Прототип инструмента для получения и анализа графа потока управления открытых исходных текстов C/C++ на основе универсального промежуточного представления

Пустыгин А. Н.
Ковалевский А. А.
Огуречникова Е. А.
Субботин А. А.

Челябинск, 2013

Цели написания прототипа

1.Проба существующих поддерживаемых инструментов извлечения из компилятора C/C++ внутренней модели исходного текста- дерева разбора
2 Проверка введенных промежуточного и эквивалентных представлений при анализе существующих проектов
3 Оценка степени сложности использования предложенных эквивалентных представлений для статического анализа
4 Проба графической интерпретации эквивалентных представлений для отображения результатов обработки эквивалентных представлений

Известные ближайшие аналоги

Прототип XBLiteGen – написан для построения блок-схем алгоритмов на языке- клоне BASIC Получено свидетельство о госрегистрации

Частичные решения:

- Elsa, Bison, Flawfinder, GCC-XML
- scrML, JavaML, c2xml, C++2XML
- Dehydra, Treegydra
- Clang

Выбранные условия разработки

- ОС Ubuntu 10.04
- Язык C/C++
- Компилятор clang
  http://clang.llvm.org/
- Библиотека libclang 3.2,
  из проекта Clang

Общий алгоритм функционирования прототипа

- Обход дерева разбора компилятора производится через интерфейс «посетитель», предоставляемый библиотекой clang (фронт-энд компилятора) Узлы AST представлены в виде специальных атрибутированных объектов — курсоров.

Для получения того или иного атрибута курсора, библиотека предоставляет набор функций

Получение срезов диаграммы путем фильтрации по условию с помощью стандартного xml-парсера

Графический Интерфейс прототипа - web-браузер

JavaScript код обеспечивает интерактивность эквивалентного представления (обратную связь с исходным текстом, другими представлениями, задание параметров срезов и их применение, навигация по странице).

Реализованные функции

-Построение эквивалентного представления в виде диаграммы потока управления в HTML-формате

-Возможность получения срезов диаграммы по ряду условий

-Переход в исходный текст из любого блока диаграммы через любой текстовый редактор операционной системы

-Возможность связи эквивалентного представления в виде диаграммы потока с другими эквивалентными представлениями

Атрибуты построения срезов

- имя идентификатора,
- метод доступа (любой, чтение, запись, вызов),
- тип идентификатора (любой, переменная, поле, функция, метод),
- модификатор доступа (любой, public, private, protected),
-имя родительского класса и/или области видимости,
-специальные параметры фильтрации (рекурсивные вызовы, инициализация «мусором», присваивание в условиях, цикл while/do-while, цикл for)

Исходный текст теста на «полноту»

```
#include <stdio.h>
#include <stdlib.h>
#include "tools.h"

using namespace tools;

long fact(long n)
{
  return n > 0 ? n * fact(n - 1) : 1;
}

int main(int argc,char *argv[])
{
        int tool_ids[]={1,2,3};
        Tools *tools=new Tools(tool_ids,sizeof(tool_ids)/sizeof(tool_ids[0]));
        Tool* tool=tools->getTool(argc>1?atoi(argv[1]):1);
        if(!tool)goto error;
        tool->setData((void*)"Test data");
        return 0;
        error:
        puts("Failed to get tool by ID");
        return -1;
}
```

Результаты теста на «производительнсть»

Входные файлы: cxx2xml.cpp, utils.h, Index.h (42 КБ)
Выходные файлы: cxx2xml.xml (350 КБ), cxx2xml.html (2.1 МБ)

Время генерации файла промежуточного представления cxx2xml.xml: 350 Кб за 350 мс

Время генерации файла эквивалентного представления cxx2xml.html: 2.1 Мб за 475 мс

Результаты опытной разработки

- Опробован формат универсального промежуточного представления исходных текстов программ для языка C/C++
- Опробовано применения стандартного фронт-энда компилятора для извлечения дерева разбора
- формат универсального промежуточного представления исходных текстов программ для языка C/C++ использован для получения эквивалентных представлений путем визуализации средствами стандартного web-браузера

