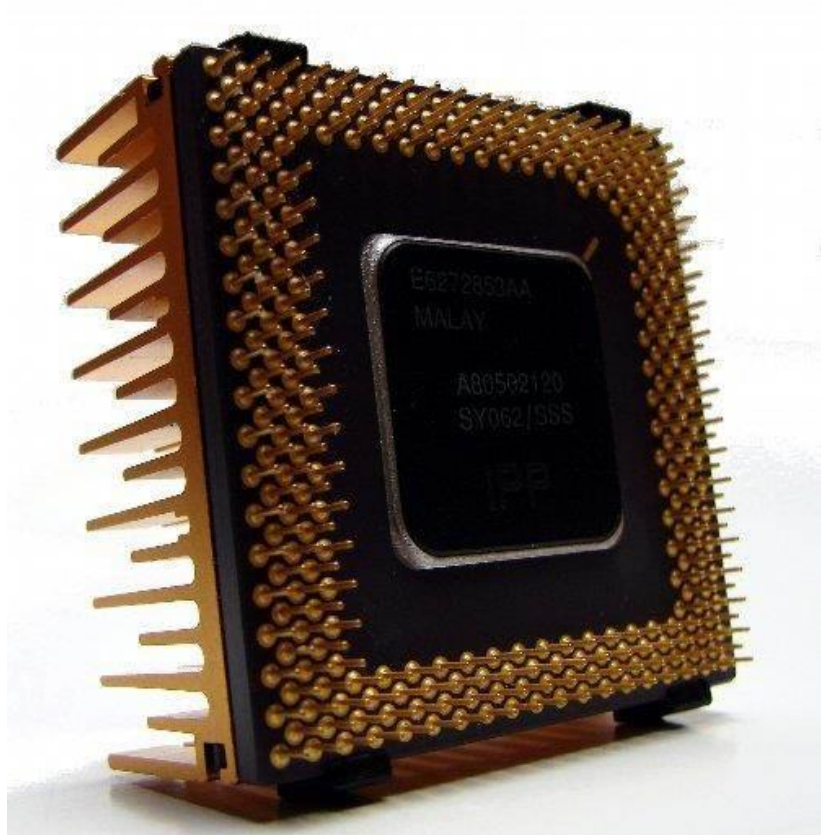


PRÁCTICA 1

MATERIAL ADICIONAL F-II (3)

- SPEC



2017

Proyecto de evaluación del rendimiento

F-II. Implementación de un benchmark reducido y evaluación del procesamiento de arquitecturas PC convencionales

Arquitectura de los Computadores

Grado en Ingeniería Informática

Dpto. Tecnología Informática y Computación

Universidad de Alicante

MATERIAL F-II (3)

PROYECTO DE EVALUACIÓN DEL RENDIMIENTO

I. INTRODUCCIÓN A SPEC

Introducción

Standard Performance Evaluation Corporation (SPEC), es un consorcio sin ánimo de lucro que incluye a vendedores, integradores de sistemas, universidades, grupos de investigación, publicadores y consultores de todo el mundo. Tiene dos objetivos: crear un benchmark estándar para medir el rendimiento de computadores y controlar y publicar los resultados de estos tests.

SPEC CPU2000

SPEC CPU2000 es un benchmark producido por la SPEC. Fue creado con el fin de proveer una medida de rendimiento que pudiese ser usada para comparar cargas de trabajo intensivas en cómputo en distintos computadores. Contiene dos benchmark suites: **CINT2000** para medir y comparar el rendimiento de computación intensiva de enteros, y **CFP2000** para medir y compara el rendimiento de computación intensiva en flotantes.

La "C" en CINT2000 y CFP2000 denotan que son benchmarks a nivel de componentes, en oposición a benchmarks de todo el sistema. Al ser intensivos en cálculos, miden el rendimiento del procesador del computador, la arquitectura de la memoria y el compilador. El CINT2000 y el CFP2000 no fuerzan la entrada/salida (unidades de disco), trabajo en red o gráficos.

Contenido

SPEC provee lo siguiente en el paquete CPU2000: las herramientas del CPU2000 para compilar, ejecutar y validar los benchmarks en una variedad de sistemas operativos, el código fuente de las herramientas, de manera que puedan ser compiladas para los sistemas no cubiertos por las herramientas pre-compiladas, el código fuente de los benchmarks, herramientas para la generación de informes de rendimiento, reglas de ejecución e informes que definen cómo deberían ser usados los benchmarks para producir resultados estándar la documentación. SPEC CPU2000 incluye herramientas para la mayoría de los sistemas operativos Unix y Windows.

