

Sun Studio Support For Intervals

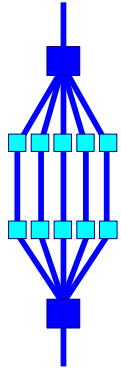
Ruud van der Pas

**Senior Staff Engineer
Compilers, Libraries and Performance Technologies
Sun Microsystems, Menlo Park, CA, USA**

***Third International Workshop On Reliable
Engineering Computing***

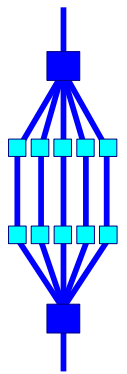
***Center for Reliable Engineering Computing
Georgia Tech Savannah, GA, USA
February 20-22, 2008***

Outline



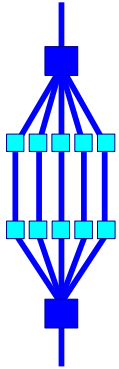
- *The Sun Studio Compilers*
- *Pointers to more information, downloads, etc*
- *Compiler support for Interval Arithmetic*

The Sun Studio™ Compilers

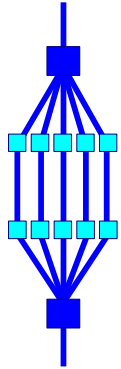


 **For Free**

Supported Platforms



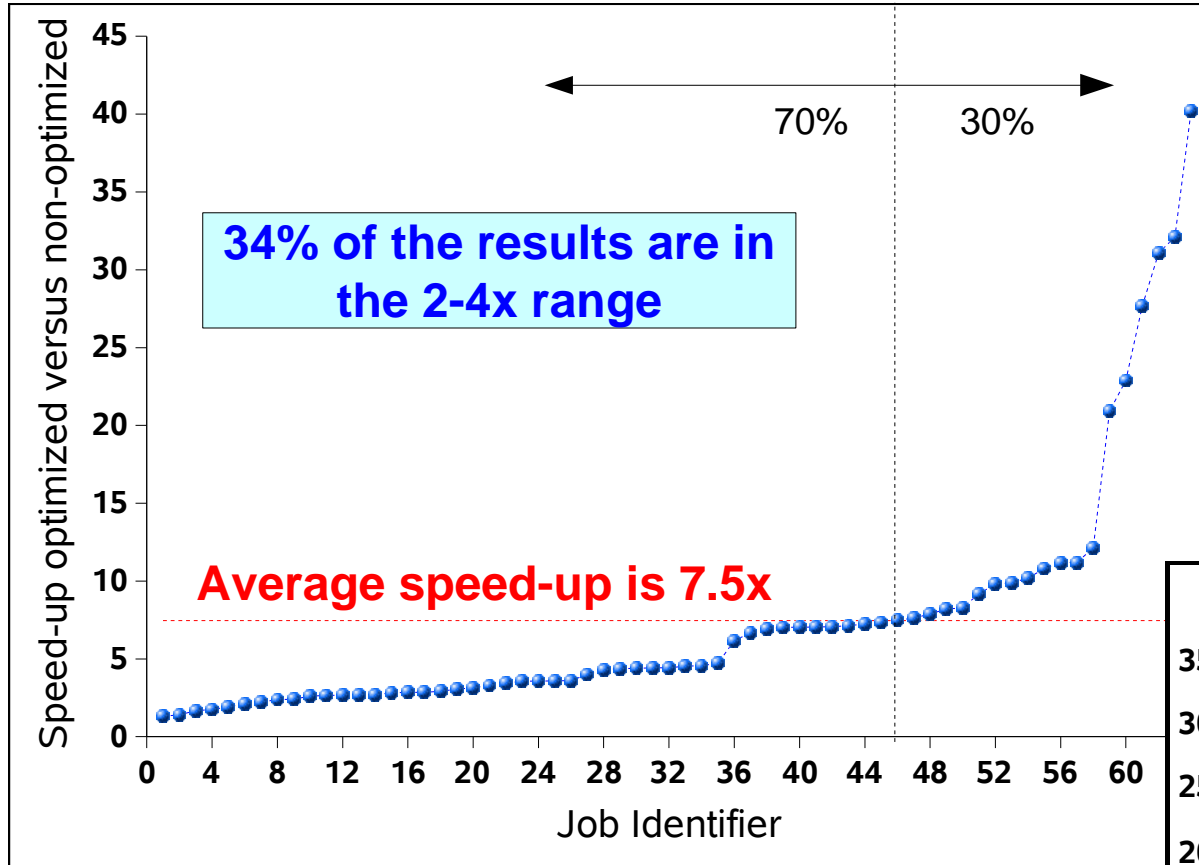
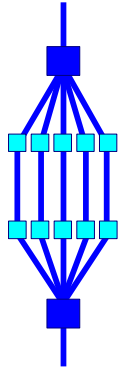
- *The Sun Studio compilers and tools are supported on various AMD and Intel processors, as well as all SPARC processors*
 - *SPARC has the `siam` instruction to better support interval arithmetic*
- *Operating Systems supported*
 - *Solaris*
 - *Certain Linux implementations (RedHat, Suse)*
- *Regarding Interval Arithmetic*
 - *Fortran has the best and easiest support*
 - ✓ *Intervals are a built in, native, data type*
 - *C++ support is through a class library*



