

# SuperTest

## More than compiler validation

*from ACE Associated Compiler Experts*

### Compiler quality and conformance

The complexity of software systems is increasing rapidly, typically doubling the number of source lines by each year. On the one hand reuse of existing code, third party libraries and automated code generation ease the software development process, but on the other hand there is great diversity of source code which needs to be compiled into target code. Clearly more important than performance, compiler quality is imperative. Both compiler vendor and end-user need to be certain that their compilers are up to the task, for a single compiler bug can cause immense problems for a multitude of programmers, software projects or even devices in the market.

The damage a compiler may cause by producing wrong code is by far more expensive compared to an investment in a reliable compiler assurance system.

### SuperTest™

SuperTest is the world's most comprehensive compiler test and validation suite, coming from ACE Associated Compiler Experts in Amsterdam. With SuperTest, ACE makes its well over 30 years of experience and expertise in compiler construction and testing available to the compiler industry. Both professional compiler developers and software quality assurance engineers will appreciate the over 40,000 source files in SuperTest, providing over a million conformance tests, as well as a host of quality and regression tests dealing with compiler internals such as analyses and optimizations. As compilation techniques evolve, this set of tests and its special features are constantly growing. The procedures include both positive and negative tests and allow for selective testing through test subsets of previously failed tests.

The straightforward user interface seamlessly integrates into any compiler development environment, allows for easy addition of new tests and provides cross platform validation support.

The language conformance checks included in SuperTest check the compiler against:

- ISO 9899:1990 C standard
- ISO 9899:1999 C99 standard
- TR18037 Embedded C extension
- ISO 14882 C++ standard

The SuperTest depth test suite verifies the compiler's ability to correctly handle combinations of function calls, operators, storage classes, data types and pointer indirection levels. The built-in regression support eases tracking of progress towards validation and uncovers quality lapses. This collection of expert compiler validation tools makes SuperTest the most versatile compiler test suite for the professional compiler developer. In order to reduce compiler verification efforts and delays in the release schedule, compiler testing should best start right from the first day of development (well before external release) and should continue throughout the lifetime of the compiler. SuperTest is suitable for the end-user who needs to understand or validate the quality of the compiler(s) to be used for a critical software project or application.



Determining the quality and robustness of C/C++ compilers is a complicated and challenging task. This is because compilers are complex by nature. They transform the source code in many ways to produce highly optimized target code and must do so without the slightest change of the application functionality. Furthermore, C and C++ are used in a wide range of applications, from small embedded systems to large distributed database applications, regularly leading to most ingenious code constructs.

Additionally, many application generator tools produce code that is very often unlike that which a programmer would write.

```
#define COUNT      6
int clear[COUNT] = {
    2, 3, 5, 7, 11, 13
};

void test_bitclear(void)
{
    int    i;
    int    bits;
    int    c;

    bits = 0xffff;

    for (i = 0; i < COUNT; i++) {
        c = clear[i];
        bits &= ~(1 << c);
    }

    CVAL_VERIFY(bits == 0xd753);
}
```

## Industry feedback

- "... Since we started using SuperTest, not a single customer has reported finding a code generation problem with the C compiler."
- "... By selecting just the ANSI C parts of the test suite we were able to rapidly check that our initial compiler was working as we expected. Problems highlighted by SuperTest have usually allowed us to home in on the cause of the problem quite quickly. SuperTest provides confidence in the scope and stability of our compiler. We view it as the prime step into testing our compiler. Regular use of SuperTest identifies regression problems in a timely manner. We use SuperTest at least weekly and more frequently when making significant changes to our compiler."
- "... Through its systematic testing, compiler stressing and corner cases, the SuperTest tests revealed possibly hundreds of issues."
- "... The test cases are well documented, which tell exactly where the problem is."
- "... SuperTest is a really valuable tool for us. The verbose mode is wonderful for pointing me at the failing section of code. SuperTest found us implementation holes, implementation bugs, documentation errors, simulation bugs and many regressions. It is my primary regression test bench. I'd start using it very early in future compiler projects."
- "... The suite detected a lot of errors during our developments. Running the test suite successfully gave us enough confidence about the acceptable quality of each compiler version to be released to a next stage."
- "... SuperTest helped us to find the problems in our compiler far more quickly than otherwise would have been possible. In comparison with the license and maintenance fees, we saved more than twice the amount in our efforts in improving the compiler quality."

