Статический анализ C++: от ненависти до любви

СТАЧКА

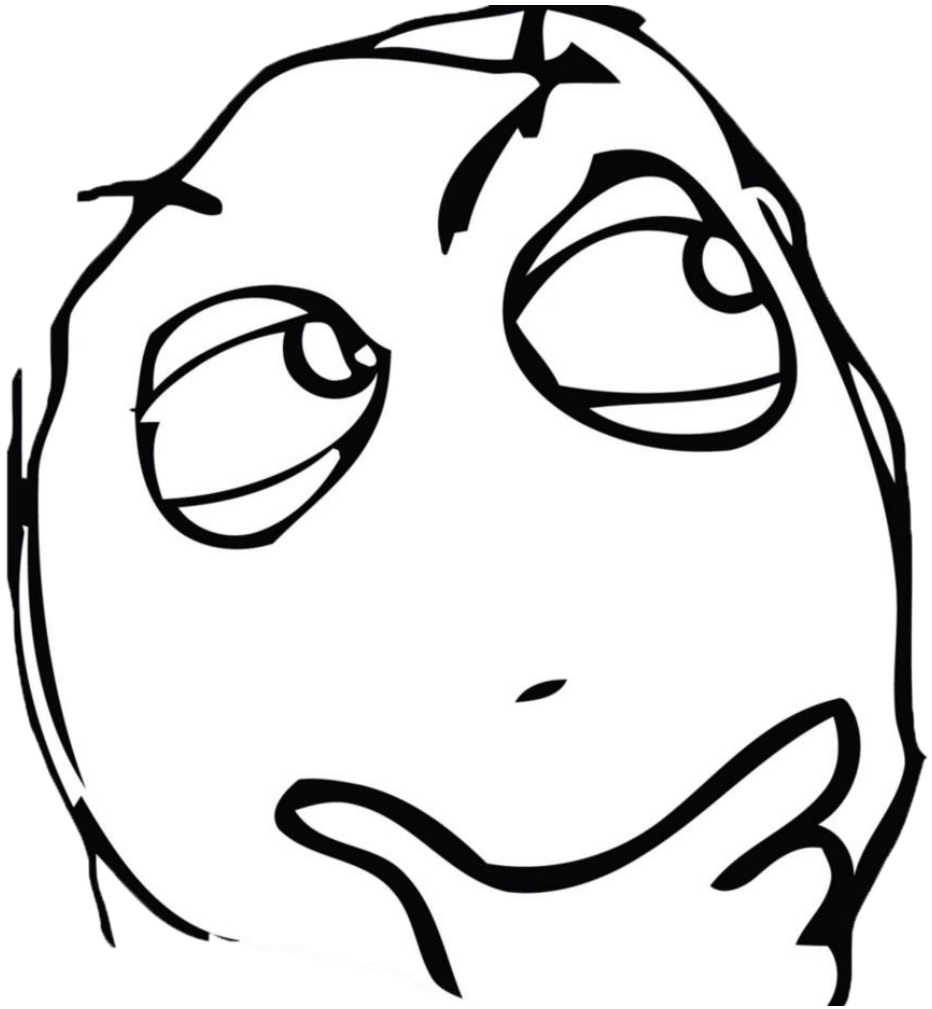
Алексей Горшков
Программист C++

Алексей Горшков

- C++ разработчик в команде PVS-Studio.
- Автор статей о проверках кода в open Source проектах.
- Просто хороший малый.

СТАЧКА

О чем мы сегодня поговорим?



1. **что** **такое** статический анализ и **как** это **работает**.

1. **что** **такое** статический анализ и **как** это **работает**.

2. **чем** статический анализ **отличается** от других технологических решений.

1. **что** **такое** статический анализ и **как** это **работает**.

2. **чем** статический анализ **отличается** от других технологических решений.

3. **что** **может** статический анализ.

1. **что** **такое** статический анализ и **как** это **работает**.

2. **чем** статический анализ **отличается** от других технологических решений.

3. **что** **может** статический анализ.

4. **множество** **примеров** из реальных Open Source проектов.

1. **что** **такое** статический анализ и **как** это **работает**.

2. **чем** статический анализ **отличается** от других технологических решений.

3. **что** **может** статический анализ.

4. **множество** **примеров** из реальных Open Source проектов.

    4.1. только реальный код из реальных проектов.

1. **что** **такое** статический анализ и **как** это **работает**.
2. **чем** статический анализ **отличается** от других технологических решений.
3. **что** **может** статический анализ.
4. **множество** **примеров** из реальных Open Source проектов.

    4.1. только реальный код из реальных проектов.

    4.2. рассматриваем статический анализатор PVS-Studio.

1. **что** **такое** статический анализ и **как** это **работает**.

2. **чем** статический анализ **отличается** от других технологических решений.

3. **что** **может** статический анализ.

4. **множество** **примеров** из реальных Open Source проектов.

   4.1. только реальный код из реальных проектов.

   4.2. рассматриваем статический анализатор PVS-Studio.

5. **интересные** **примеры** из нашей поддержки.

Что такое статический анализ?

Как Шерлок, только круче...

1. Статический анализатор находит в коде **возможные** **ошибки**.

Как Шерлок, только круче...

Clang

1. Стати

```
MapTy PerPtrTopDown;
MapTy PerPtrBottomUp;
void clearBottomUpPointers()
{
  PerPtrTopDown.clear();
}
void clearTopDownPointers()
{
  PerPtrTopDown.clear();
}
```

Как Шерлок, только круче...

Clang

1. Стати

```
MapTy PerPtrTopDown;
MapTy PerPtrBottomUp;
void clearBottomUpPointers()
{
  PerPtrTopDown.clear();   ←   PerPtrBottomUp.clear();
}
void clearTopDownPointers()
{
  PerPtrTopDown.clear();
}
```

Б - Безопасность

1. Статический анализатор находит в коде **возможные ошибки**.

2. Обнаруживает ошибки и **потенциальные уязвимости**.

Б - Безопасность

1. [illegible]
2. [illegible]

- Common Weakness Enumeration(**CWE**);
- SEI Computer Emergency Response Team(**CERT**) Coding Standards;
- Motor Industry Software Reliability Association (**MISRA**);
- AUTomotive Open System ARchitecture (**AUTOSAR**).