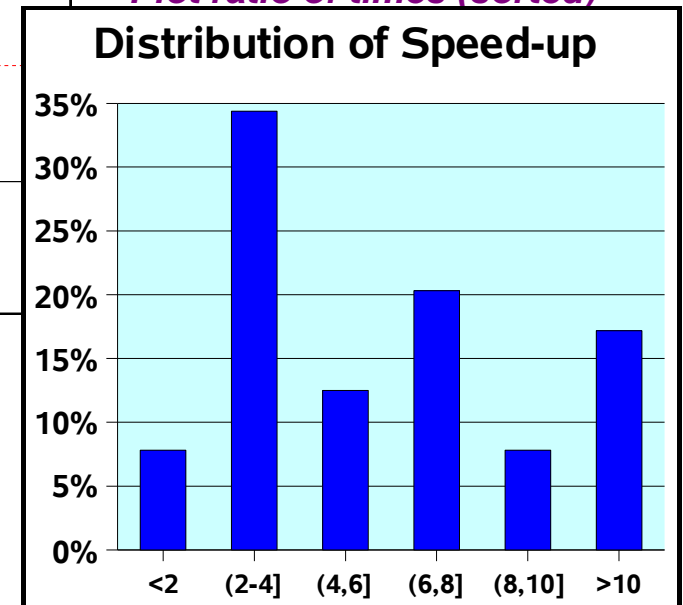
Sun Studio Compilers and Tools

- **Fortran (f95), C (cc) and C++ (CC) compilers**
 - *Support sequential optimization, automatic parallelization and OpenMP*
- **Sun Performance Analyzer**
 - *Languages supported: Fortran, C, C++ and Java*
 - *Parallel: AutoPar, OpenMP, POSIX Threads (and MPI)*
- **Sun Thread Analyzer**
 - *Languages supported: Fortran, C, C++*
 - *Parallel: OpenMP, POSIX Threads, Solaris Threads*
- **Sun Studio Integrated Development Environment**
- **Additional tools**

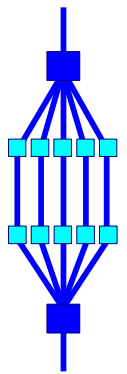
The Benefit Of A Compiler



- ◆ *Twenty real user codes*
 - ✓ *Chemistry, Physics, Mathematics*
- ◆ *Identified by Job ID*
 - ✓ *Different versions, different jobs*
- ◆ *Compiled and ran:*
 - ✓ *No optimization, full optimization*
- ◆ *Plot ratio of times (sorted)*



SF E6900 - US IV @ 1200MHz
Single processor results



Sun Microsystems - Sun Developer Network (SDN)