Файл промежуточного текстового представления в XML-формате

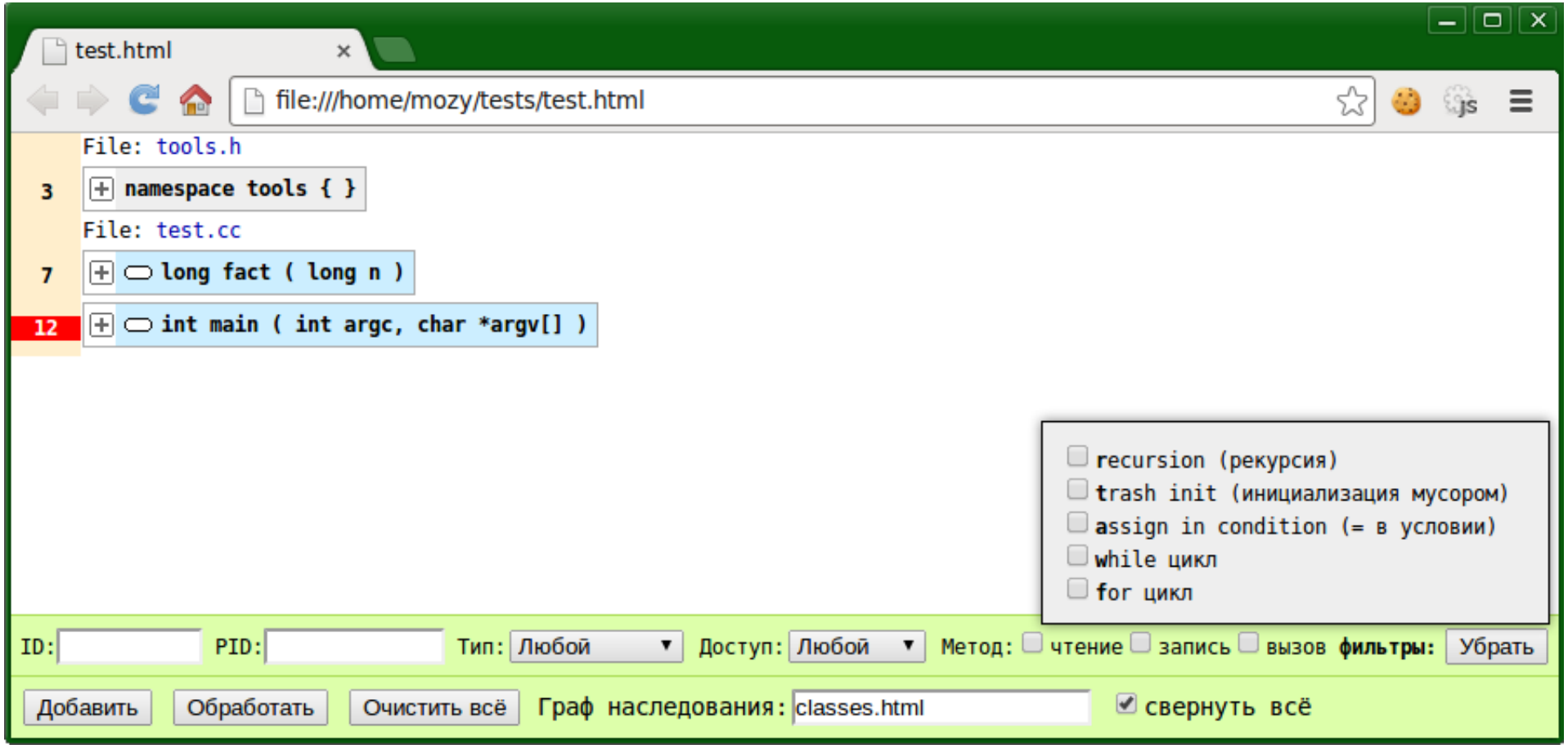
```
1  long fact(long n)
2  {
3      return n > 0 ? n * fact(n - 1) : 1;
4  }
```