Эффективная разработка

1. Статический анализатор находит в коде **возможные ошибки**.

2. Обнаруживает ошибки и **потенциальные уязвимости**.

3. Обнаруживает проблемы на **раннем этапе** разработки.

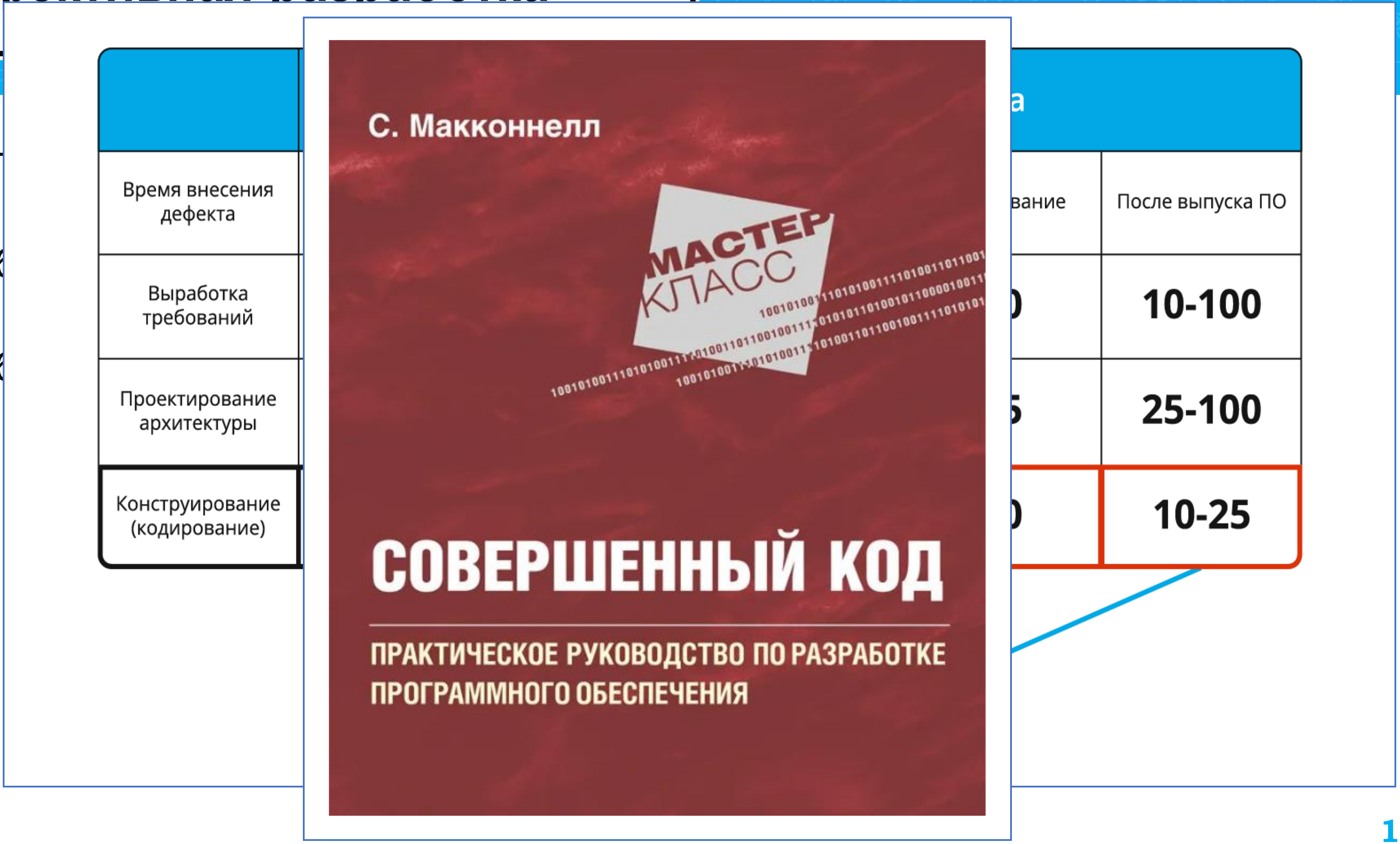
Эффективная разработка

1. С
2. О
3. О

	Время обнаружения дефекта				
Время внесения дефекта	Выработка требований	Проектирование архитектуры	Конструирование (кодирование)	Тестирование	После выпуска ПО
Выработка требований	1	3	5-10	10	10-100
Проектирование архитектуры	-	1	10	15	25-100
Конструирование (кодирование)	-	-	1	10	10-25

Эффективная разработка

1. С
2. Об
3. Об



Эффективная разработка

1. С
2. О
3. О

Время внесения дефекта	Время обнаружения дефекта: Выработка требований	Проектирование архитектуры	Конструирование (кодирование)	Тестирование	После выпуска ПО
Выработка требований	1	3	5-10	10	10-100
Проектирование архитектуры	-	1	10	15	25-100
Конструирование (кодирование)	-	-	1	10	10-25

Эффективная разработка

1. Статический анализатор находит в коде **возможные ошибки**.
2. Обнаруживает ошибки и **потенциальные уязвимости**.
3. Обнаруживает проблемы на **раннем этапе** разработки.
4. Может давать рекомендации по **оформлению** и **оптимизации кода**.

Эффективная разработка

1. Статический анализатор находит в коде **возможные ошибки**.

2. Обнаруживает ошибки и **потенциальные уязвимости**.

3. Обнаруживает проблемы на **раннем этапе** разработки.

4. Может давать рекомендации по **оформлению** и **оптимизации кода**.

5. Ещё **много чего ещё**, но время ограничено. Не будем отходить от темы доклада...

5 причин полюбить статический анализ

1. Внимательнее программиста

Как PVS-Studio оказался внимательнее, чем три с половиной программиста

PVS-Studio, как и другие статические анализаторы кода, часто выдаёт ложные срабатывания. Но не стоит спешить считать странные срабатывания ложными. Это короткая история о том, как PVS-Studio вновь оказался внимательнее нескольких человек.