The Sun Web Site for Developers

Sun ▾ Java ▾ Solaris ▾ Communities ▾ My SDN Account ▾ Join SDN ▾

Sun Developer Network (SDN) » search tips

APIs Downloads Products Support Training Participate

SDN: A Community for Sun Developers

SXDE 1/08
MOVE TO THE LEADING EDGE
» Download Now
» Learn More
» Free DVD

LATEST TECH TRENDS ON CAMPUS
» CampusCast
» Student Developer Resources

NETBEANS 6.0 THE ONLY IDE YOU NEED
» Download Now
» Learn More
» Free DVD

Featured Content

Student Developers
Where can you find hot technologies, open-source communities, and job opportunities? Sun is looking for students who are ready to innovate and create the future. » Learn More

Sun Developer Community

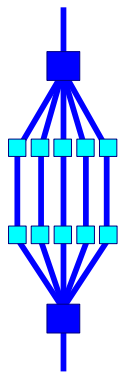
SDN Videos
Footnote on Blu-ray Disc Java
In this video interview, Sun's Blu-ray Disc Java (BDJ) architect Bill Foote talks about this powerful technology and shows some

Get Involved, Join a Community
Mobile & Embedded
The Mobile & Embedded Community is a gathering place that enables and empowers

Quiz
Project Darkstar facilitates what critical community function?
 a) Blogging

<http://developers.sun.com>

java EE SDK Fuels Efficiency
NetBeans Simple, Intuitive IDE
Run Desktop Apps Faster with Java



The web site for the Sun Studio Compilers and Tools

Sun Developer Network (SDN) search tips Search

APIs Downloads Products Support Training Participate

Developers Home > Sun Studio

Sun Studio Topics

THE RIGHT TOOLS
Make all the difference. Choose the best tools for your development environment.
→ Download Sun Studio today

downloads

Overview Features Documentation Community Support Downloads

At a Glance | What's New | **Topics** | Product Tour | Heroes | Participate

C/C++/Fortran 95 Compilers

The Sun C, C++, and Fortran compilers include advanced features for developing applications on Sun Solaris SPARC and x86/x64 platforms. They utilize a common optimizing backend code generator, and accept standard C, C++, and Fortran with extensions.

The Sun Studio Performance Tools

The Sun Studio performance tools are designed to help answer questions about application performance. This article discusses the kinds of performance questions that users typically ask.

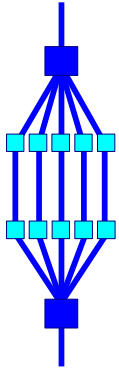
Debugging

Successful program debugging is more an art than a science. dbx is an interactive, source-level, post-mortem and real-time command-line debugging

Edition 9/07
Get the best OS platform for Java, Web 2.0, and C/C++/Fortran developers.
» Free DVD
» Download now
» Learn More

Vote Total
43

Support for Interval Arithmetic



- *Available in Sun Studio since 2000 !*
- *A closed interval system in which all expressions (including singularities and indeterminate forms) are defined*
 - *Examples: $1/0$, x^y with $x=y=0$, operations involving $+\infty$ and/or $-\infty$*
- *Domain constraints on intrinsic functions are gracefully handled*
 - *Example: $SQRT([-1, +1]) = [0, 1]$*

Documentation on Interval Arithmetic support



Sun Developer Network (SDN)

[APIs](#) [Downloads](#) [Products](#) [Support](#) [Training](#) [Participate](#)
» search tips

Developers Home > Sun Studio >

Sun Studio

Sun Studio: Numerical Computation



http://developers.sun.com/sunstudio/overview/topics/numerics_index.html

Latest Documentation: Sun Studio 12

Reference Manuals

- **Numerical Computation Guide**

A complete application programmer's handbook to understanding the data structures and operations made available by hardware, system software, and software libraries that together implement IEEE Standard 754. IEEE Standard 754 makes it easier to write numerical applications. It is a solid, well-thought-out basis for computer arithmetic that advances the art of numerical programming. (November, 2005)

- **Fortran 95 Interval Arithmetic Programming Reference**

Documents the intrinsic INTERVAL data types in the Sun Fortran 95 compiler (f95). (November, 2005)

- **C++ Interval Arithmetic Programming Reference**

Documents the C++ interface to the C++ interval arithmetic library provided with the Sun C++ compilers. (November, 2005)

