Highlighting C++ with C++ is... hard

Marcin Zdun

m.zdun@samsung.com

```
switch (c) {
case '[': return set_next(offset, SQBRAKET_O);
case ']': return set_next(offset, SQBRAKET_C);
case '{': return set_next(offset, CBRAKET_O);
case '}': return set_next(offset, CBRAKET_C);
case '(': return set_next(offset, BRAKET_O);
case ')': return set_next(offset, BRAKET_C);
case '=': return set_next(offset, EQ_SIGN);
case ';': return set_next(offset, SEMI);
case ',': return set_next(offset, COMMA);
case '"': {
        bool escaping = false;
```

```
switch (c) {
case '[': return set_next(offset, SQBRAKET_O);
case ']': return set_next(offset, SQBRAKET_C);
case '{': return set_next(offset, CBRAKET_O);
case '}': return set_next(offset, CBRAKET_C);
case '(': return set_next(offset, BRAKET_O);
case ')': return set_next(offset, BRAKET_C);
case '=': return set_next(offset, EQ_SIGN);
case ';': return set_next(offset, SEMI);
case ',': return set_next(offset, COMMA);
case '"': {
        bool escaping = false;
```

Random rule

header-name:

< h-char-sequence >

" q-char-sequence "

Boost.Spirit X3

header-name:
  < h-char-sequence >
  " q-char-sequence "

```
auto header_name =
    ('<' >> h_char_sequence >> '>')
  | ('"' >> q_char_sequence >> '"')
  ;
```


Not
invented
here

constexpr all the things!

Spirit-inspired homegrown template programming

```
constexpr auto header_name =
  ('<' >> h_char_sequence >> '>') [on_system_header]
  | ('"' >> q_char_sequence >> '"') [on_local_header]
  ;
```

Spirit-inspired homegrown template programming

```
constexpr auto header_name =
  ('<' >> h_char_sequence >> '>') [on_system_header]
  | ('"' >> q_char_sequence >> '"') [on_local_header]
  ;
```

- 456 characters (MS ABI)
- 218 characters (Itanium ABI)

