки и вывод информации в файл формата TXT. | 2.8.2.1.4 | 6.4 |
| Поддержка ключей запуска, перечисленных в приложении, при запуске из командной строки. | 2.8.2.1.5 | 6.5 |
| 2. | Функции подсистемы визуализации «GMESH Visualizer 1.0» | Открытие регулярной сетки из файла формата OBJ. | 2.8.2.2.1 | 6.6 |
| Открытие контура из файлов форматов XML и OBJ. | 2.8.2.2.2 | 6.7 |
| Визуализация регулярной сетки и контура. | 2.8.2.2.3 | 6.8 |
| Произведение оценки качества регулярной сетки. | 2.8.2.2.4 | 6.9 |
| Проведение анализа регулярной сетки на согласованность и вывод результатов на элементы пользовательского интерфейса. | 2.8.2.2.5 | 6.10 |
| Проведение анализа регулярной сетки на наличие разрывов и вывод результатов на элементы пользовательского интерфейса. | 2.8.2.2.6 | 6.11 |

# 5. Средства и ПОРЯДОК ИСПЫТАНИЙ

Программные средства испытаний ПО «GMESH 1.1» включают в себя:

* общее программное обеспечение – операционную систему Windows 7/8.1,10 (32 или 64-разрядная) с установленной средой Microsoft .net 3.5 и SQLServerManagementStudio 2015;
* тестовое программное обеспечение – файлы .xml, .obj, располагаемые в данном каталоге: GMESH\ Documents\ Test\_base.

# 

# 6. МетодИКА испытаний

Для демонстрации методики испытаний используется тестовый файл «triangle\_01» формата .xml, располагающийся в каталоге GMESH\ Documents\ Test\_base\triangles. Файл содержит описание треугольника с координатами вершин - (204,95), (377,95), (285,255). Описание формата .xml файла можно посмотреть в руководстве оператора.

**6.1.** **Ввод исходных данных через файл формата XML и OBJ.**

* Запустить консольное приложение GMESH Generator;
* Ввести ключ «–o» в командную строку, указать путь, имя файла (Пример: -o “GMESH\Documents\Test\_base\triangles\triangle\_01.xml”);
* Нажать ENTER;
* В том же каталоге, где находится файл с исходными данными (Пример: GMESH\Documents\Test\_base\triangles \triangle\_01.xml), будет создан файл формата .obj, содержащий сгенерированную сетку. К имени файла будет добавлена строка «MESH»

(Пример:

Исходный файл: “GMESH\Documents\Test\_base\triangles\triangle\_01.xml”

Сохраняемый файл: “GMESH\Documents\Test\_base\triangles\triangle\_01MESH.obj”).

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется консольное окно приложения GMESH Generator;
2. После введения ключа, указания пути и имени файла и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

1. В каталоге, где находится файл с исходными данными (Пример: “GMESH\Documents\Test\_base\triangles \triangle\_01.xml”), будет создан файл, формата obj, содержащий сгенерированную сетку (Пример: “GMESH\Documents\Test\_base\triangles\ triangle\_01MESH.obj”).

**6.2 Сохранение файла (в формате .obj) регулярной сетки декомпозированного n-угольника в желаемую директорию.**

* Запустить консольное приложение GMESH Generator;

**(Внимание! Все выбранные вами ключи следует вводить в одну строку)**

* Выполнить пункт методики 6.1;
* Ввести ключ сохранения «–s» в командную строку, указать желаемую директорию и желаемое имя файла (Пример: -s “GMESH\triangle1.obj”);
* Нажать ENTER;
* В указанной директории будет создан файл с указанным именем, содержащий сгенерированную сетку (Пример: в директории GMESH\triangle1.obj будет создан файл triangle1. obj).
* Если ключ сохранения не указывается, то файл регулярной сетки сохраняется в ту же директорию, где находится файл с исходными данными (Пример: GMESH\Documents\Test\_base\triangles \triangle\_01.xml). В случае по умолчанию к имени файла будет добавлена строка «MESH».

(Пример:

Исходный файл: “GMESH\Documents\Test\_base\triangles\triangle\_01.xml”

Сохраняемый файл: “GMESH\Documents\Test\_base\triangles\triangle\_01MESH.obj”).

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется консольное окно приложения GMESH Generator;
2. После введения ключа «-s», указания желаемой директории и желаемого имени файла и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

1. В указанной директории создан файл с указанным именем, содержащий сгенерированную сетку (Пример: в директории GMESH\triangle1.obj будет создан файл triangle1. obj).

**6.3 Генерация регулярной сетки для декомпозированного многоугольника.**

* Запустить консольное приложение GMESH-1.1;
* Выполнить пункт методики 6.1;
* Декомпозиция многоугольника проводится автоматически, без помощи ключей;
* Полученный в результате файл формата \*obj открыть в стороннем ПО «Open 3D Model Viewer» и увидеть результат работы генератора.

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется консольное окно приложения GMESH Generator;
2. После введения ключа «-o», указания пути и имени файла в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

1. При открытии полученного файла формата .obj в стороннем ПО «Open 3D Model Viewer» присутствует изображение сетки;

**6.4 Анализ регулярной сетки для декомпозированного n-угольника на качество регулярной сетки и вывод информации в файл формата TXT.**

* Выполнить пункт методики 6.1;
* Ввести ключ «–q»;
* Нажать ENTER.