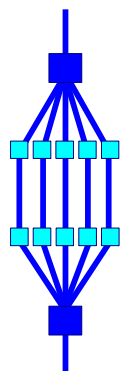
Standard for
Binary Floating-
Point Arithmetic

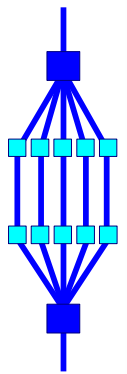
Compilers and Tools Topics

- C/C++/Fortran Compilers
- High Performance Technical Computing
- Performance Analyzer
- Debugging (dbx)
- Sun Performance Library
- Support
- Latest News

Math Library Release Notes

Related Links






Sun Studio Code Samples

Code samples (Fortran and C++)


Interval Arithmetic Code Samples			
Compiler Name/Description	Documentation	Code	
C++	C++ Interval Arithmetic Examples A listing of all the code examples in the C++ Interval Arithmetic Programming Reference	C++ Interval Arithmetic code example support documents: <ul style="list-style-type: none">• C++ Interval Arithmetic code examples compilation	TAR
Fortran 95	General Fortran 95 Interval Arithmetic Examples A tar file containing the Fortran 95 interval arithmetic examples included in the examples directory of the installed product	Provided in the README file included in the tar file	TAR
Fortran 95	Fortran 95 Interval Arithmetic Examples A listing of all the code examples in the Fortran 95 Interval Arithmetic Programming Reference.	Fortran 95 Interval Arithmetic code example support documents:	TAR

Train. Learn. Win.
Solaris Training Instant Win and Sweepstakes. Chance to Win \$50,000.
» Register Now

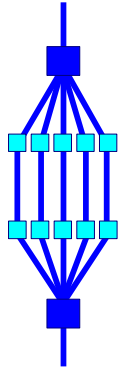


<http://developers.sun.com/sunstudio/documentation/codesamples/index.jsp>

Installation & configuration support for Solaris Express Developer Edition.
» Get Support Now



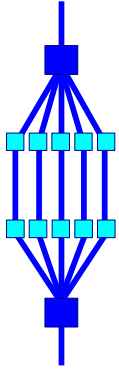
Making Porting Easy
Test your applications on Solaris for free with



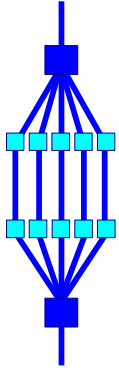
Intervals in Fortran - Key Features

- ❑ *Native Interval Data Type*
- ❑ *Fortran Intrinsic Functions (e.g. EXP, LOG, SIN, ...)*
- ❑ *Interval Specific Intrinsic (Set) Functions*
 - *width, midpoint, hull, union, subset, element of,*
- ❑ *Order Relations (e.g. “certainly less than”)*
- ❑ *Input/Output can be handled in different ways*
- ❑ *Integer Power understands Dependence*
- ❑ *Mixed mode interval expressions*
- ❑ *Context dependent literal interval constants*

Support in C++



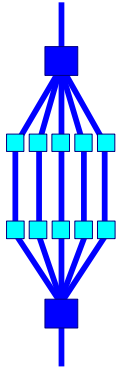
- ❑ *Implemented as class library*
- ❑ *SPARC only*
- ❑ *Same functionality as Fortran*
 - *No mixed mode support because of C++ language standard and not a native data type*