Случай из поддержки

```
if (ch >= 0x0FF00)
{
    if (!((ch >= 0x0FF10) && (ch <= 0x0FF19)) ||
          ((ch >= 0x0FF21) && (ch <= 0x0FF3A)) ||
          ((ch >= 0x0FF41) && ((ch <= 0x0FF5A)))
    {
      if (j == 0)
        continue;
      ch = chx;
    }
}
```

Случай из поддержки

```
if (ch >= 0x0FF00)
{                                          //диапазоны значений:
  if (!((ch >= 0x0FF10) && (ch <= 0x0FF19)) ||  // 0..9
      ((ch >= 0x0FF21) && (ch <= 0x0FF3A)) ||   // A..Z
      ((ch >= 0x0FF41) && (ch <= 0x0FF5A))   )  // a..z
  {
    if (j == 0)
      continue;
```

V560 – A part of conditional expression is **always false**: (**ch >= 0x0FF21**);
V560 – A part of conditional expression is **always true**: (**ch <= 0x0FF3A**);
V560 – A part of conditional expression is **always false**: (**ch >= 0x0FF41**);
V560 – A part of conditional expression is **always true**: (**ch <= 0x0FF5A**);

Случай из поддержки

ch вне диапазона.

```
if (ch >= 0x0FF00)
{                                          //диапазоны значений:
   if (!((ch >= 0x0FF10) && (ch <= 0x0FF19)) ||  // 0..9
         ((ch >= 0x0FF21) && (ch <= 0x0FF3A)) ||
         ((ch >= 0x0FF41) && (ch <= 0x0FF5A))   )
   {
     if (j == 0)
       continue;
     ch = chx;
   }
}
```

Случай из поддержки

ch в диапазоне.

```
if (ch >= 0x0FF00)
{                                          //диапазоны значений:
   if (!((ch >= 0x0FF10) && (ch <= 0x0FF19)) ||  // 0..9
         ((ch >= 0x0FF21) && (ch <= 0x0FF3A)) ||  // A..Z
         ((ch >= 0x0FF41) && (ch <= 0x0FF5A))   ) // a..z
   {
     if (j == 0)
       continue;
     ch = chx;
   }
}
```

ch >= 0x0FF21 false
ch <= 0x0FF3A true
ch >= 0x0FF41 false
ch <= 0x0FF5A true

Как оказалось...

```
if (ch >= 0x0FF00)
{
    if ( !(((ch >= 0x0FF10) && (ch <= 0x0FF19)) ||
           ((ch >= 0x0FF21) && (ch <= 0x0FF3A)) ||
           ((ch >= 0x0FF41) && (ch <= 0x0FF5A))) )
    {
      if (j == 0)
        continue;
      ch = chx;
    }
}
```

```
typedef struct CurvesGeometry { .... };
```

Blender

Blender

```
typedef struct CurvesGeometry { .... };

namespace bke
{
  ....
  class CurvesGeometry : public ::CurvesGeometry { .... };
  ....
}
```

Blender

```
typedef struct CurvesGeometry { .... };

namespace bke
{
  ....
  class CurvesGeometry : public ::CurvesGeometry { .... };
  ....
  class CurvesFieldInput : public fn::FieldInput
  {
    ....
    virtual std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry &curves) const;
  };
  ....
}
```

```
namespace blender::nodes::node_geo_input_curve_handles_cc
{
  ....
  class HandlePositionFieldInput final :
                              public bke::CurvesFieldInput
  {
    ....
    std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry & /*curves*/) const;
  };
  ....
}
```

```
namespace blender::nodes::node_geo_input_curve_handles_cc
{
  ....
  class HandlePositionFieldInput final :
                              public bke::CurvesFieldInput
  {
    ....
    std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry & /*curves*/) const;
  };
```

V762 - It is possible a virtual function was overridden incorrectly. See first argument of function 'preferred_domain' in derived class 'HandlePositionFieldInput' and base class 'CurvesFieldInput'

```
namespace blender::nodes::node_geo_input_curve_handles_cc
{
  ....
  class HandlePositionFieldInput final :
                              public bke::CurvesFieldInput
  {
    ....
    std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry & /*curves*/) const override;
  };
  ....
}
```

```
namespace blender::nodes::node_geo_input_curve_handles_cc
{
  ....
  class HandlePositionFieldInput final :
                          public bke::CurvesFieldInput
  {
    ....
    std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry & /*curves*/) const override;
  };
  ....
}
```

Blender

```
typedef struct CurvesGeometry { .... };

namespace bke
{
  ....
  class CurvesGeometry : public ::CurvesGeometry { .... };
  ....
  class CurvesFieldInput : public fn::FieldInput
  {
    ....
    virtual std::optional<AttrDomain> preferred_domain(
      const CurvesGeometry &curves) const;
  };
  ....
}
```

```
namespace blender::nodes::node_geo_input_curve_handles_cc
{
  ....
  class HandlePositionFieldInput final :
                              public bke::CurvesFieldInput
  {
    ....
    std::optional<AttrDomain> preferred_domain(
      const bke::CurvesGeometry & /*curves*/) const
                                                override;
  };
}
```

Сообщение пользователя:

"PVS tool is reporting V575(The null pointer is passed into 'operator delete[]'. Inspect the argument. ConsoleApplication1.cpp 20) for the following code.

- Инструмент PVS выдает сообщение V575 (В 'operator delete[]' передан нулевой указатель. Проверьте аргумент. ConsoleApplication1.cpp 20) на использование следующего кода.

Expected outcome should be that V575 should not be reported for such scenarios. Could you please look into this?"

- Ожидаемый результат не должен быть таким, V575 не должна выдаваться для таких случаев. Не могли бы вы изучить этот вопрос?