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется консольное окно приложения GMESH Generator;
2. После введения ключа «-q» и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

1. В директории, где располагается файл со сгенерированной сеткой, будет создан файл формата .txt (Пример: GMESH\triangle\_01GradAnalize.txt), который будет содержать число (Пример: в документе triangle\_01GradAnalize.txt данное число будет равняться 0,76183969524613), означающее качество регулярной сетки.

**6.5 Поддержка ключей запуска, перечисленных в приложении, при запуске из командной строки.**

* Запустить консольное приложение GMESH-1.1;
* Использовать доступные команды;
* Нажать ENTER.

Доступные команды:

* «-help» - выводит справочную информацию о доступных командах;
* «-cl» - завершает сеанс программы. Если стоит в конце набора параметров, то завершает сеанс программы после их исполнения;
* «-nc» - оставляет командную строку в ждущем режиме;
* «-o» - открытие файла формата .xml или .obj по указанному пути (см. пункт методики 6.1);
* «-s» - указание директории для сохранения файла сгенерированной сетки в формате .obj. По умолчанию сохраняется в директорию исходных данных (см. пункт методики 6.2);
* «-q» - вывод качества сетки в файл формата .txt в директории сгенерированной сетки (см. пункт методики 6.4).

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется консольное окно приложения GMESH Generator;
2. При использовании ключа «-help» и нажатия ENTER выводится справочная информация о доступных командах;
3. При использовании ключа «-cl» и нажатия ENTER завершается сеанс программы. Если стоит в конце набора параметров, то завершается сеанс программы после их исполнения;
4. При использовании ключа «-nc» и нажатия ENTER командная строка остается в ждущем режиме;
5. При использовании ключа «-o»:
   1. После введения ключа, указания пути и имени файла и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

* 1. В каталоге, где находится файл с исходными данными (Пример: “GMESH\Documents\Test\_base\triangles\triangle\_01.xml”), будет создан файл, формата obj, содержащий сгенерированную сетку (Пример: “GMESH\Documents\Test\_base\triangles\triangle\_01MESH.obj”).

1. При использовании ключа «-s»:
   1. После введения ключа «-s», указания желаемой директории и желаемого имени файла и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

* 1. В указанной директории будет создан файл с указанным именем, содержащий сгенерированную сетку (Пример: в директории GMESH\triangle1.obj будет создан файл triangle1. obj).

1. При использовании ключа «-q»:
   1. После введения ключа «-q» и нажатия ENTER в консольном окне отобразится следующее текстовое сообщение:

«Читаем файл…

Готово  
Декомпозируем треугольник…

Генерируем сетку…

Пишем сетку…»

* 1. В директории, где располагается файл со сгенерированной сеткой, будет создан файл формата .txt (Пример: GMESH\triangle\_01GradAnalize.txt), который будет содержать число (Пример: в документе triangle\_01GradAnalize.txt данное число будет равняться 0,76183969524613), означающее качество регулярной сетки.

**6.6 Открытие регулярной сетки из файла формата OBJ**

* Запустить приложение «GMESH Visualizer 1.0»;
* В подменю File выбрать вкладку Open и выбрать Mesh, затем выбрать расположение декомпозированного n-угольника.

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. Получена требуемая сетка.

**6.7 Открытие контура из файлов форматов XML и OBJ**

* Запустить приложение «GMESH Visualizer 1.0»;
* В подменю File выбрать вкладку Open и выбрать Contour, затем выбрать расположение n-угольника.

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. Получен контур требуемого n-угольника.

**6.8 Визуализация регулярной сетки и контура.**

* Выполнить пункт методики 6.6;

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. Получен контур требуемого n-угольника;
3. Получено отображение линий декомпозиции;
4. Получена регулярная сетка.

**6.9 Произведение оценки качества регулярной сетки**

* Выполнить пункт методики 6.6;

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. Получен требуемый n-угольник;
3. Получено отображение линий декомпозиции;
4. Получена регулярная сетка.
5. Отображение значения оценки качества сетки находится в поле MESH Quality. В зависимости от результата система заполняет ячейку определенным цветом в градации от красного до светло-зеленого. Где красный цвет означает плохое качество, зеленый – отличное качество.

**6.10. Проведение анализа регулярной сетки на согласованность и вывод результатов на элементы пользовательского интерфейса.**

* Выполнить пункт методики 6.6;

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. В поле справа выводится результат проведения анализа n-угольника на согласованность регулярной сетки.

**6.11. Проведение анализа регулярной сетки на наличие разрывов и вывод результатов на элементы пользовательского интерфейса.**

* Выполнить пункт методики 6.6;

Результат испытания считается положительным, если будут выполнены условия:

1. Откроется главная форма ПО «GMESH Visualizer 1.0»;
2. В поле справа выводится результат проведения анализа n-угольника на наличие разрывов в регулярной сетке.

# 7. ОТЧЕТНОСТЬ

По всем видам испытаний оформляется протокол испытаний, который подписывается членами комиссии, проводившими испытания.

Протокол испытаний утверждается председателем комиссии.

# 

# ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ

1. ТЗ на научно-исследовательскую работу ««Программный инструмент генерации регулярных сеток для выпуклых n-угольников» (ПО GMESH-1.1);
2. Руководство оператора «GMESH Generator»;
3. Руководство оператора «GMESH Visualizator»;
4. Программное обеспечение «GMESH-1.1».