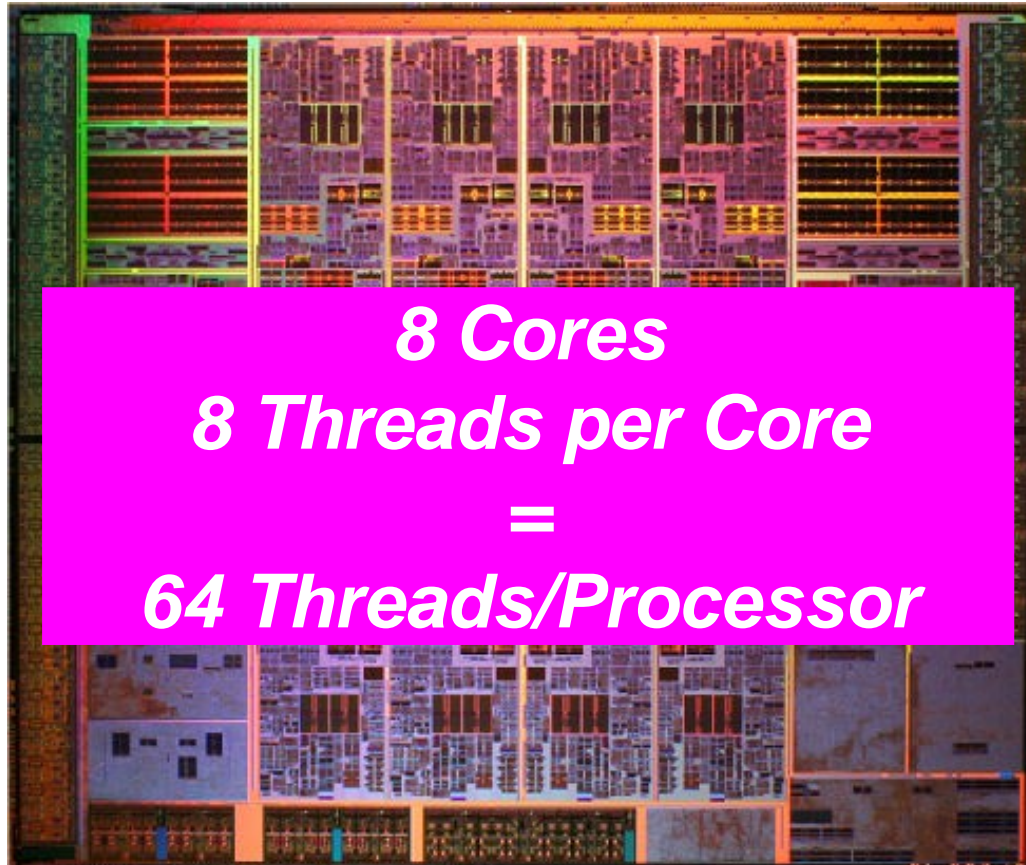
Multicore Processor Technology Could Be Of Interest To This Group

Why ?

Many Interval Algorithms Exhibit An Natural Level Of Parallelism

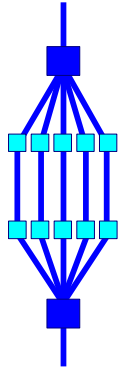


The UltraSPARC T2 Processor



Think Parallel !

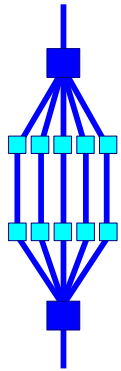
IEEE 754 compliant



***Although The Hardware Has Increased
Support for Parallelism, The Issues Are
The Same Still***

So ?