Случай из поддержки

```
char* pchrTmpData = NULL;

void func1()
{
  pchrTmpData = new char[20];
}
```

```
int main()
{
   func1();

   if ( !pchrTmpData )
     delete[ ] pchrTmpData;

  return 0;
}
```

Случай из поддержки

```
char* pchrTmpData = NULL;

void func1()
{
  pchrTmpData = new char[20];
}
```

```
int main()
{
   func1();

  if ( !pchrTmpData )
     delete[ ] pchrTmpData;

  return 0;
}
```

2. Помогает учиться на ошибках

V547 Expression 'SIGNALS_HANDLER == nullptr' is always true. async_signals_handler.cpp

YTsaurus

```cpp
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

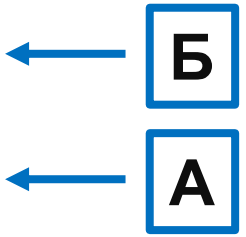
Б

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б



YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

Б

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

← **Б**

← **А**

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

A

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TGuard dnd(lock);

    if (SIGNALS_HANDLER == nullptr)
    {
      // NEVERS GETS DESTROYED
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Б

Потоки А и Б

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TG
    if
    {
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Б

Потоки А и Б

```
auto tmp = (TAsyncSignalsHandler *)malloc(sizeof(TAsyncSignalsHandler));
new (tmp) TAsyncSignalsHandler();
SIGNALS_HANDLER = tmp;
```

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TG
    if
    {
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Б

Потоки А и Б

```
auto tmp = (TAsyncSignalsHandler *)malloc(sizeof(TAsyncSignalsHandler));
new (tmp) TAsyncSignalsHandler();
SIGNALS_HANDLER = tmp;
```

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TG
    if
    {
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Б

Потоки А и Б

```
auto tmp = (TAsyncSignalsHandler *)malloc(sizeof(TAsyncSignalsHandler));

SIGNALS_HANDLER = tmp;

new (tmp) TAsyncSignalsHandler();
```

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TG
    if
    {
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

Потоки А и Б

Б

```
auto tmp = (TAsyncSignalsHandler *)malloc(sizeof(TAsyncSignalsHandler));

SIGNALS_HANDLER = tmp;

new (tmp) TAsyncSignalsHandler();
```

А

YTsaurus

```
TAsyncSignalsHandler *SIGNALS_HANDLER = nullptr;

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;

  if (Y_UNLIKELY(SIGNALS_HANDLER == nullptr))
  {
    TG
    if
    {
      SIGNALS_HANDLER = new TAsyncSignalsHandler();
    }
  }

  SIGNALS_HANDLER->Install(signum, handler);
}
```

auto tmp = (TAsyncSignalsHandler *)malloc(sizeof(TAsyncSignalsHandler));

SIGNALS_HANDLER = tmp;

new (tmp) TAsyncSignalsHandler(); ← А

Потоки А и Б

Б

```
std::atomic<TAsyncSignalsHandler*> SIGNALS_HANDLER { nullptr };

void SetAsyncSignalHandler(int signum, TAutoPtr<TEventHandler> handler)
{
  static TAdaptiveLock lock;
  auto tmp = SIGNALS_HANDLER.load(std::memory_order_acquire);

  if (Y_UNLIKELY(tmp == nullptr))
  {
    TGuard dnd(lock);

    tmp = SIGNALS_HANDLER.load(std::memory_order_relaxed);
    if (tmp == nullptr)
    {
      // NEVERS GETS DESTROYED
      tmp = new TAsyncSignalsHandler();
      SIGNALS_HANDLER.store(tmp, std::memory_order_release);
    }
  }
  tmp->Install(signum, handler);
}
```

YTsaurus

```
BezTriple *bezt2 = (BezTriple *)
                        MEM_malloc_arrayN(u,
                                          sizeof(BezTriple),
                                          "duplichar_bezt2" );

...
for (int i = nu2->pntsu; i > 0; i--)
{
  float *fp = bezt2->vec[0];
  fp[0] = (fp[0] + ofsx) * fsize;
  fp[1] = (fp[1] + ofsy) * fsize;
  fp[3] = (fp[3] + ofsx) * fsize;
  fp[4] = (fp[4] + ofsy) * fsize;
  fp[6] = (fp[6] + ofsx) * fsize;
  fp[7] = (fp[7] + ofsy) * fsize;
  bezt2++;
}
```

```
BezTriple *bezt2 = (BezTriple *)
                        MEM_malloc_arrayN(u,
                                          sizeof(BezTriple),
                                          "duplichar_bezt2" );

...
for (int i = nu2->pntsu; i > 0; i--)
{
  float *fp = bezt2->vec[0];
  fp[0] = (fp[0] + ofsx) * fsize;
  fp[1] = (fp[1] + ofsy) * fsize;
  fp[3] = (fp[3] + ofsx) * fsize;
  fp[4] = (fp[4] + ofsy) * fsize;
  fp[6] = (fp[6] + ofsx) * fsize;
  fp[7] = (fp[7] + ofsy) * fsize;
  bezt2++;
}
```