My little *control-line* fiasco

cxx.cc

hilite-cxx | hl::cxx::parser

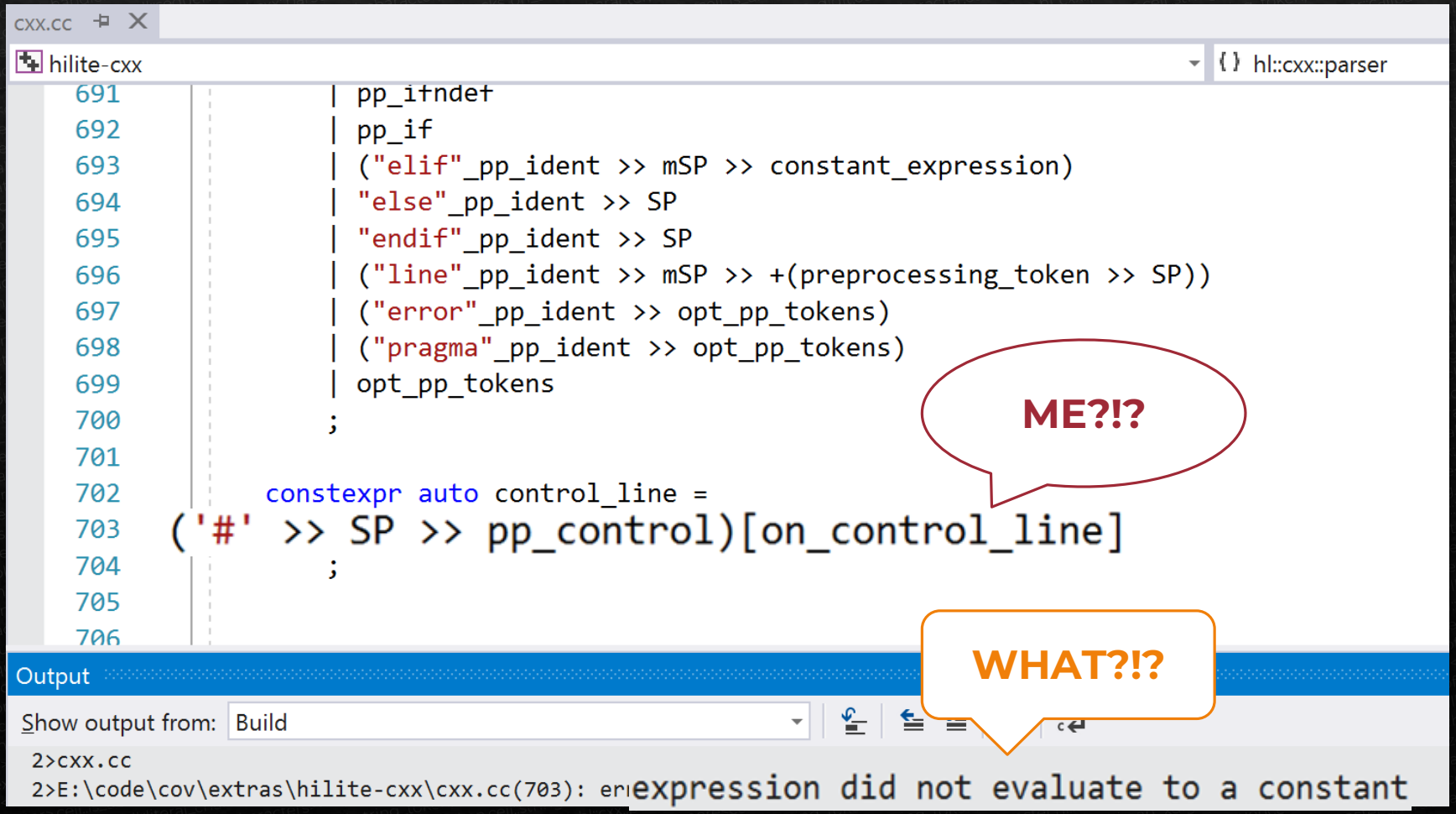
```
691         | pp_ifndef
692         | pp_if
693         | ("elif"_pp_ident >> mSP >> constant_expression)
694         | "else"_pp_ident >> SP
695         | "endif"_pp_ident >> SP
696         | ("line"_pp_ident >> mSP >> +(preprocessing_token >> SP))
697         | ("error"_pp_ident >> opt_pp_tokens)
698         | ("pragma"_pp_ident >> opt_pp_tokens)
699         | opt_pp_tokens
700         ;
701
702     constexpr auto control_line =
703         ('#' >> SP >> pp_control)[on_control_line]
704         ;
705
706
```

Output

Show output from: Build

```
2>cxx.cc
2>E:\code\cov\extras\hilite-cxx\cxx.cc(703): error C2131: expression did not evaluate to a constant
```

My little *control-line* fiasco




DEAD END

Workarounds:
A class with shorter name

```
class shorter_name: public parser<shorter_name> {
    static constexpr auto grammar =
        pp_include
        | pp_define
        // ...
};

constexpr auto pp_control = shorter_name{};
```

Workarounds: wrapped<*Lambda*> wrap(...)

```
constexpr auto pp_control = wrap([]() -> decltype(auto) {
  static constexpr auto grammar =
      pp_include
    | pp_define
    // ...
    ;
  return (grammar);
});
```

Workarounds: compilers update

```
constexpr auto pp_control =
    pp_include
    | pp_define
    // ...
    ;
```