Вычисление факториала

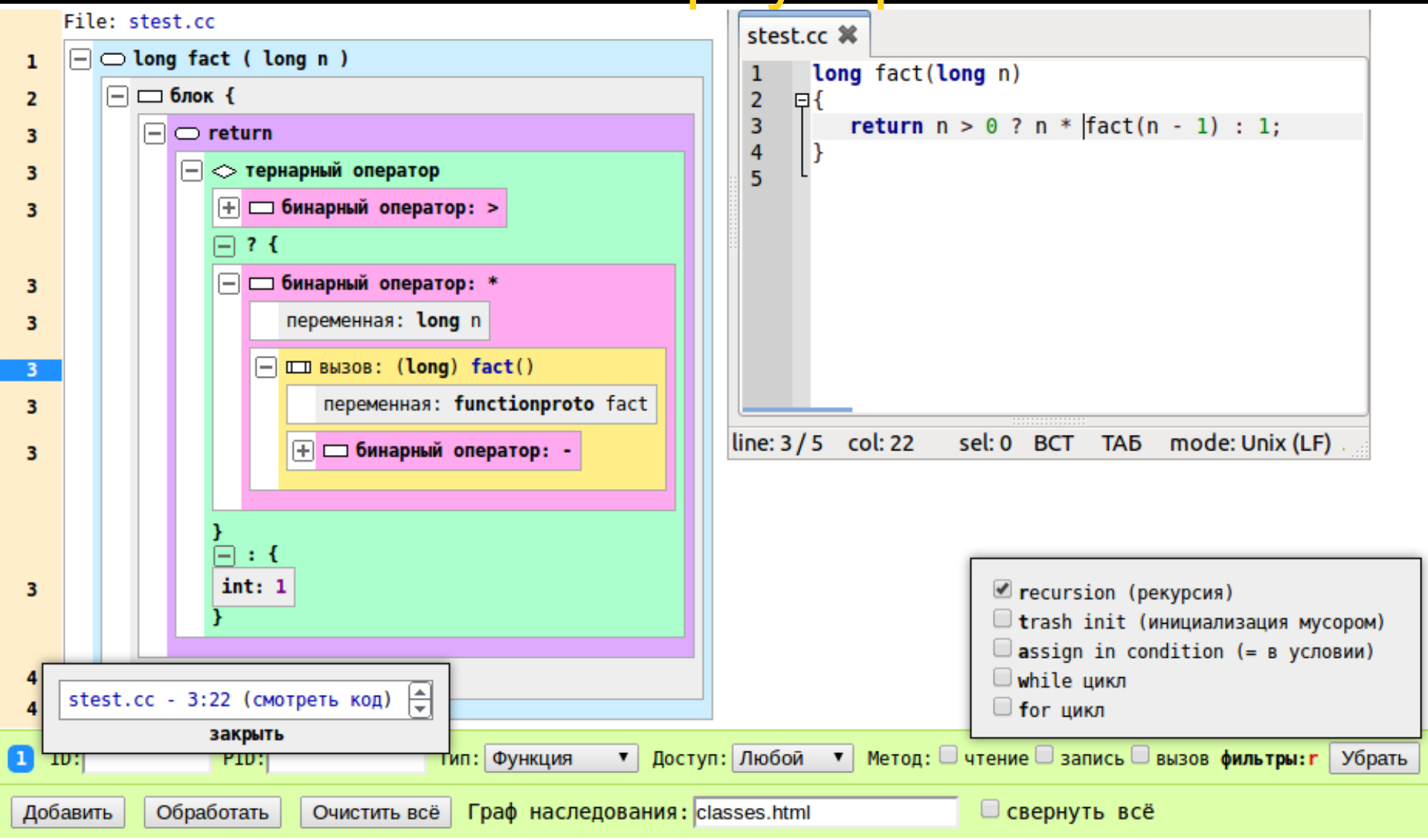
```
▼<files>
 ▼<file name="/home/mozy/tests/test/stest.cc" language="C/C++">
  ▼<function name="fact" rtype="long" argc="1" line="1" col="6" line2="4" col2="2" file="/home/mo
    <argument type="long" name="n" line="1" col="16"/>
   ▼<block line="2" col="1" line2="4" col2="2">
    ▼<return line="3" col="4">
     ▼<terop line="3" col="11">
      ▼<binop type=">" line="3" col="11">
        <var name="n" type="long" line="3" col="11"/>
        <literal value="0" type="int" line="3" col="15"/>
       </binop>
      ▼<binop type="*" line="3" col="19">
        <var name="n" type="long" line="3" col="19"/>
       ▼<call name="fact" rtype="long" line="3" col="23" line2="3" col2="34">
          <var name="fact" type="functionproto" line="3" col="23"/>
         ▼<binop type="-" line="3" col="28">
            <var name="n" type="long" line="3" col="28"/>
            <literal value="1" type="int" line="3" col="32"/>
           </binop>
         </call>
       </binop>
       <literal value="1" type="int" line="3" col="37"/>
     </terop>
    </return>
   </block>
  </function>
```