```
typedef struct BezTriple
{
  float vec [3] [3];
  ....
}
```

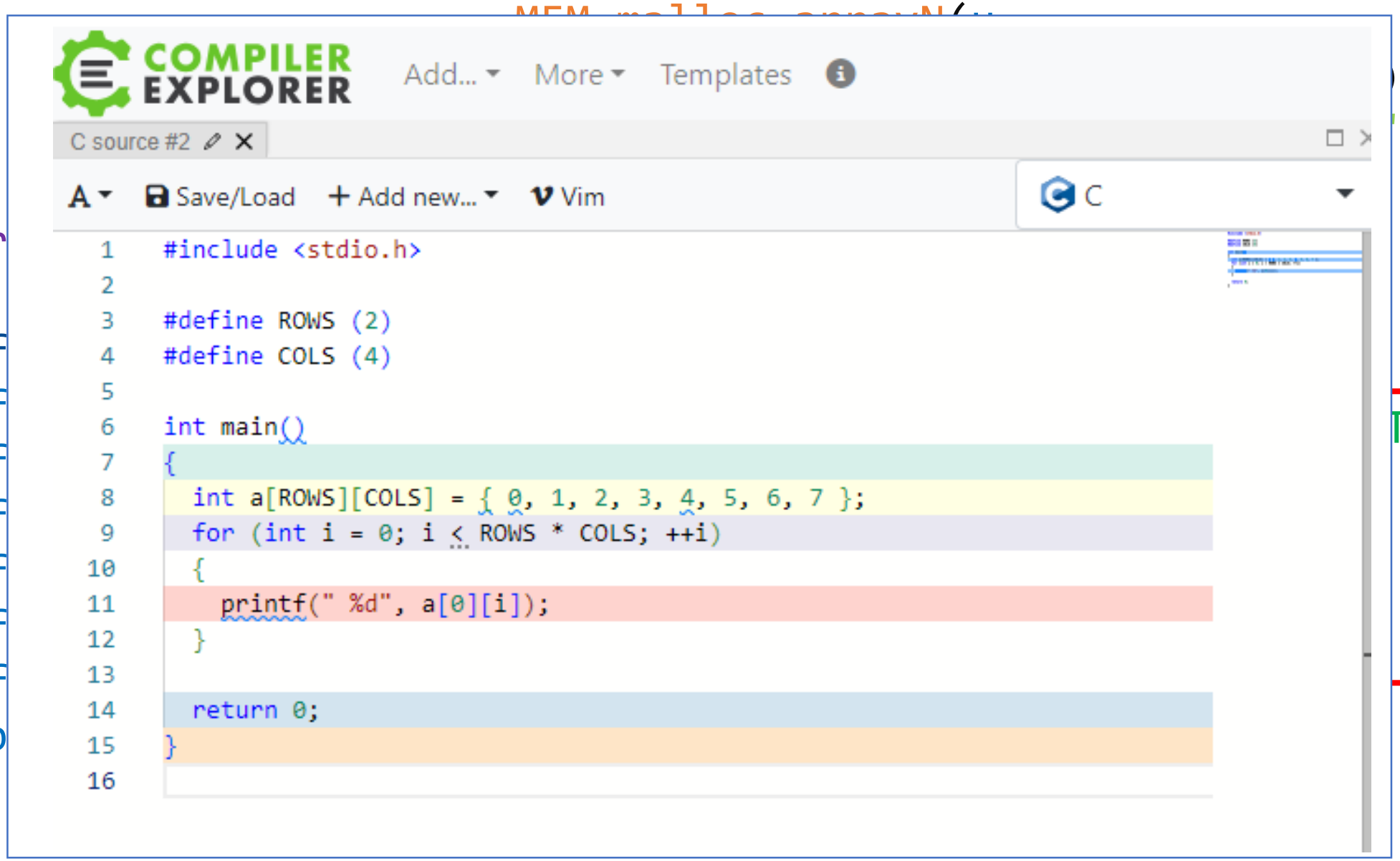
```
BezTriple *bezt2 = (BezTriple *)
                        MEM_malloc_arrayN(u,
                                          sizeof(BezTriple),
                                          "duplichar_bezt2" );

...
for (int i = nu2->pntsu; i > 0; i--)
{
  float *fp = bezt2->vec[0];
  fp[0] = (fp[0] + ofsx) * fsize;
  fp[1] = (fp[1] + ofsy) * fsize;
  fp[3] = (fp[3] + ofsx) * fsize;
  fp[4] = (fp[4] + ofsy) * fsize;
  fp[
  fp[
  bez
}
```

```
typedef struct BezTriple
{
  float vec [3] [3];
  ....
```

:А как же арифметика указателей и размещение ячеек массива последовательно в памяти Алексей?

```
BezTriple *bezt2 = (BezTriple *)
                   MEM_malloc_arrayN(...
...                                    ),
for                                     );
{
    f
    f
    f
    f
    f
    f
    f
    b
}
```

Triple



```
BezTriple *bezt2 = (BezTriple *)
..
fo
{
}
```

Output of x86-64 clang 17.0.1 (Compiler #1)

A ▾ | Wrap lines | Select all

```
      |                          ^~~~~~~~~~
      |                          {         }
<source>:8:37: warning: suggest braces around initialization of subobject [-Wmissing-braces]
    8 |   int a[ROWS][COLS] = { 0, 1, 2, 3, 4, 5, 6, 7 };
      |                                     ^~~~~~~~~~
      |                                     {         }
3 warnings generated.
ASM generation compiler returned: 0
<source>:6:9: warning: a function declaration without a prototype is deprecated in all versions of C [-Wstrict-prototypes]
    6 | int main()
      |         ^
      |          void
<source>:8:25: warning: suggest braces around initialization of subobject [-Wmissing-braces]
    8 |   int a[ROWS][COLS] = { 0, 1, 2, 3, 4, 5, 6, 7 };
      |                          ^~~~~~~~~~
      |                          {         }
<source>:8:37: warning: suggest braces around initialization of subobject [-Wmissing-braces]
    8 |   int a[ROWS][COLS] = { 0, 1, 2, 3, 4, 5, 6, 7 };
      |                                     ^~~~~~~~~~
      |                                     {         }
3 warnings generated.
Execution build compiler returned: 0
Program returned: 0
    0 1 2 3 4 5 6 7
```

Output of x86-64 clang 17.0.1 (Compiler #1)

```
Execution build compiler returned: 0
Program returned: 0
   0 1 2 3 4 5 6 7
```

Bez

```
...
for
{
    f
    f
    f
    f
    f
    f
    f
    b
}
```

Output of x86-64 gcc 13.2 (Compiler #2)

A ▾ | Wrap lines | Select all

```
<source>: In function 'main':
<source>:8:23: warning: missing braces around initializer [-Wmissing-braces]
    8 |   int a[ROWS][COLS] = { 0, 1, 2, 3, 4, 5, 6, 7 };
      |                       ^
      |                         {         } {         }
<source>:11:5: warning: iteration 4 invokes undefined behavior [-Waggressive-loop-optimizations]
   11 |     printf(" %d", a[0][i]);
      |     ^~~~~~~~~~~~~~~~~~~~~~
<source>:9:21: note: within this loop
    9 |   for (int i = 0; i < ROWS * COLS; ++i)
      |                   ^
ASM generation compiler returned: 0
<source>: In function 'main':
<source>:8:23: warning: missing braces around initializer [-Wmissing-braces]
    8 |   int a[ROWS][COLS] = { 0, 1, 2, 3, 4, 5, 6, 7 };
      |                       ^
      |                         {         } {         }
<source>:11:5: warning: iteration 4 invokes undefined behavior [-Waggressive-loop-optimizations]
   11 |     printf(" %d", a[0][i]);
      |     ^~~~~~~~~~~~~~~~~~~~~~
<source>:9:21: note: within this loop
    9 |   for (int i = 0; i < ROWS * COLS; ++i)
      |                   ^
Execution build compiler returned: 0
Program returned: 0
   0 1 2 3 1589009929 32766 100 0
```

Output of x86-64 gcc 13.2 (Compiler #2)