Original code

- GCC 8: 5 sec.
- VS 2019: 15 sec.
- VS 2017: **61 sec.** 💥

shorter_typename

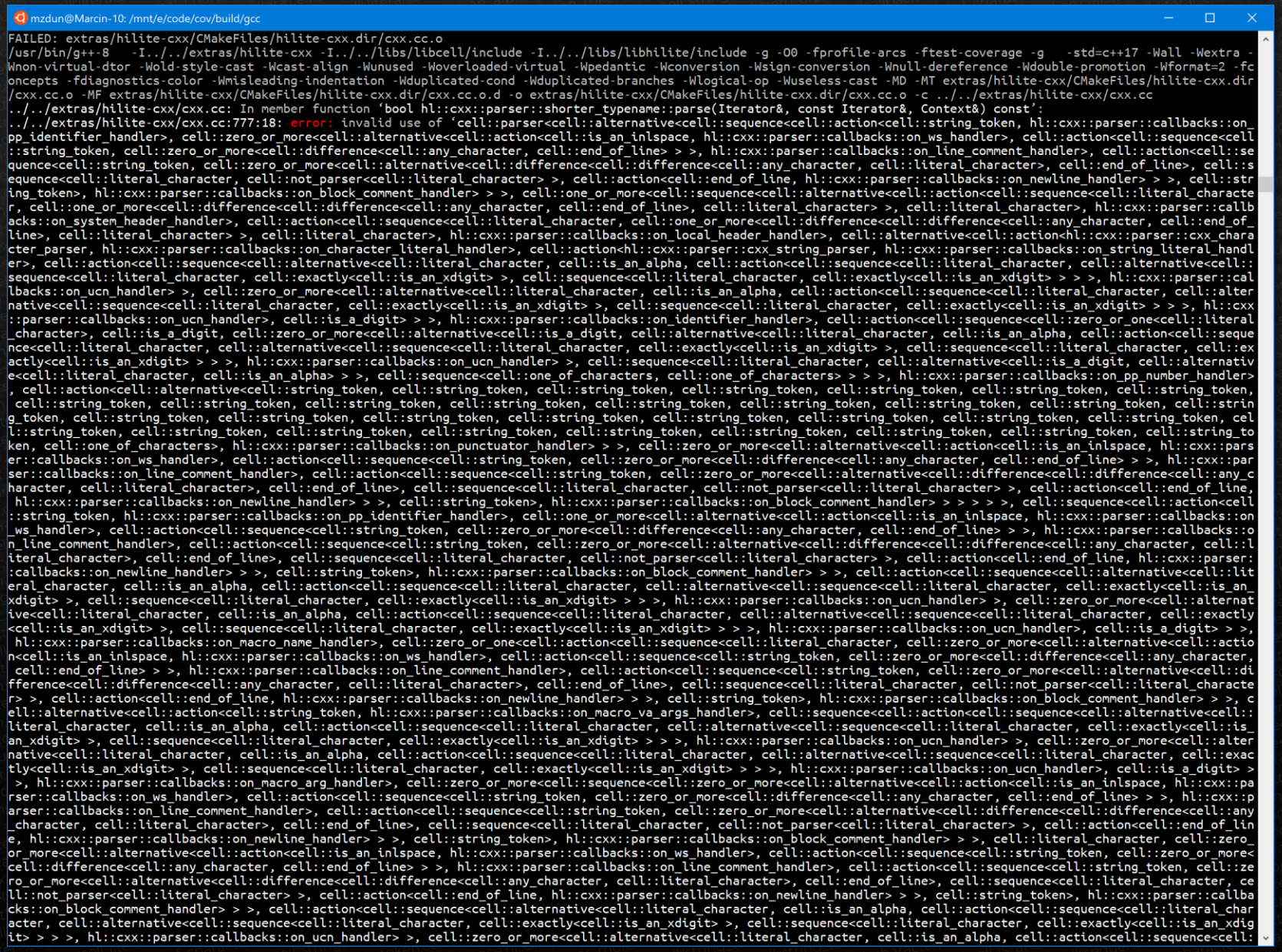
- GCC 8: 5 sec.
- VS 2019: 12 sec.
- VS 2017: **31 sec.** ⬆