***Increased Need For Tools To Assist
The Developer To Write Parallel
Programs***



OpenMP™

<http://www.openmp.org>

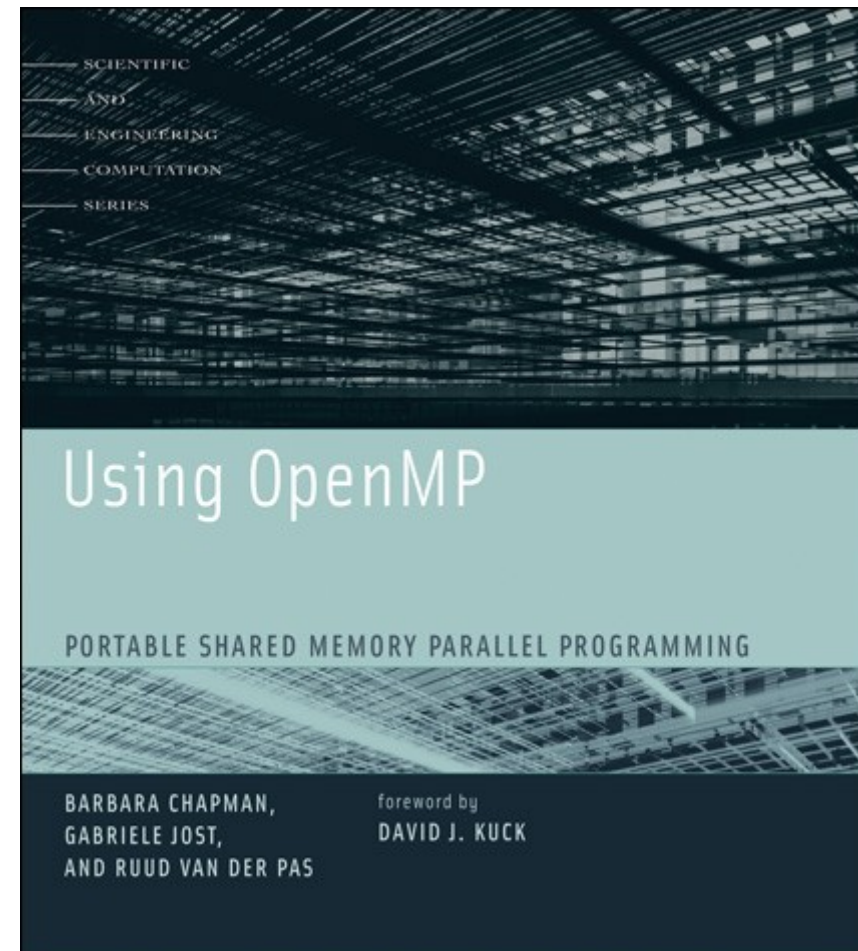
“Using OpenMP”
Portable Shared Memory Parallel Programming

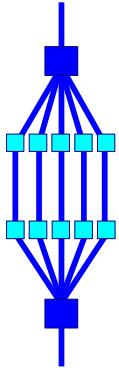
Chapman, Jost, van der Pas

MIT Press, Oct. 2007, 35 \$US

ISBN-10: 0-262-53302-2

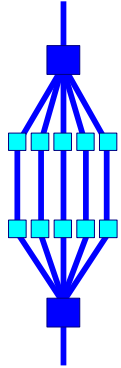
ISBN-13: 978-0-262-53302-7





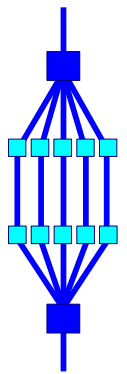
Wrap Up - Why Use Sun Studio?

- ❑ High quality, officially supported product
- ❑ Available on a wide range of systems and platforms
- ❑ State of the art performance
- ❑ Support for Interval Arithmetic
- ❑ Easy parallelization with OpenMP
 - Great programming model for multicore processors
- ❑ Additional tools supported, for example
 - Sun Performance Analyzer
 - Sun Studio Thread Analyzer
- ❑ For free!



Thank You !

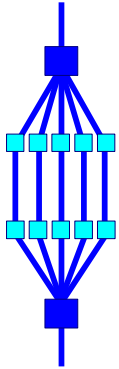
ruud.vanderpas@sun.com



Back Up Slides

Support For Intrinsic Functions

All Fortran intrinsic functions have an interval counterpart if they either return a REAL, or accept a REAL type argument



```
% cat -n cos.f95
1   program demo
2
3   print *, 'cos (-0.5)           = ', cos (-0.5D0)
4   print *, 'cos (+0.5)           = ', cos (+0.5D0)
5   print *, 'cos [-0.5,+0.5]     = ', cos ([-0.5,+0.5])
6
7   stop
8   end
```

```
% f95 -o cos -xia cos.f95
```

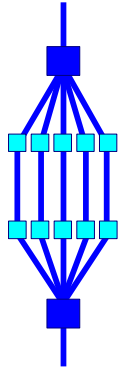
```
% ./cos
```

```
cos (-0.5)           = 0.8775825618903728
```

```
cos (+0.5)           = 0.8775825618903728
```

```
cos [-0.5,+0.5]     = [0.87758256189037264,1.0]
```

Integer Powers



The Dependence Problem:

$$[-1, 2] * [-1, 2] = [-2, 4]$$

The Sun Compiler will do
the right thing:

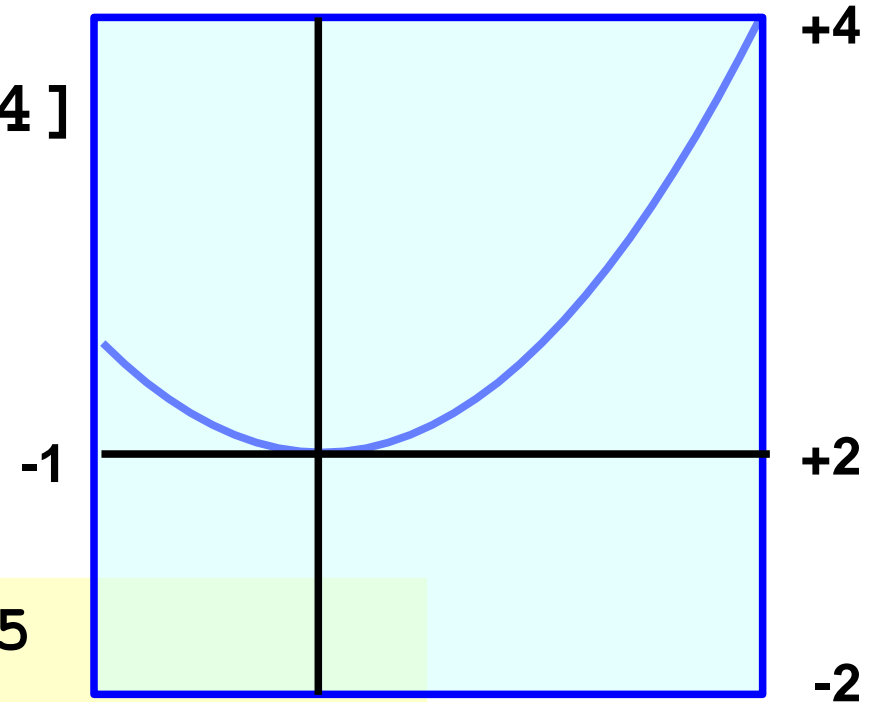
```
% f95 -o pow -xia pow.f95
```

```
% ./pow
```

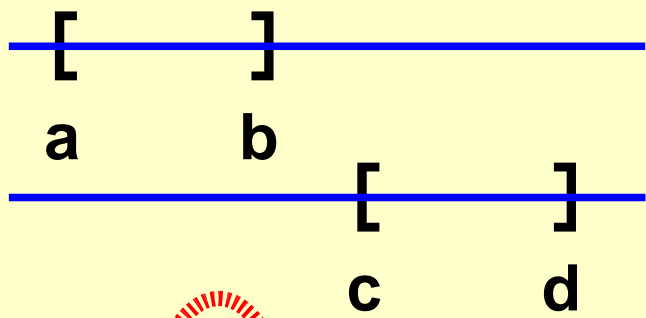
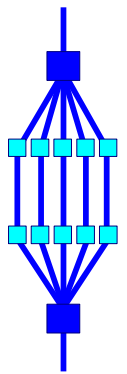
```
X      = [ -1.000000000,  2.000000000]
```

```
X*X    = [ -2.000000000,  4.000000000]
```

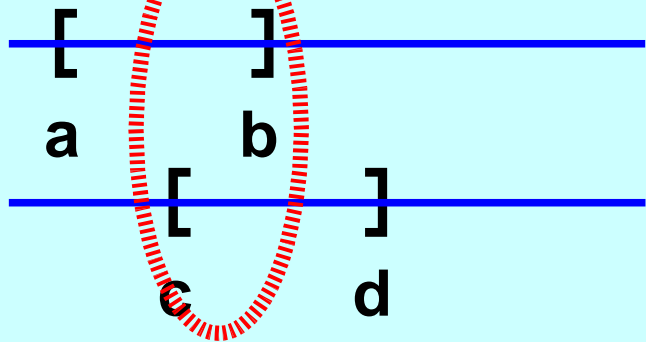
```
X**2  = [  0.000000000,  4.000000000]
```



Order Relations - What To Do ?