Эквивалентное представления исходного текста в виде диаграммы потока управления в окне web-браузера

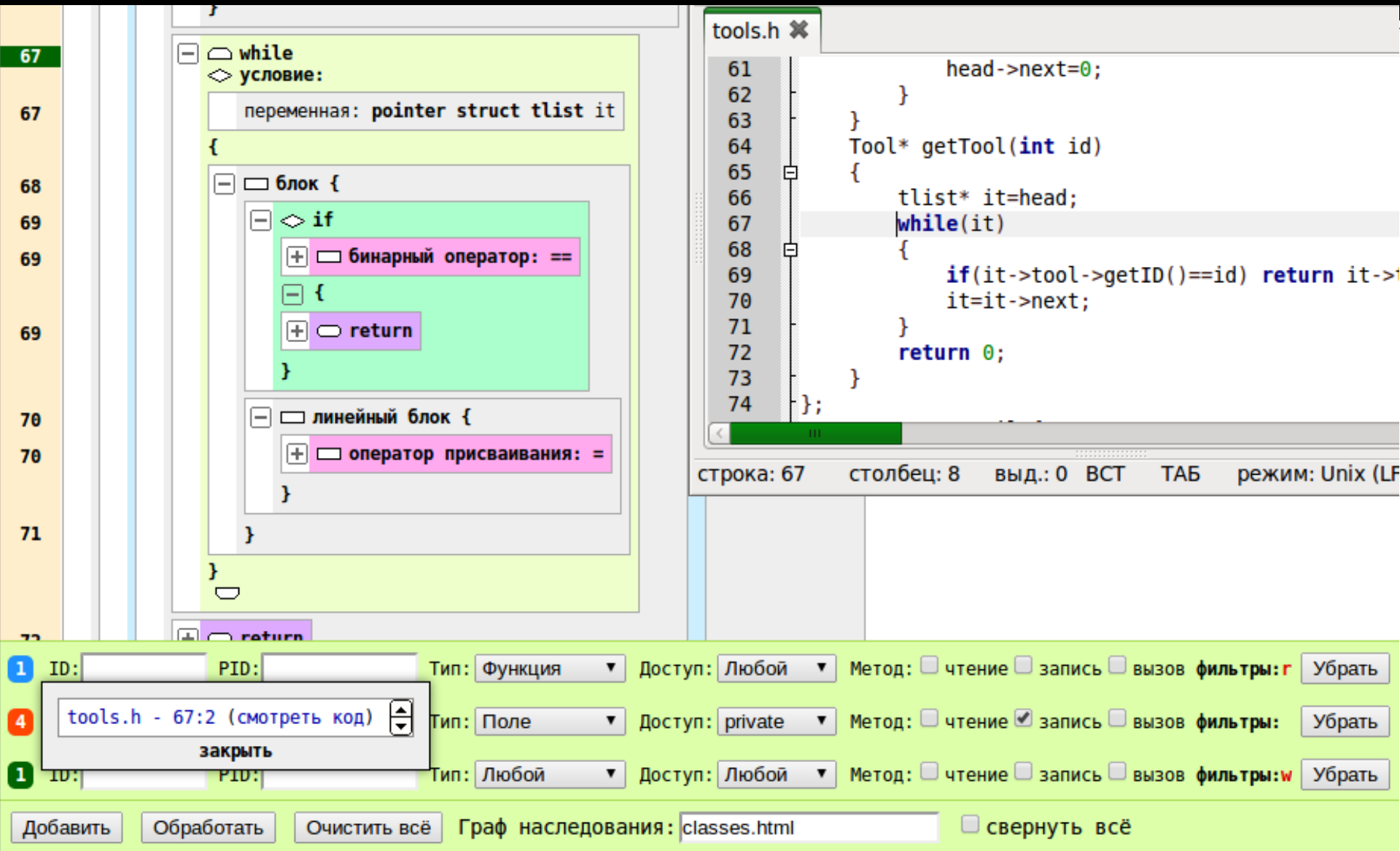
Начальное состояние диаграммы в окне браузера