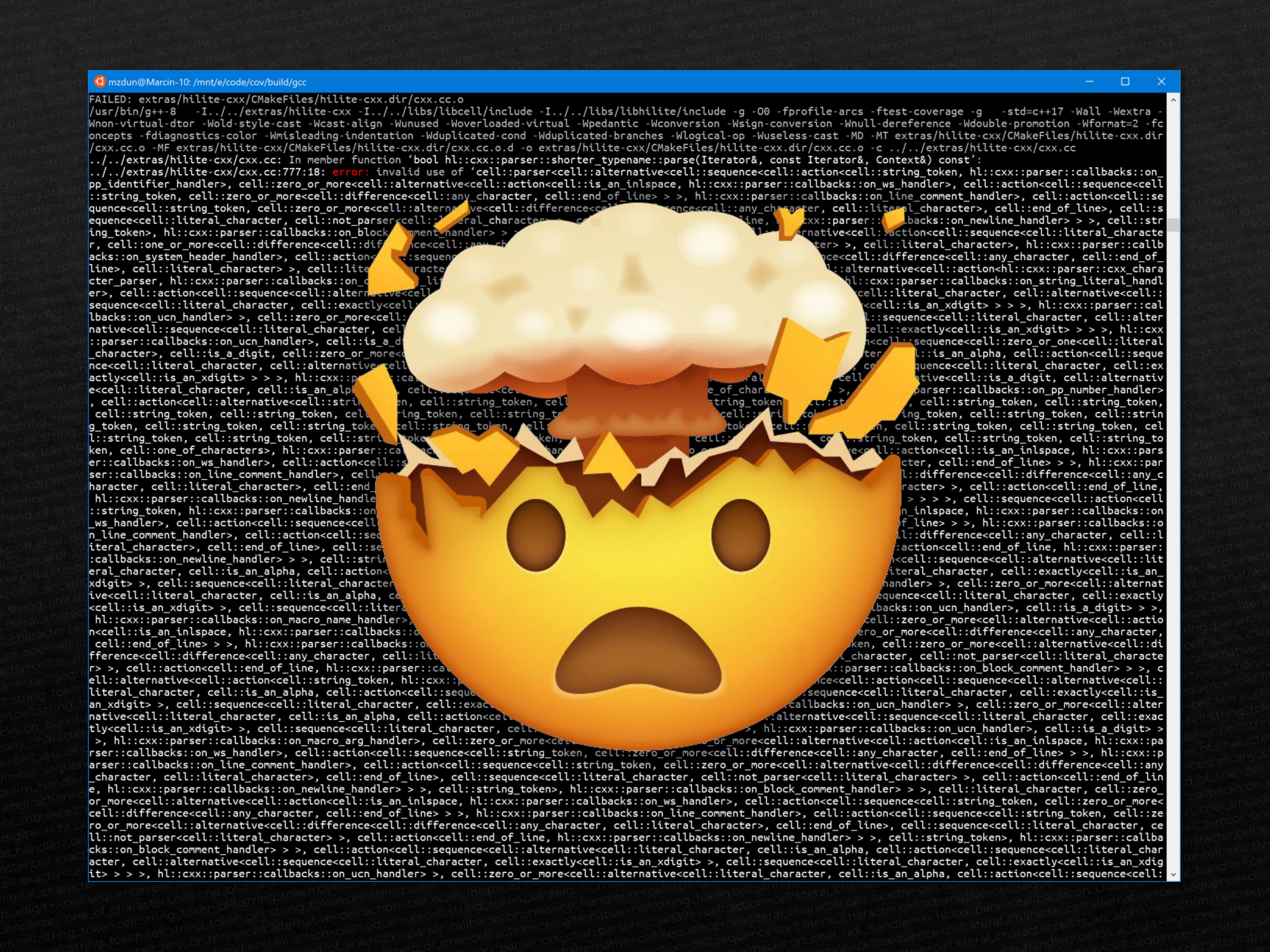
wrapped<*Lambda*>

- GCC 8: 5 sec.
- VS 2019: 15 sec.
- VS 2017: **22 sec.** 

Symbol name

- Top-level grammar
- Itanium ABI: 2’345
- c++filt: **297’589**

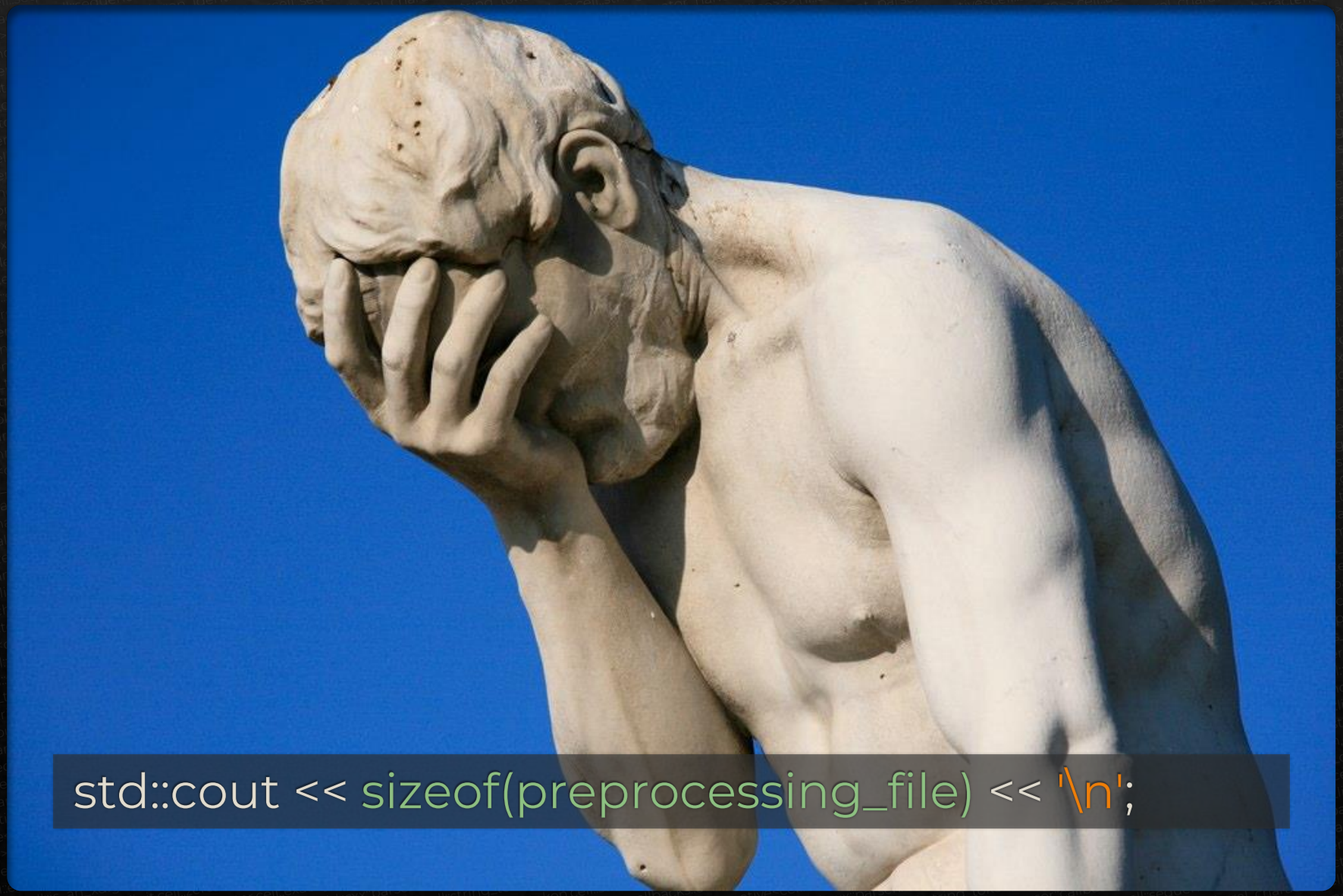

mzdun@Marcin-10: /mnt/e/code/cov/build/gcc
FAILED: extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o
/usr/bin/g++-8 -I../../extras/hilite-cxx -I../../libs/libcell/include -I../../libs/libhilite/include -g -O0 -fprofile-arcs -ftest-coverage -g -std=c++17 -Wall -Wextra -Wnon-virtual-dtor -Wold-style-cast -Wcast-align -Wunused -Woverloaded-virtual -Wpedantic -Wconversion -Wsign-conversion -Wnull-dereference -Wdouble-promotion -Wformat=2 -fconcepts -fdiagnostics-color -Wmisleading-indentation -Wduplicated-cond -Wduplicated-branches -Wlogical-op -Wuseless-cast -MD -MT extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o -MF extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o.d -o extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o -c ../../extras/hilite-cxx/cxx.cc
../../extras/hilite-cxx/cxx.cc: In member function 'bool hl::cxx::parser::shorter_typename::parse(Iterator&, const Iterator&, Context&) const':
../../extras/hilite-cxx/cxx.cc:777:18: error: invalid use of 'cell::parser<cell::alternative<cell::sequence<cell::action<cell::string_token, hl::cxx::parser::callbacks::on_pp_identifier_handler>, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > >, cell::one_or_more<cell::sequence<cell::alternative<cell::action<cell::sequence<cell::literal_character, cell::one_or_more<cell::difference<cell::difference<cell::any_character, cell::end_of_line>, cell::literal_character> >, cell::literal_character>, hl::cxx::parser::callbacks::on_system_header_handler>, cell::action<cell::sequence<cell::literal_character, cell::one_or_more<cell::difference<cell::difference<cell::any_character, cell::end_of_line>, cell::literal_character> >, cell::literal_character>, hl::cxx::parser::callbacks::on_local_header_handler>, cell::alternative<cell::action<hl::cxx::parser::cxx_character_parser, hl::cxx::parser::callbacks::on_character_literal_handler>, cell::action<hl::cxx::parser::cxx_string_parser, hl::cxx::parser::callbacks::on_string_literal_handler>, cell::action<cell::sequence<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::zero_or_more<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler>, cell::is_a_digit> > >, hl::cxx::parser::callbacks::on_identifier_handler>, cell::action<cell::sequence<cell::zero_or_one<cell::literal_character>, cell::is_a_digit, cell::zero_or_more<cell::alternative<cell::is_a_digit, cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::sequence<cell::literal_character, cell::alternative<cell::is_a_digit, cell::alternative<cell::literal_character, cell::is_an_alpha> > >, cell::sequence<cell::one_of_characters, cell::one_of_characters> > > >, hl::cxx::parser::callbacks::on_pp_number_handler>, cell::action<cell::alternative<cell::string_token, cell::one_of_characters>, hl::cxx::parser::callbacks::on_punctuator_handler> > >, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > > > >, cell::sequence<cell::action<cell::string_token, hl::cxx::parser::callbacks::on_pp_identifier_handler>, cell::one_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > >, cell::action<cell::sequence<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::zero_or_more<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler>, cell::is_a_digit> > >, hl::cxx::parser::callbacks::on_macro_name_handler>, cell::zero_or_one<cell::action<cell::sequence<cell::literal_character, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > >, cell::alternative<cell::action<cell::string_token, hl::cxx::parser::callbacks::on_macro_va_args_handler>, cell::sequence<cell::action<cell::sequence<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::zero_or_more<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler>, cell::is_a_digit> > >, hl::cxx::parser::callbacks::on_macro_arg_handler>, cell::zero_or_more<cell::sequence<cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > >, cell::literal_character, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> > >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> > >, cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::zero_or_more<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::