CINT2000 contiene 11 aplicaciones escritas en C y una en C++ (252.eon) que son usadas como benchmarks.

- 164.gzip: Utilidad de compresión de datos.
- 175.vpr: Direccionamiento y ubicación de circuitos FPGA.
- 176.gcc: Compilador C.
- 181.mcf: Costo mínimo de flujo de red.
- 186.crafty: Programa de ajedrez.
- 197.parser: Procesamiento de lenguaje natural.

- 252.eon: Efectos producidos por distintas fuentes de luz.
- 253.perlbmk: Perl.
- 254.gap: Teoría de grupo computacional.
- 255.vortex: Base de datos orientada a objetos.
- 256.bzip2: Utilidad de compresión de datos.
- 300.twolf: Simulador de ubicación y ruteo.

CFP2000 contiene 14 aplicaciones (seis en Fortran77, cuatro en FORTRAN90 y cuatro en C) que son usadas como benchmarks:

- 168.wupwise: Cromodinámica de cuantos.
- 171.swim: Modelado de aguas poco profundas.
- 172.mgrid: Multi-grilla en campos potenciales 3D.
- 173.applu: Ecuaciones diferenciales parciales parabólicas/elípticas.
- 177.mesa: Biblioteca de gráficos 3D.
- 178.galgel: Dinámica de fluidos: análisis de inestabilidad oscilatoria.
- 179.art: Simulación de red neuronal: teoría de la resonancia adaptativa.
- 183.equake: Simulación de elementos finitos: modelado de terremotos.
- 187.facerec: Reconocimientos de imágenes: reconocimiento de rostros.
- 188.amp: Química computacional.
- 189.lucas: Teoría de los números: prueba de primalidad.
- 191.fma3d: Simulación de elementos finitos en choque.
- 200.sixtrack: Modelo de acelerador de partículas.
- 301.apsi: Problemas de temperatura, viento y distribución de contaminantes.

Resultados

El SPEC2000 provee cuatro medidas para cada componente (consistente en la media geométrica de los tiempos de ejecución de los programas). Esto es porque tiene dos clasificaciones. La primera es en cuanto a la optimización del compilador por el usuario:

- base ("conservadora"): sirve para encuadrar a los usuarios que compilan con las opciones de optimización generales ofrecidas por el compilador, de manera que tiene una serie de referencias establecidas para usar las opciones del compilador (por ej. deben usarse las mismas opciones en el mismo orden en todos los benchmarks de un mismo lenguaje, no más de cuatro opciones de optimización, y no deben usarse banderas de reivindicación).
- no base ("agresiva"): intenta encuadrar a los usuarios que intentan lograr el mejor rendimiento de sus programas. Son opcionales y tienen requerimientos mucho menos estrictos (por ej. pueden

usarse diferentes opciones de compilación para distintos programas escritos en el mismo lenguaje)

La segunda, se refiere a la cantidad de tareas que se ejecutan al mismo tiempo:

- no rate (monotarea): corresponde a la velocidad de ejecución de una sola tarea.
- rate (multitarea): corresponde a la capacidad de la máquina de llevar a cabo un cierto número de tareas similares. Tradicionalmente usado para medir el rendimiento de multiprocesadores.

La combinación de estas dos clasificaciones nos otorga 4 medidas distintas para el CINT2000 y 4 para el CFP2000. SPEC usa una Sun Ultra5_10 con un procesador de 300 MHz como máquina de referencia.