```
Execution build compiler returned: 0
Program returned: 0
   0 1 2 3 1589009929 32766 100 0
```

Вариант N1:

```
typedef struct BezTriple
{
  float vec [9];
  ....
}
```

Вариант N2:

```
for (....)
{
  for (....)
  {
    ....
  }
}
```

```
static void rigidbody_update_ob_array(RigidBodyWorld *rbw)
{
  if (rbw->group == nullptr)
  {
    rbw->numbodies = 0;
    rbw->objects = static_cast<Object **>
                   (realloc(rbw->objects, 0));
    return;
  }
  ....
}
```

Blender

```
static void rigidbody_update_ob_array(RigidBodyWorld *rbw)
{
  if (rbw->group == nullptr)
  {
    rbw->numbodies = 0;
    rbw->objects = static_cast<Object **>
                   (realloc(rbw->objects, 0));
    return;
  }
```

Blender

V575 The 'realloc' function processes '0' elements. Inspect the second argument

```
static void rigidbody_update_ob_array(RigidBodyWorld *rbw)
{
  if (rbw->group == nullptr)
  {
    rbw->numbodies = 0;
    rbw->objects = static_cast<Object **>
                   (realloc(rbw->objects, 0));
    return;
```

Blender

if `new_size` is zero, the behavior is implementation defined (null pointer may be returned (in which case the old memory block may or may not be freed), or some non-null pointer may be returned that may not be used to access storage). Such usage is deprecated (via C DR 400 🔒).(since C17) (until C23)
if `new_size` is zero, the behavior is undefined. (since C23)

3. Замечает нашу лень

Замечает нашу лень!

Blender

```
static int gizmo_cage2d_modal(....)
{
  ....
  if ((transform_flag & ED_GIZMO_CAGE_XFORM_FLAG_SCALE_UNIFORM) == 0)
  {
    const bool use_temp_uniform = (event->modifier & KM_SHIFT) != 0;
    const bool changed = data->use_temp_uniform != use_temp_uniform;
    data->use_temp_uniform = data->use_temp_uniform;
    ....
  }
  ....
}
```

Замечает нашу лень!

Blender

```
static int gizmo_cage2d_modal(....)
{
  ....
  if ((transform_flag & ED_GIZMO_CAGE_XFORM_FLAG_SCALE_UNIFORM) == 0)
  {
    const bool use_temp_uniform = (event->modifier & KM_SHIFT) != 0;
    const bool changed = data->use_temp_uniform != use_temp_uniform;
    data->use_temp_uniform = data->use_temp_uniform;
    ....
  }
  ....
}
```

Замечает нашу лень!

```
void BKE_gpencil_stroke_copy_settings(const bGPDstroke *gps_src,
                                      bGPDstroke *gps_dst)
{
  ....
  gps_dst->mat_nr = gps_src->mat_nr;
  copy_v2_v2_short(gps_dst->caps, gps_src->caps);
  gps_dst->hardness = gps_src->hardness;
  copy_v2_v2(gps_dst->aspect_ratio, gps_src->aspect_ratio);
  gps_dst->fill_opacity_fac = gps_dst->fill_opacity_fac;
  copy_v3_v3(gps_dst->boundbox_min, gps_src->boundbox_min);
  copy_v3_v3(gps_dst->boundbox_max, gps_src->boundbox_max);
  ....
}
```

Blender

Замечает нашу лень!

```
void BKE_gpencil_stroke_copy_settings(const bGPDstroke *gps_src,
                                      bGPDstroke *gps_dst)
{
  ....
  gps_dst->mat_nr = gps_src->mat_nr;
  copy_v2_v2_short(gps_dst->caps, gps_src->caps);
  gps_dst->hardness = gps_src->hardness;
  copy_v2_v2(gps_dst->aspect_ratio, gps_src->aspect_ratio);
  gps_dst->fill_opacity_fac = gps_dst->fill_opacity_fac;
  copy_v3_v3(gps_dst->boundbox_min, gps_src->boundbox_min);
  copy_v3_v3(gps_dst->boundbox_max, gps_src->boundbox_max);
  ....
}
```

Blender

Замечает нашу лень!

```
void BKE_gpencil_stroke_copy_settings(const bGPDstroke *gps_src,
                                      bGPDstroke *gps_dst)
{
  gps_dst->thickness = gps_src->thickness;
  gps_dst->flag = gps_src->flag;
  gps_dst->inittime = gps_src->inittime;
  gps_dst->mat_nr = gps_src->mat_nr;
  copy_v2_v2_short(gps_dst->caps, gps_src->caps);
  gps_dst->hardness = gps_src->hardness;
  copy_v2_v2(gps_dst->aspect_ratio, gps_src->aspect_ratio);
  gps_dst->fill_opacity_fac = gps_dst->fill_opacity_fac;
  copy_v3_v3(gps_dst->boundbox_min, gps_src->boundbox_min);
  copy_v3_v3(gps_dst->boundbox_max, gps_src->boundbox_max);
  gps_dst->uv_rotation = gps_src->uv_rotation;
  copy_v2_v2(gps_dst->uv_translation, gps_src->uv_translation);
  gps_dst->uv_scale = gps_src->uv_scale;
  gps_dst->select_index = gps_src->select_index;
  copy_v4_v4(gps_dst->vert_color_fill, gps_src->vert_color_fill);
}
```

Blender

Замечает нашу лень!

```
void BKE_gpencil_stroke_copy_settings(const bGPDstroke *gps_src,
                                      bGPDstroke *gps_dst)
{
  gps_dst->thickness = gps_src->thickness;
  gps_dst->flag = gps_src->flag;
  gps_dst->inittime = gps_src->inittime;
  gps_dst->mat_nr = gps_src->mat_nr;
  copy_v2_v2_short(gps_dst->caps, gps_src->caps);
  gps_dst->hardness = gps_src->hardness;
  copy_v2_v2(gps_dst->aspect_ratio, gps_src->aspect_ratio);
  gps_dst->fill_opacity_fac = gps_dst->fill_opacity_fac;
  copy_v3_v3(gps_dst->boundbox_min, gps_src->boundbox_min);
  copy_v3_v3(gps_dst->boundbox_max, gps_src->boundbox_max);
  gps_dst->uv_rotation = gps_src->uv_rotation;
  copy_v2_v2(gps_dst->uv_translation, gps_src->uv_translation);
  gps_dst->uv_scale = gps_src->uv_scale;
  gps_dst->select_index = gps_src->select_index;
  copy_v4_v4(gps_dst->vert_color_fill, gps_src->vert_color_fill);
}
```

Blender

Замечает нашу лень!