## SuperTest Features

- Nearly 3,000,000 language conformance and compiler quality checks
- Easy to install
- Parallel testing
- Powerfull and flexible test generator
- Positive and negative testing
- User-friendly interface
- Easy addition of tests
- Seamless integration in compiler development environment
- Clear reporting in HTML

Statistics	LOG-00 bi	LOG-01 bi	LOG-02 bi	LOG-03 bi	LOG-04 bi	LOG-04
	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed
Compile errors	0/0/4	0/0/4	0/0/4	0/0/4	0/0/4	0/0/4
2.1	0/0/32	0/0/32	0/0/32	0/0/32	0/0/32	0/0/32
2.2	0/0/253	0/0/253	0/0/253	0/0/253	0/0/253	0/0/253
3.0	0/3/119	0/3/119	0/3/119	0/3/119	0/3/119	0/3/119
3.1	0/0/342	0/0/342	0/0/342	0/0/342	0/0/342	0/0/342
3.2	2/5/447	2/5/447	2/5/447	2/6/446	2/6/446	2/6/446
3.3	0/1/6	0/1/6	0/1/6	0/1/6	0/1/6	0/1/6
3.4	1/7/226	1/7/226	1/7/226	1/7/226	1/7/226	1/7/226
3.5	0/0/175	0/0/175	0/0/175	0/0/175	0/0/175	0/0/175
3.6	0/9/20	0/9/20	0/9/20	0/9/20	0/9/20	0/9/20
3.7	0/1/92	0/1/92	0/1/92	0/1/92	0/1/92	0/1/92
3.8	0/0/218	0/0/218	0/0/218	0/0/218	0/0/218	0/0/218
4	0/0/2668	0/0/2668	0/0/2668	0/0/2668	0/0/2668	0/0/2668
major	3/26/4602	3/26/4602	3/26/4602	3/27/4601	4/27/4600	4/27/4600
Subtotal	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed	failed/check/passed
Run errors	0/0/3	0/0/3	0/0/3	0/0/3	0/0/3	0/0/3
2.1	0/0/31	0/0/31	0/0/31	0/0/31	0/0/31	0/0/31
2.2	0/0/253	0/0/253	0/0/253	0/0/253	0/0/253	0/0/253
3.0	0/0/74	0/0/74	0/0/74	0/0/74	0/0/74	0/0/74
3.1	0/0/341	1/0/381	0/0/341	1/0/381	0/0/341	0/0/382
3.2	1/0/381	0/0/5	0/0/5	1/0/381	1/0/381	0/0/5
3.3	0/0/5	1/0/146	0/0/146	0/0/5	1/0/146	0/0/147
3.4	1/0/146	0/0/162	0/0/162	0/0/146	0/0/161	0/0/161
3.5	0/0/11	0/0/11	0/0/11	0/0/11	0/0/11	0/0/11
3.6	0/0/73	0/0/73	0/0/73	0/0/73	0/0/73	0/0/73
3.7	9/0/209	9/0/209	9/0/209	4/0/2664	9/0/209	2/0/216
3.8	10/0/2658	10/0/2658	10/0/2658	4/0/2664	4/0/2664	4/0/2664
4	10/0/2658	21/0/4347	15/0/4353	15/0/4353	15/0/4352	6/0/4361
major	21/0/4347	24/26/4581	18/26/4587	18/27/4586	19/27/4585	10/27/4594
Subtotal	24/26/4581	24/26/4581	18/26/4587	18/27/4586	19/27/4585	10/27/4594
Total	LOG-00 bi	LOG-01 bi	LOG-02 bi	LOG-03 bi	LOG-04 bi	LOG-04
Regression	failed	failed	failed	failed	failed	passed
3.3.2.2/tellipais	passed	passed	passed	passed	passed	passed
3.3.2.2/xsprit5020a	failed	failed	failed	failed	failed	passed
3.5.4/teich	passed	passed	passed	passed	passed	passed
3.6.5.3/tspr1981	failed	failed	failed	failed	failed	passed
4.0/tspr1182	failed	failed	failed	failed	failed	passed
4.8/tbysman3	failed	failed	failed	failed	failed	passed
4.9.6.5/11	failed	failed	failed	failed	failed	passed
4.9.6.5/2	failed	failed	failed	failed	failed	passed
4.9.6.5/tspr4261	failed	failed	failed	failed	failed	passed
4.9.6.5/tspr4265	failed	failed	failed	failed	failed	passed