[a,b] certainly less than [c,d]



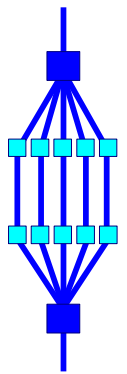
[a,b] possibly less than [c,d]

Implementation in the Sun compiler:

One of {C, P, S}, followed by LT/LE/EQ/NE/GE/GT

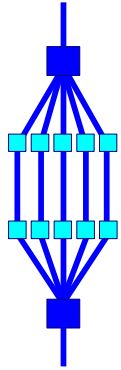
Example: A .CLT. B

Set-Theoretic Interval Operators



Name	Math. Notation	Fortran	Result Type
Interval hull	$X \cup Y$	<code>X .IH. Y</code>	Interval
Intersection	$X \cap Y$	<code>X .IX. Y</code>	Interval
Disjoint	$X \cap Y = \emptyset$	<code>X .DJ. Y</code>	Logical
Element	$r \in Y$	<code>R .IN. Y</code>	Logical
Interior	$\underline{X} < \underline{Y}$ and $\overline{X} < \overline{Y}$	<code>X .INT. Y</code>	Logical
Proper subset	$X \subset Y$	<code>X .PSB. Y</code>	Logical
Proper superset	$X \supset Y$	<code>X .PSP. Y</code>	Logical
Subset	$X \subseteq Y$	<code>X .SB. Y</code>	Logical
Superset	$X \supseteq Y$	<code>X .SP. Y</code>	Logical

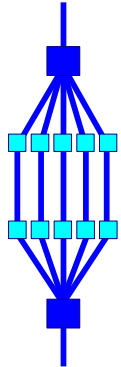
Interval Specific Intrinsic



Name	Definition	Name	Result Type
Infimum	$\text{inf}([a,b]) = a$	INF	REAL
Supremum	$\text{sup}([a,b]) = b$	SUP	REAL
Width	$w([a,b]) = b-a$	WID	REAL
Midpoint	$(a+b) / 2$	MID	REAL
Magnitude	$\max(a , b)$	MAG	REAL
Mignitude	$\min(a , b)^*$	MIG	REAL
Empty Test	TRUE if empty	ISEMPTY	LOGICAL
Number Of Digits	Max. digits	NDIGITS	INTEGER

**) Returns 0 if $0 \in [a,b]$*

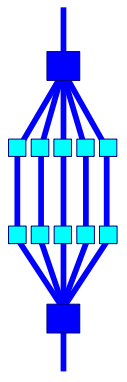
Example Code



Program Demo

```
logical :: not_done = .true.  
interval(kind=8)      :: ai, bi  
write(*,*) 'Please give values for A and B'  
do while ( not_done )  
    read(*,*,end=9000) ai, bi  
  
    write(*,9010) '+',ai,'+',bi,ai+bi  
    write(*,9010) '-',ai,'-',bi,ai-bi  
    write(*,9010) '*',ai,'*',bi,ai*bi  
    write(*,9010) '/',ai,'/',bi,ai/bi  
    write(*,*)  
end do  
  
9000  continue  
      stop  
9010  format(1X,'A',1X,(A),1X,'B =',VF17.4,1X,(A), &  
           1X,VF17.4,' = ',VF17.4)  
end
```

Example Closed Interval System



```
% f95 -xia math.f95  
% ./a.out
```

Please give values for A and B

$$\begin{aligned} A + B &= [-1.0000, 3.0000] + [1.0000, 2.0000] = [0.0000, 5.0000] \\ A - B &= [-1.0000, 3.0000] - [1.0000, 2.0000] = [-3.0000, 2.0000] \\ A * B &= [-1.0000, 3.0000] * [1.0000, 2.0000] = [-2.0000, 6.0000] \\ A / B &= [-1.0000, 3.0000] / [1.0000, 2.0000] = [-1.0000, 3.0000] \end{aligned}$$

$$\begin{aligned} A + B &= [1.0000, 2.0000] + [-1.0000, 3.0000] = [0.0000, 5.0000] \\ A - B &= [1.0000, 2.0000] - [-1.0000, 3.0000] = [-2.0000, 3.0000] \\ A * B &= [1.0000, 2.0000] * [-1.0000, 3.0000] = [-2.0000, 6.0000] \\ A / B &= [1.0000, 2.0000] / [-1.0000, 3.0000] = [-Inf, Inf] \end{aligned}$$

