CFP2000 Result

Copyright ©1999-2005, Standard Performance Evaluation Corporation

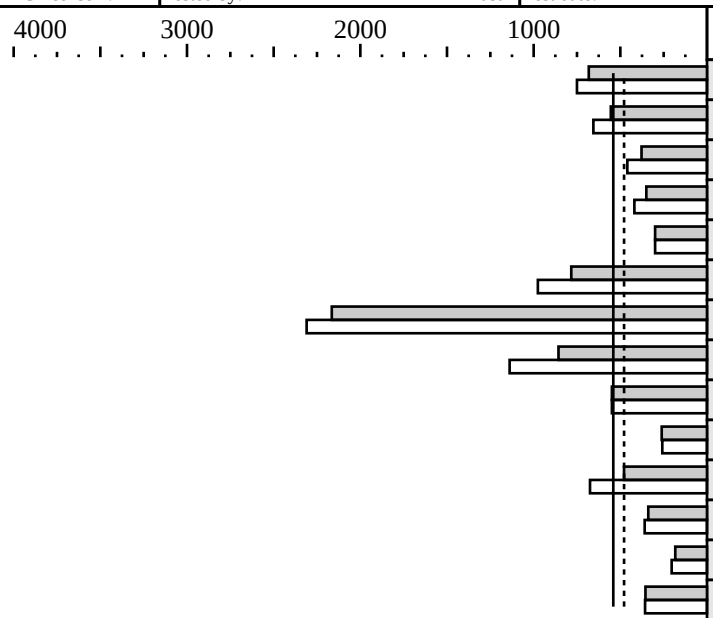
IBM Corporation

IBM System p5 575 (1900 Mhz, 16 CPU, SLES)

SPECfp_rate2000 = 541

SPECfp_rate_base2000 = 478

SPEC license #: 11 | Tested by: IBM Austin | Test date: Oct-2006 | Hardware Avail: Feb-2006 | Software Avail: Dec-2006



Benchmark	Base Copies	Base Runtime	Base Ratio	Copies	Runtime	Ratio
168.wupwise	32	87.1	682	32	79.2	750
171.swim	32	207	555	32	176	656
172.mgrid	32	177	378	32	145	460
173.applu	32	223	350	32	186	419
177.mesa	32	174	299	32	174	299
178.galgel	32	137	783	32	110	976
179.art	32	44.6	2165	32	41.8	2310
183.equake	32	56.3	857	32	42.4	1139
187.facerec	32	129	549	32	129	549
188.ammp	32	313	261	32	317	258
189.lucas	32	155	479	32	110	675
191.fma3d	32	230	339	32	217	360
200.sixtrack	32	223	183	32	200	204
301.apsi	32	272	355	32	270	357

Hardware

CPU:	POWER5+
CPU MHz:	1900
FPU:	Integrated
CPU(s) enabled:	16 cores, 8 chips, 2 cores/chip (SMT on)
CPU(s) orderable:	8,16 core
Parallel:	No
Primary Cache:	64 KB I + 32 KB D on chip per core
Secondary Cache:	1920 KB I+D on chip per chip
L3 Cache:	36 MB unified off chip per chip
Other Cache:	None
Memory:	64 GB (32x2GB)
Disk Subsystem:	1x73GB SCSI, 15K RPM
Other Hardware:	

Software

Operating System:	SLES SUSE Linux Enterprise Server 10 (ppc) VERSION = 10 w/2.6.16.21-0.8-ppc64 Linux kernel
Compiler:	IBM XL C/C++ Advanced Edition V8.0.1 for Linux IBM XL Fortran Advanced Edition V10.1.1 for Linux
	Other software: - IBM Engineering and Scientific Subroutine Library (ESSL) for Linux - Version 4.2.5
File System:	reiserfs
System State:	Multi-User

Notes/Tuning Information

```
+FDO
    Feedback directed optimization enabled by:  PASS1=-qpdf1  PASS2=-qpdf2

FP compilers
   C: invoked as xlc
   Fortran 77 and Fortran 90: invoked as xlf90, except as noted below

FP Portability Flags
  -qfixed used in:        168.wupwise, 171.swim, 172.mgrid, 173.applu,
                          178.galgel, 200.sixtrack, 301.apsi
  -qsuffix=f=f90 used in: 178.galgel, 187.facerec, 189.lucas, 191.fma3d

FP Base Optimization Flags:
  C:       +FDO -O5
  Fortran: +FDO -O5
```

Copyright ©1999-2005, Standard Performance Evaluation Corporation

Notes/Tuning Information (Continued)

```
Floating Point Peak Flags

168.wupwise
    +FDO -O5 -qsave -lmass
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
171.swim
    +FDO -O5
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
172.mgrid
    +FDO -O4 -q64
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
173.applu
    +FDO -O5 -q64
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
177.mesa
    basepeak=1
178.galgel
    Fortran invoked as xlf90_r
    +FDO -O5 -qessl -lessl -lmass
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
179.art
    +FDO -O5
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
183.equake
    +FDO -O5
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
187.facerec
    basepeak=1
188.ammp
    +FDO -O3 -qalign=linuxppc
189.lucas
    +FDO -O3 -qarch=auto -qtune=auto
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
191.fma3d
    +FDO -O5
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
200.sixtrack
    +FDO -O3 -qarch=auto -qtune=auto
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
301.apsi
    Fortran invoked as xlf90_r
    +FDO -O5 -qessl
    -B/usr/share/libhugetlbfs/ -tl -Wl,--hugetlbfs-link=BDT
    extra_libs = -lessl


System Settings:
-- ulimit stack size set to unlimited


SMT: Acronym for 'Simultaneous Multi-Threading'.  A processor technology that allows
     the simultaneous execution of multiple thread contexts within a single processor
     core. SMT is enabled by default.

Large pages reserved as follows by root user:
    echo 960 > /proc/sys/vm/nr_hugepages
```

Copyright ©1999-2005, Standard Performance Evaluation Corporation

Notes/Tuning Information (Continued)

System configured with libhugetlbfs library for application access to large pages
Environment variables set as follows:

```
export HUGETLB_MORECORE=yes
```

Each process was bound to a cpu using submit= with the taskset command

```
submit = taskset -p -c $SPECUSERNUM $$ >/dev/null ; $command
```