mzdun@Marcin-10: /mnt/e/code/cov/build/gcc
FAILED: extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o
/usr/bin/g++-8 -I../../extras/hilite-cxx -I../../libs/libcell/include -I../../libs/libhilite/include -g -O0 -fprofile-arcs -ftest-coverage -g -std=c++17 -Wall -Wextra -Wnon-virtual-dtor -Wold-style-cast -Wcast-align -Wunused -Woverloaded-virtual -Wpedantic -Wconversion -Wsign-conversion -Wnull-dereference -Wdouble-promotion -Wformat=2 -fconcepts -fdiagnostics-color -Wmisleading-indentation -Wduplicated-cond -Wduplicated-branches -Wlogical-op -Wuseless-cast -MD -MT extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o -MF extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o.d -o extras/hilite-cxx/CMakeFiles/hilite-cxx.dir/cxx.cc.o -c ../../extras/hilite-cxx/cxx.cc
../../extras/hilite-cxx/cxx.cc: In member function ‘bool hl::cxx::parser::shorter_typename::parse(Iterator&, const Iterator&, Context&) const’:
../../extras/hilite-cxx/cxx.cc:777:18: error: invalid use of ‘cell::parser<cell::alternative<cell::sequence<cell::action<cell::string_token, hl::cxx::parser::callbacks::on_pp_identifier_handler>, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> >, cell::literal_character, cell::zero_or_more<cell::alternative<cell::action<cell::is_an_inlspace, hl::cxx::parser::callbacks::on_ws_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::difference<cell::any_character, cell::end_of_line> > >, hl::cxx::parser::callbacks::on_line_comment_handler>, cell::action<cell::sequence<cell::string_token, cell::zero_or_more<cell::alternative<cell::difference<cell::difference<cell::any_character, cell::literal_character>, cell::end_of_line>, cell::sequence<cell::literal_character, cell::not_parser<cell::literal_character> >, cell::action<cell::end_of_line, hl::cxx::parser::callbacks::on_newline_handler> >, cell::string_token>, hl::cxx::parser::callbacks::on_block_comment_handler> >, cell::action<cell::sequence<cell::alternative<cell::literal_character, cell::is_an_alpha>, cell::action<cell::sequence<cell::literal_character, cell::alternative<cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> >, cell::sequence<cell::literal_character, cell::exactly<cell::is_an_xdigit> > > >, hl::cxx::parser::callbacks::on_ucn_handler> >, cell::zero_or_more<cell::alternative<cell::literal_character, cell::is_an_alpha, cell::action<cell::sequence<cell::

ICE-bergs are everywhere!


DETOUR

Do not fixate!


std::cout << sizeof(preprocessing_file) << '\n';

Thank You