```
void Reconfigure(TDistributedThrottlerConfigPtr config) override
{
  DiscoveryClient_->Reconfigure(config->DiscoveryClient);
  ....
  if (oldConfig->LimitUpdatePeriod != config->LimitUpdatePeriod)
  {
    UpdateLimitsExecutor_->SetPeriod(config->LimitUpdatePeriod);
  }
  if (oldConfig->LeaderUpdatePeriod != config->LeaderUpdatePeriod)
  {
    UpdateLeaderExecutor_->SetPeriod(config->LimitUpdatePeriod);
  }
  ....
  Config_.Store(std::move(config));
}
```

Blender

Замечает нашу лень!

Blender

```
void Reconfigure(TDistributedThrottlerConfigPtr config) override
{
  DiscoveryClient_->Reconfigure(config->DiscoveryClient);
  ....
  if (oldConfig->LimitUpdatePeriod != config->LimitUpdatePeriod)
  {
    UpdateLimitsExecutor_->SetPeriod(config->LimitUpdatePeriod);
  }
  if (oldConfig->LeaderUpdatePeriod != config->LeaderUpdatePeriod)
  {
    UpdateLeaderExecutor_->SetPeriod(config->LimitUpdatePeriod);
  }
  ....
  Config_.Store(std::move(config));
}
```

4. Помогает оптимизировать код

Оптимизируем?

Blender

```
static void
    add_uv_primitive_shared_uv_edge(const MeshData &mesh_data,
                                    UVIsland &island,
                                    UVVertex *connected_vert_1,
                                    UVVertex *connected_vert_2,
                                    float2 uv_unconnected,
                                    const int mesh_primitive_i )
{
  ....
}
```

А зачем?

Blender

```
static void
    add_uv_primitive_shared_uv_edge(const MeshData &mesh_data,
                                    UVIsland &island,
                                    UVVertex *connected_vert_1,
                                    UVVertex *connected_vert_2,
                                    float2 uv_unconnected,
                                                          e_i )
{
  ....
}
```

V813 Decreased performance. The 'uv_unconnected' argument should probably be rendered as a constant reference. pbvh_uv_islands.cc 800

А как так то?

Blender

```
static void
    add_uv_primitive_shared_uv_edge(const MeshData &mesh_data,
                                    UVIsland &island,
                                    UVVertex *connected_vert_1,
                                    UVVertex *connected_vert_2,
                                    float2 uv_unconnected,
                                    const int mesh_primitive_i )
{
 ....
}
```

```
typedef struct
float2
{
   ....
};
```

Сложно...

YTsaurus

```
int TComparator::CompareKeyBounds(const TKeyBound& lhs,
                                  const TKeyBound& rhs,
                                  int lowerVsUpper) const
{
  ....
  {
    auto lhsInclusivenessAsUpper = (lhs.IsUpper && lhs.IsInclusive) ||
                                   (!lhs.IsUpper && !lhs.IsInclusive);
    auto rhsInclusivenessAsUpper = (rhs.IsUpper && rhs.IsInclusive) ||
                                   (!rhs.IsUpper && !rhs.IsInclusive);
    if (lhsInclusivenessAsUpper != rhsInclusivenessAsUpper)
    {
      return lhsInclusivenessAsUpper - rhsInclusivenessAsUpper;
    }
  }
  ....
}
```

Но, есть статический анализ

YTsaurus

```
int TComparator::CompareKeyBounds(const TKeyBound& lhs,
                                  const TKeyBound& rhs,
                                  int lowerVsUpper) const
{
  ....
  {
    auto lhsInclusivenessAsUpper = (lhs.IsUpper && lhs.IsInclusive) ||
                                   (!lhs.IsUpper && !lhs.IsInclusive);
    auto rhsInclusivenessAsUpper = (rhs.IsUpper && rhs.IsInclusive) ||
                                   (!rhs.IsUpper && !rhs.IsInclusive);
    if (lhsInclusivenessAsUpper != rhsInclusivenessAsUpper)
    {

    }
  }
  ....
}
```

V728 – Message: An excessive check can be simplified. The '(A && B) || (!A && !B)' expression is equivalent to the **'bool(A) == bool(B)'** expression.

Стало проще и понятнее

```
int TComparator::CompareKeyBounds(const TKeyBound& lhs,
                                  const TKeyBound& rhs,
                                  int lowerVsUpper) const
{
  ....
  {
    auto lhsInclusivenessAsUpper = lhs.IsUpper == lhs.IsInclusive;
    auto rhsInclusivenessAsUpper = rhs.IsUpper == rhs.IsInclusive;

    if (lhsInclusivenessAsUpper != rhsInclusivenessAsUpper)
    {
      return lhsInclusivenessAsUpper - rhsInclusivenessAsUpper;
    }
  }
  ....
}
```

YTsaurus

Непонятные условия, что могло пойти не так?

```
void BLI_threadpool_init(ListBase *threadbase, void *(*do_thread)(void *), int tot)
{
....
  if (threadbase != nullptr && tot > 0)
  {
    ....
    if (tot > RE_MAX_THREAD) { tot = RE_MAX_THREAD; }
    else
    if (tot < 1) { tot = 1; }
    ....
  }
....
}
```

Blender

Непонятные условия, что могло пойти не так?

```
void BLI_threadpool_init(ListBase *threadbase, void *(*do_thread)(void *), int tot)
{
....
  if (threadbase != nullptr && tot > 0)
  {
    ....
    if (tot > RE_MAX_THREAD) { tot = RE_MAX_THREAD; }
    else
    if (tot < 1) { tot = 1; }
    ....
  }
....
}
```

V547 Expression 'tot < 1' is always false.

Blender

Такое бывает и часто...

