Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

УТВЕРЖДАЮ:

Зав. кафедрой ПОАС

\_\_\_\_\_\_\_\_\_\_ Орлова Ю.А.

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

**КАЧЕСТВО И НАДЕЖНОСТЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

«Отображение сложной структуры»

Программа и методика испытаний

СОГЛАСОВАНО: Разработчик:

Руководитель работы: Студент ПрИн-266

Доцент кафедры ПОАС \_\_\_\_\_\_\_\_\_\_ Пехтелев Е. П.

\_\_\_\_\_\_\_\_\_\_ Сычев О.А. «\_\_\_»\_\_\_\_\_\_\_ 2020 г.

«\_\_\_»\_\_\_\_\_\_\_ 2020 г.

Нормоконтроллер:

Преподаватель кафедры ПОАС \_\_\_\_\_\_\_\_\_\_ Матюшечкин Д.С.

«\_\_\_»\_\_\_\_\_\_\_ 2020 г.

2020

Содержание

[1 Объект испытаний 3](#_Toc52971798)

[2 Цель испытаний 3](#_Toc52971799)

[3 Требования к программе 3](#_Toc52971800)

[4 Требования к программной документации 3](#_Toc52971801)

[5 Средства и порядок испытаний 4](#_Toc52971802)

[6 Методы испытаний 4](#_Toc52971803)

[Приложение А. Методика тестирования функции parceStruct 5](#_Toc52971804)

[Приложение Б. Методика тестирования функции parceExp 8](#_Toc52971805)

[Приложение В. Методика тестирования функции main 12](#_Toc52971806)

# Объект испытаний

Программа называется «StructureView» (Программа для нахождения и отображения части структуры, описываемой выражением).

# Цель испытаний

Целью испытаний является проверка соответствия программы требованиям к функциональным характеристикам.

# Требования к программе

Требования к программе изложены в Техническом задании в пункте 3.

# Требования к программной документации

Программная документация должна быть представлена в бумажной форме в виде:

* техническое задание;
* технический проект в виде описания программы;
* программа и методика испытаний;
* руководство программиста.

Программная документация должна быть представлена в электронной форме в виде:

* копии всех документов бумажной формы;
* рабочая документация;
* текст программы.

Вся документация должна быть составлена в соответствии с ГОСТ 19.

# Средства и порядок испытаний

Для запуска тестов необходимо подключение библиотеки QTestLib к проекту программы.

# Методы испытаний

Перечень проверок для:

* функции обработки данных о структуре parceStruct() представлен в приложении А;
* функции обработки выражения parceExp() представлен в приложении Б;
* главной функции main - в приложении В.

Приложение А

Методика тестирования функции parceStruct()

Входные данные:

Функция берёт документ json из поля класса

Выходные данные:

Результат функции помещаются в поле error если присутствует ошибка или в поле класса если данные обработаны без ошибок

void DataStruct::parceStruct()

|  |  |  |
| --- | --- | --- |
| Входные данные | | Ожидаемый результат |
| 1 | В данных не указано название структуры | |
| {  "objectName": "building",  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  }  }  } | | structName:  objectName:  elemType:  elemName:  elemValue:  elemIsStatic:  error: "Отсутствует название структуры" |
| 2 | В данных не указано название объекта структуры | |
| {  "structName": "process",  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  }  }  } | | structName:”process”  objectName:  elemType:  elemName:  elemValue:  elemIsStatic:  error: "Отсутствует имя объекта структуры" |
| 3 | В данных структуры не указано хотя бы одного свойства или метода структуры | |
| {  "structName": "process",  "objectName": "building"  } | | structName: "process"  objectName: "building"  elemType:  elemName:  elemValue:  elemIsStatic:  error: "Отсутствуют методы или параметры" |
| 4 | Default тест | |
| {  "structName": "process",  "objectName": "building",  "properties": {  "variable1": {  "type": "int",  "name": "number",  "value": 3,  "isStatic":true  },  "variable2": {  "type": "int",  "name": "number2",  "value": 9,  "isStatic":false  }  },  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  }  }  } | | structName: "process"  objectName: "building"  elemType: “int”, “int”, “bool”  elemName: “number”, “number2”, “isCheck()”  elemValue: 3, 9, 1  elemIsStatic: true, false, true  error: |
| 5 | Только переменные | |
| {  "structName": "process",  "objectName": "building",  "properties": {  "variable1": {  "type": "int",  "name": "number",  "value": 3,  "isStatic":true  },  "variable2": {  "type": "int",  "name": "number2",  "value": 9,  "isStatic":false  }  }} | | structName: "process"  objectName: "building"  elemType: “int”, “int”  elemName: “number”, “number2”  elemValue: 3, 9  elemIsStatic: true, false  error: |
| 6 | Только методы | |
| {  "structName": "process",  "objectName": "building",  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  },  "variable2": {  "type": "int",  "name": "getNum()",  "arguments": "void",  "isStatic":false  }  }  } | | structName: "process"  objectName: "building"  elemType: “bool”, “int”,  elemName: “isCheck()”, “getNum()”  elemValue: 1, 1  elemIsStatic: true, false  error: |

Приложение Б

Методика тестирования функции parceExp()

Входные данные:

Ссылка на данные о структуре

Функция берёт выражение из поля класса

Выходные данные:

Результат функции помещаются в поле error если присутствует ошибка или в поле result если данные обработаны без ошибок