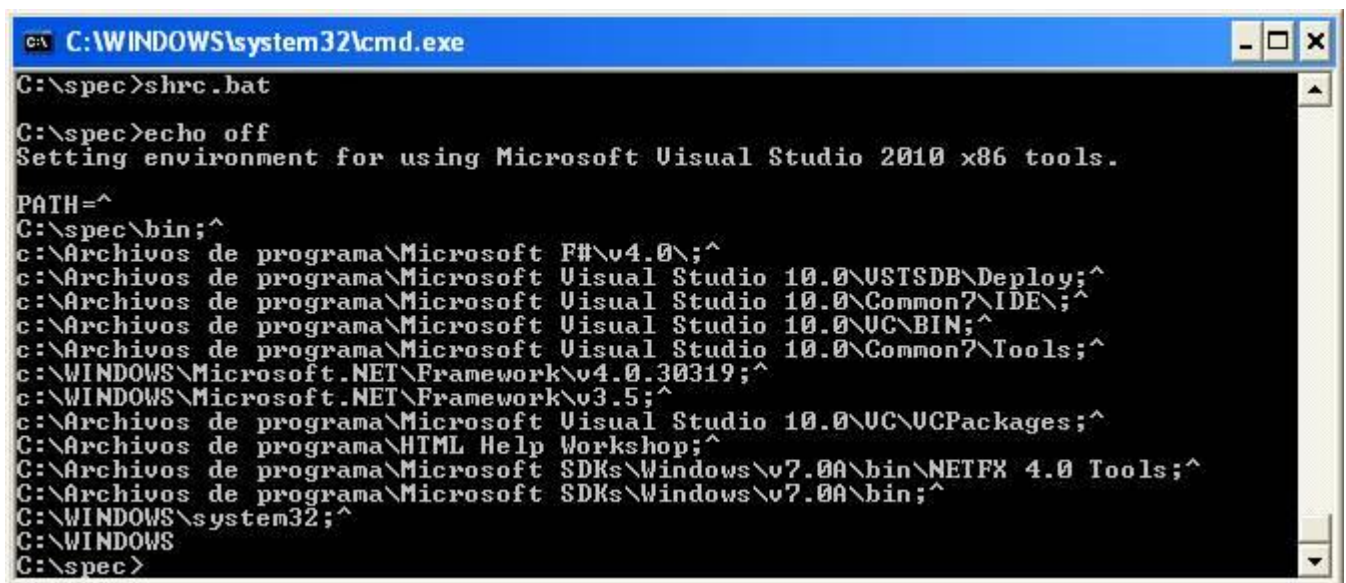
Preparación del entorno

Después de la instalación de SPEC (algo que ya está hecho en el laboratorio), hay que editar el fichero shrc.bat y hacer las modificaciones necesarias para que SPEC encuentre la ruta del compilador que usaremos, en nuestro caso será el Visual Studio.

Se recomienda al alumno leer los comentarios del archivo y localizar las líneas que hay que modificar, que concretamente son las siguientes:

```
set SHRC_COMPILER_PATH_SET=yes
call "C:\Archivos de programa\Microsoft Visual Studio 10.0\VC\bin\vcvars32.bat"
```

El archivo vcvars32.bat establece las variables de entorno apropiadas para habilitar generaciones desde la línea de comandos de 32 bits. El siguiente paso es ejecutar el archivo shrc.bat para establecer las variables de entorno. El resultado debe ser el siguiente:



```
C:\WINDOWS\system32\cmd.exe
C:\spec>shrc.bat
C:\spec>echo off
Setting environment for using Microsoft Visual Studio 2010 x86 tools.
PATH=^
C:\spec\bin;^
c:\Archivos de programa\Microsoft F#\v4.0\;^
c:\Archivos de programa\Microsoft Visual Studio 10.0\USTSDB\Deploy;^
c:\Archivos de programa\Microsoft Visual Studio 10.0\Common7\IDE\;^
c:\Archivos de programa\Microsoft Visual Studio 10.0\VC\BIN;^
c:\Archivos de programa\Microsoft Visual Studio 10.0\Common7\Tools;^
c:\WINDOWS\Microsoft.NET\Framework\v4.0.30319;^
c:\WINDOWS\Microsoft.NET\Framework\v3.5;^
c:\Archivos de programa\Microsoft Visual Studio 10.0\VC\UCPackages;^
C:\Archivos de programa\HTML Help Workshop;^
C:\Archivos de programa\Microsoft SDKs\Windows\v7.0A\bin\NETFX 4.0 Tools;^
C:\Archivos de programa\Microsoft SDKs\Windows\v7.0A\bin;^
C:\WINDOWS\system32;^
C:\WINDOWS
C:\spec>
```

Prueba que puedes compilar un benchmark utilizando la configuración por defecto y ejecútalo:

```

C:\WINDOWS\system32\cmd.exe
C:\spec>runspec --action=build --tune=base --config=win32-x86-vc7.cfg gzip
runspec v1585 - Copyright 1999-2005 Standard Performance Evaluation Corporation
Loading standard modules.....
Loading runspec modules.....
Identifying output formats...asc...config...html...pdf...ps...raw...
We will use: 164.gzip
Compiling Binaries
  Building 164.gzip ref base vc7.x87.exe default
Build Complete
C:\spec>runspec --size=test --iterations=1 --noreportable --config=win32-x86-vc7
.cfg gzip
runspec v1585 - Copyright 1999-2005 Standard Performance Evaluation Corporation
Loading standard modules.....
Loading runspec modules.....
Identifying output formats...asc...config...html...pdf...ps...raw...
We will use: 164.gzip
Compiling Binaries
  Up to date 164.gzip test base vc7.x87.exe default
Setting Up Run Directories
  Setting up 164.gzip test base vc7.x87.exe default: created
Running Benchmarks
  Running 164.gzip test base vc7.x87.exe default
Success: 1x164.gzip
Producing Reports
mach: default
  ext: vc7.x87.exe
  size: test
  set: int
    format: raw -> C:/spec/result/CINT2000.005.raw
    format: ASCII -> C:/spec/result/CINT2000.005.asc
  set: fp
runspec finished
C:\spec>