```
void BLI_threadpool_init(ListBase *threadbase, void *(*do_thread)(void *), int tot)
{
....
  if (threadbase != nullptr && tot > 0)
  {
    ....
    if (tot > RE_MAX_THREAD) { tot = RE_MAX_THREAD; }
    else
    if (tot < 1) { tot = 1; }
    ....
  }
....
}
```

Blender

Ничего лишнего!

```
void BLI_threadpool_init(ListBase *threadbase, void *(*do_thread)(void *), int tot)
{
....
  if (threadbase != nullptr && tot > 0)
  {
    ....
    if (tot > RE_MAX_THREAD) { tot = RE_MAX_THREAD; }
    ....
  }
....
}
```

Blender

5. Экономит наше время

olekl 24 окт 2013 в 13:09

Пример использования статического анализатора

3 мин 19К

Программирование*, Совершенный код*, C*

И ещё раз, чтобы не забыть

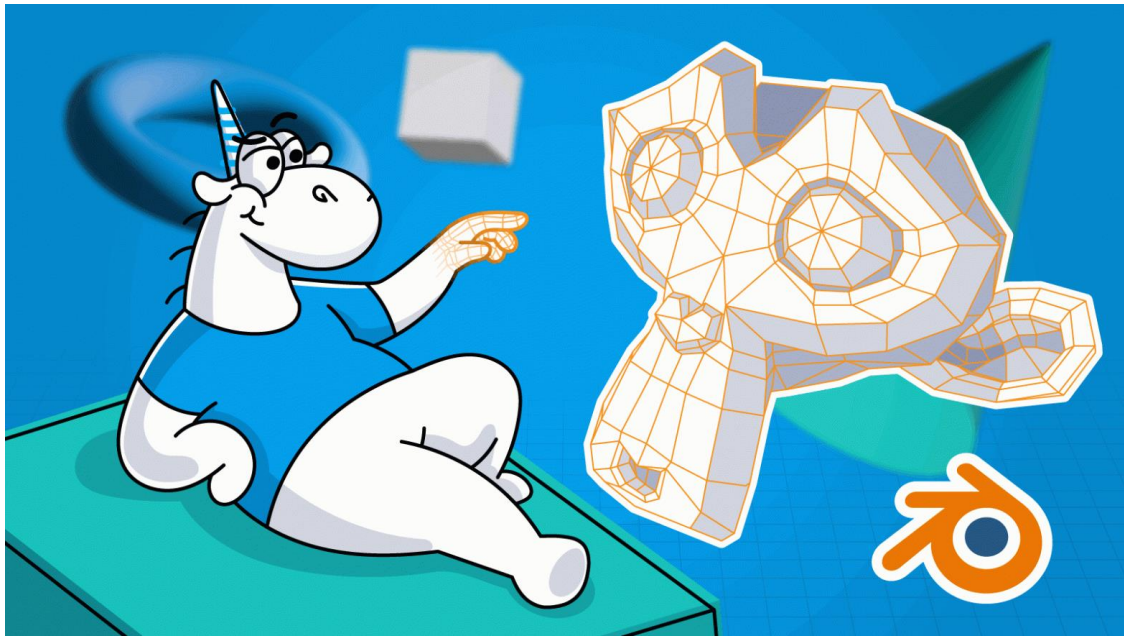
1. Внимательнее программиста
2. Помогает нам учиться на чужом негативном опыте и ошибках
3. Замечает нашу лень или проблема COPY_PASTE
4. Помогает оптимизировать код и сделать его чистым
5. Экономит наше время

Алексей Горшков

04 Мар 2024

Теги:
#Cpp

Проверяем Blender

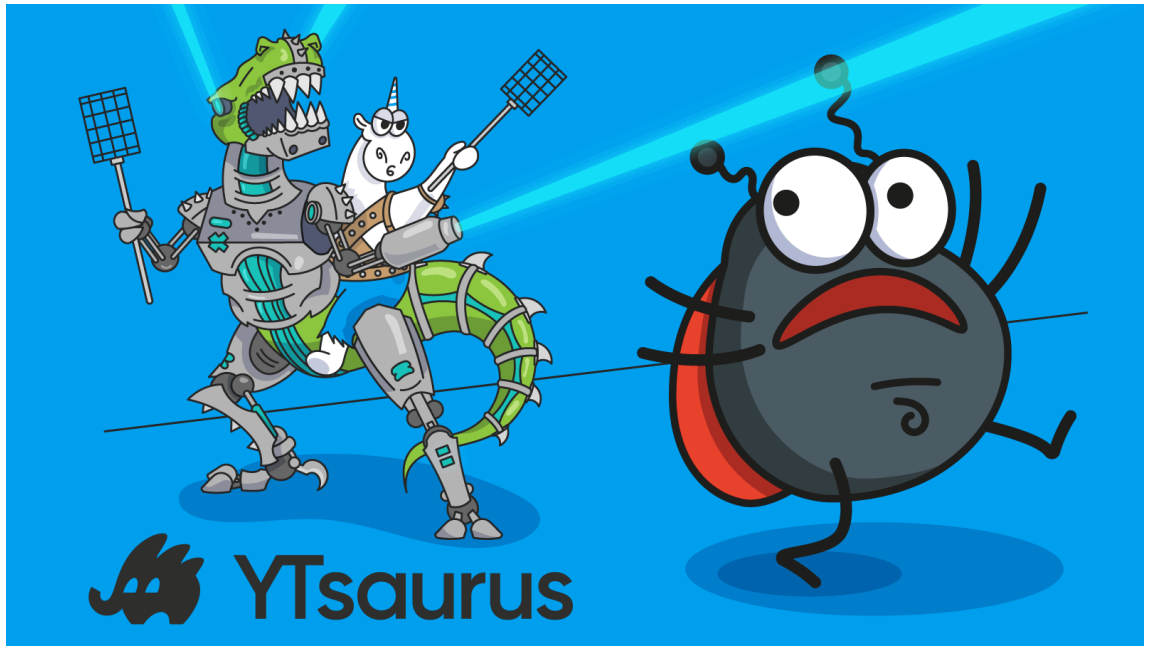


Алексей Горшков

31 Окт 2023

Теги:
#Cpp

Проверяем YTsaurus. Доступность, надёжность, open source



Q&A

pvs-studio.com
gorshkov@viva64.com