Эквивалентное представление в форме диаграммы потока управления в окне web-браузера



Эквивалентное представление и навигация к исходному тексту в окне web-браузера



Файл эквивалентного представления в окне текстового редактора Geany

```
<!DOCTYPE HTML>
<html>
<head>
 <meta http-equiv='Content-Type' content='text/html; charset=utf-8' />
 <script type='text/javascript' src='/usr/local/share/conan/graph.js'></script>
 <link rel='stylesheet' type='text/css' href='/usr/local/share/conan/graph.css'>
 </head>
<body>

<div id='wrap'><div class='lineNumbers'><div class='canvas'>
<div class='cur_file'>File: <a href='navtext:geany:/home/mozy/tests/test/stest.cc#1' title='Open in geany'>st
 <span class='block_title'><span class='begin_pic'></span>long fact ( long n )</span></div>
<div style='display:block' id='in_4245298'><div class='inner_block'><table class='block_tbl'><tr><th><span cl
<div style='display:block' id='in_4245307'><div class='inner_block'><table class='block_tbl'><tr><th><span cl
<div style='display:none' id='in_4245311'><div class='inner_block'><table class='block_tbl'><tr><th><span cla
 <span class='block_title'><span class='cond_pic'></span>тернарный оператор</span></div>
<div style='display:block' id='in_4245313'><div class='inner_block'><table class='block_tbl'><tr><th><span cl
<span class='block_title'><span class='block_pic'></span>бинарный оператор: &gt;</span></div><div style='disp
 <span class='block_title_n'>переменная: <b>long</b> n</span></div></div>
 </td></tr></table><div class='block literal' id='4245321'><div class='linenumber'><a href='navtext:geany:/hom
 </div><br>
 </div></div>
 </div>
 </td></tr></table><div>
<span class='close_but' onclick='close_open_block(this,"in_4245325way1");return false;'></span> <span class='
<div style='display:block' id='in_4245325way1'><div class='inner_block'><table class='block_tbl'><tr><th><spa
<span class='block_title'><span class='block_pic'></span>бинарный оператор: *</span></div><div style='display
 <span class='block_title_n'>переменная: <b>long</b> n</span></div></div>
</td></tr></table><table class='block_tbl'><tr><th><span class='open_but' onclick='close_open_block(this,"in_
<span class='block_title_n'><span class='call_pic'></span>вызов: (<b>long</b>) <b><a href='#fact'>fact</a></b
 <span class='block_title_n'>переменная: <b>functionproto</b> fact</span></div></div>
</td></tr></table><table class='block_tbl'><tr><th><span class='open_but' onclick='close_open_block(this,"in_
<span class='block_title'><span class='block_pic'></span>бинарный оператор: -</span></div><div style='display
```

Видео демонстрация