Возвращает true если выражение обработано без ошибок, иначе false

bool Expression::parceExp(DataStruct\* curDataStruct)

|  |  |  |
| --- | --- | --- |
| Входные данные | | Ожидаемый результат |
| 1 | Недостаточно операндов для сложения | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  1 + | | result:  error: “Недостаточно операндов для сложения” |
| 2 | Недостаточно операндов для разности | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  5 - | | result:  error: “Недостаточно операндов для разности” |
| 3 | Недостаточно операндов для произведения | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  5 6 + \* | | result:  error: “Недостаточно операндов для произведения” |
| 4 | Недостаточно операндов для деления | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  building number2 . / | | result: “number2”  error: “Недостаточно операндов для деления” |
| 5 | Недостаточно операндов для инкрементации | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  ++ | | result:  error: “Недостаточно операндов для инкрементации” |
| 6 | Недостаточно операндов для декрементации | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  -- | | result:  error: “Недостаточно операндов для декрементации” |
| 7 | Недостаточно операндов для обращения к элементу структуры | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  process :: | | result:  error: “ Недостаточно операндов для обращения к элементу структуры ” |
| 8 | Один или оба операнда являются неизвестной переменной | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  a count + | | result:  error: “ Неизвестные переменные ” |
| 9 | Операнд является переменной, не относящейся к структуре | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  plot ++ | | result:  error: “ Неизвестная переменная ” |
| 10 | Обращение к статическому элементу как к динамическому | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  building number . | | result:  error: “ Обращение к статическому элементу ” |
| 11 | Обращение к нестатическому элементу как к статическому | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  process number2 :: | | result:  error: “ Обращение к нестатическому элементу ” |
| 12 | Обращение по неверному имени структуры | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  destroy number :: | | result:  error: “ Обращение по неверному имени структуры ” |
| 13 | Обращение по неверному имени объекта | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  destroy number2 . | | result:  error: “ Обращение по неверному имени объекта ” |
| 13 | Переменная не является элементом структуры | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  process count :: | | result:  error: “ count не является элементом структуры ” |
| 14 | Default тест | |
| structName: "process"  objectName: "building"  elemName: “number”, “number2”, “isCheck()”  elemIsStatic: true, false, true  process count :: building getFloat() . building number2 . + \* | | result: “count”, “getFloat()”, “number2”  error: |

Приложение В

Методика тестирования функции main

Входные данные:

argc - количество переданных аргументов командной строки

argv[] – аргрументы командной строки:

argv[1] - путь к файлу

Выходные данные:

Если функция завершена без ошибок, то сохраняет в корень папки изображение, иначе выводит сообщение об ошибке

int main(int argc, char \*argv[])

|  |  |  |
| --- | --- | --- |
| Входные данные | | Ожидаемый результат |
| 1 | Ни один из элементов структуры не задействован в выражении | |
| 1 2 + 3 + 34 \* ++  {  "structName": "process",  "objectName": "building",  "properties": {  "variable1": {  "type": "int",  "name": "number",  "value": 33,  "isStatic":true  },  "variable2": {  "type": "int",  "name": "number2",  "value": 9,  "isStatic":false  },  "variable3": {  "type": "int",  "name": "count",  "value": 13,  "isStatic":true  },  "variable4": {  "type": "int",  "name": "minimum",  "value": 532,  "isStatic":false  }  },  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  },  "variable2": {  "type": "float",  "name": "getFloat()",  "arguments": "void",  "isStatic":false  },  "variable3": {  "type": "int",  "name": "getNumber()",  "arguments": "void",  "isStatic":true  }  }  } | | D:\build-StructureView-Desktop_Qt_5_12_6_MinGW_64_bit-Debug\test5.png |
| 2 | Все элементы структуры задействованы | |
| process number :: process count :: + process isCheck() :: - process getNumber() :: building number2 . / / building minimum . \* building getFloat() . -  {  "structName": "process",  "objectName": "building",  "properties": {  "variable1": {  "type": "int",  "name": "number",  "value": 33,  "isStatic":true  },  "variable2": {  "type": "int",  "name": "number2",  "value": 9,  "isStatic":false  },  "variable3": {  "type": "int",  "name": "count",  "value": 13,  "isStatic":true  },  "variable4": {  "type": "int",  "name": "minimum",  "value": 532,  "isStatic":false  }  },  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  },  "variable2": {  "type": "float",  "name": "getFloat()",  "arguments": "void",  "isStatic":false  },  "variable3": {  "type": "int",  "name": "getNumber()",  "arguments": "void",  "isStatic":true  }  }  } | | D:\build-StructureView-Desktop_Qt_5_12_6_MinGW_64_bit-Debug\test6.png |
| 3 | Default тест | |
| process count :: building getFloat() . building number2 . + \*  {  "structName": "process",  "objectName": "building",  "properties": {  "variable1": {  "type": "int",  "name": "number",  "value": 33,  "isStatic":true  },  "variable2": {  "type": "int",  "name": "number2",  "value": 9,  "isStatic":false  },  "variable3": {  "type": "int",  "name": "count",  "value": 13,  "isStatic":true  },  "variable4": {  "type": "int",  "name": "minimum",  "value": 532,  "isStatic":false  }  },  "methods": {  "variable1": {  "type": "bool",  "name": "isCheck()",  "arguments": "void",  "isStatic":true  },  "variable2": {  "type": "float",  "name": "getFloat()",  "arguments": "void",  "isStatic":false  },  "variable3": {  "type": "int",  "name": "getNumber()",  "arguments": "void",  "isStatic":true  }  }  } | | D:\build-StructureView-Desktop_Qt_5_12_6_MinGW_64_bit-Debug\test3.png |