```

II. EJERCICIO INDIVIDUAL

Analiza la documentación relativa al conjunto de benchmarks SPEC CPU2000 que encontrarás en <http://www.spec.org/cpu2000/docs/>

Analiza las tablas de medidas correspondientes al benchmark SPEC CPU publicadas periódicamente por la organización SPEC. Comprueba qué tipo de CPU tienen los ordenadores del laboratorio y busca los benchmark SPEC para ese procesador. Si no existe ese tipo concreto de CPU busca el que te parezca más parecido y analiza los resultados que da SPEC.

Una vez encontrado y chequeadas las medidas de velocidad de base y de pico, visualiza el fichero de resultados correspondiente (p.e. en formato HTML) mediante el enlace habilitado para ello. Observar detalladamente los resultados de cada uno de los programas y analizar los diferentes factores que han influido en los resultados obtenidos en dicha máquina, tales como frecuencia del procesador, memoria instalada, compilador utilizado, optimizaciones, etc.

Antes de ejecutar los benchmark de SPEC con el siguiente comando, piensa y justifica por escrito qué ocurrirá y qué resultados esperas:

```
runspec --reportable --config=win32-x86-vc7.cfg -T base int
```

Contesta a las siguientes preguntas:

¿Para qué sirve cada una de las opciones empleadas en el comando?

¿La ejecución de los benchmark ha dado los resultados que esperabas?, ¿por qué?

```
C:\WINDOWS\system32\cmd.exe - runspec --reportable --config=win32-x86-vc7.cfg -T bas...
C:\spec>runspec --reportable --config=win32-x86-vc7.cfg -T base int
runspec v1585 - Copyright 1999-2005 Standard Performance Evaluation Corporation
Loading standard modules.....
Loading runspec modules.....
Identifying output formats...asc...config...html...pdf...ps...raw...
We will use: 164.gzip, 175.vpr, 176.gcc, 181.mcf, 186.crafty, 197.parser, 252.eon,
253.perlbmk, 254.gap, 255.vortex, 256.bzip2, 300.twolf
Compiling Binaries
Building 164.gzip ref base vc7.x87.exe default
Building 175.vpr ref base vc7.x87.exe default
Building 176.gcc ref base vc7.x87.exe default
Building 181.mcf ref base vc7.x87.exe default
Building 186.crafty ref base vc7.x87.exe default
Building 197.parser ref base vc7.x87.exe default
Building 252.eon ref base vc7.x87.exe default
Building 253.perlbmk ref base vc7.x87.exe default
Error with make 'specmake build > make.out 2> make.err': check file 'C:/spec/benchspec/cint2000/253.perlbmk/run/00000001/make.err'
Error with make!
*** Error building 253.perlbmk
Building 254.gap ref base vc7.x87.exe default
Building 255.vortex ref base vc7.x87.exe default
Building 256.bzip2 ref base vc7.x87.exe default
Building 300.twolf ref base vc7.x87.exe default
Setting Up Run Directories
Setting up 164.gzip ref base vc7.x87.exe default: created
```

Por último, elabora un ranking de velocidad de cuatro modelos de computadores para SPECint2000 y SPECfp2000, utilizando los datos proporcionados en la página de SPEC y las métricas de velocidad y productividad de base. Para ello, considera modelos similares del mismo fabricante, frecuencias de procesador similares y con 2, 4, 8 y 16 núcleos respectivamente. Compara y comenta los resultados. ¿Cómo varían las métricas con el número de núcleos?, ¿Varía el ranking al pasar de enteros a flotantes?

III. DESARROLLO DE LA PRÁCTICA EN GRUPO

El objetivo de esta parte es que cada grupo evalúe el rendimiento de diferentes máquinas utilizando los benchmarks de SPEC CPU para la evaluación del procesador en al menos 6 máquinas diferentes. Para ello, copiar todo el directorio "spec" en la máquina que vayas a analizar y vuelve a ejecutar los pasos descritos en el apartado "Preparación del entorno". Analizar y comentar los resultados de manera apropiada elaborando una tabla comparativa con los datos obtenidos de la ejecución de los benchmark en las 6 máquinas y justificar los resultados.