Actividad grupal: Ejercicios sobre Prestaciones

**Objetivos**

El objetivo de esta actividad es que los estudiantes apliquen los conceptos vistos en clase sobre las prestaciones que podemos medir de los ordenadores, calculando estas prestaciones, así como definir cómo estas prestaciones podrían mejorar mediante la inclusión de nuevos componentes en los ordenadores.

**Descripción** de la actividad

Esta actividad está compuesta de tres ejercicios sobre prestaciones que se describen a continuación:

**EJERCICIO-1**

Se pretende ejecutar una aplicación en un computador cuyo procesador emplea únicamente dos tipos de instrucción:

* Instrucción 1 con CPI = 3
* Instrucción 2 con CPI = 2

Se sabe además que el procesador opera a 200 MHz.

Para generar el ejecutable del programa se pueden emplear dos compiladores diferentes, A y B, que ofrecen dos interpretaciones distintas del código, como indica la tabla adjunta.

|  |  |  |
| --- | --- | --- |
| **Compilador** | **Instrucción 1** | **Instrucción 2** |
| A | 6 | 2 |
| B | 4 | 6 |

Se pide:

* Obtener los MIPS necesarios para ejecutar ambas opciones en la máquina indicada.
* Teniendo en cuenta el tiempo de CPU, compara y comenta los resultados obtenidos en el apartado anterior.

**SOLUCIÓN**

* Obtener los MIPS necesarios para ejecutar ambas opciones en la máquina indicada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  | = |  |  |

|  |  |
| --- | --- |
| **FORMULAS** | |
| **CPI** | **MIPS** |
|  |  |

Donde:

**CPI:** Son el número de ciclos por instrucciones que queremos calcular.

**I:** Son las instrucciones de cada instrucción.

**NI:** Son el número de instrucciones total por programa.

**MIPS:** Son los millones de instrucciones por segundo que queremos calcular

**Fcpu:** Es la frecuencia, en Herzios, a la que trabaja el procesador.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

* Teniendo en cuenta el tiempo de CPU, compara y comenta los resultados obtenidos en el apartado anterior.

Comenzamos calculando los tiempos para el compilador *A* y *B*.

|  |
| --- |
| **FORMULA** |
|  |

Donde:

**Tcpu:** Es el tiempo de CPU que querremos calcular.

**NI:** Son el número de instrucciones total por programa.

**CPI:** Son el número de ciclos por instrucciones que queremos calcular.

**Fcpu:** Es la frecuencia, en Herzios, a la que trabaja el procesador.

|  |  |
| --- | --- |
| **Compilador A** | **Compilador B** |
|  |  |
|  |  |

Comparativa de resultado y conclusiones:

|  |  |  |  |
| --- | --- | --- | --- |
| **Compilador** | **MIPS** | **NI** | **Tcpu** |
| A | 72.73 | 8 |  |
| B | 83.33 | 10 |  |

Por lo tanto, el tiempo de ejecución de B es mayor que el de A, porque el compilador de B realiza mas instrucciones por segundo () y esto significa que el es mas potente.

**EJERCICIO-2**

Durante el desarrollo de una CPU se evalúa su rendimiento mediante un benchmark y, entre otros resultados, se obtiene la siguiente distribución del tiempo de ejecución:

|  |  |
| --- | --- |
| **Instrucción** | % |
| **Multiplicación** | 20 |
| **Acceso a memoria** | 50 |
| **Resto** | 30 |

En función de estos resultados se estudia introducir dos opciones de mejora en la configuración:

* Opción 1: Acelerar en 4 veces la ejecución de multiplicaciones.
* Opción 2: Mejorar el acceso a operandos en memoria, reduciendo a la mitad el tiempo necesario para ello.

Se pide:

* ¿Cuál es la aceleración si solo se aplica la opción 1?
* ¿Y si solo se aplica la opción 2?
* Evalúa de nuevo la aceleración si se aplican las dos opciones planteadas.

**SOLUCIÓN**

Considerando:

la aceleración.

, el tiempo sin mejora.

, el tiempo con mejora **.**

, la fracción de código que no se aprovecha de la mejora (, el que aprovecha la mejora).

|  |  |
| --- | --- |
| **FORMULA** | |
|  |  |

* ¿Cuál es la aceleración si solo se aplica la opción 1?

|  |
| --- |
| **Multiplicación (0.2)** |
|  |

* ¿Y si solo se aplica la opción 2?

|  |
| --- |
| **Acceso Memoria con m/2** |
|  |

* Evalúa de nuevo la aceleración si se aplican las dos opciones planteadas.

|  |
| --- |
| **FORMULA** |
|  |

**EJERCICIO-3**

Suponga que en un programa la frecuencia de fallos de la cache de instrucciones es el 2%, y la frecuencia de fallos de la cache de datos es el 4%. Si un procesador tiene un CPI igual a 2 sin incluir paradas debidas a la memoria, y la penalización por fallo es de 100 ciclos para todo tipo de fallo, determinar cuánto más rápido es un procesador cuya cache es perfecta, es decir, que nunca falla. Suponga que la frecuencia de las instrucciones de carga y almacenamiento es el 36%

**SOLUCIÓN**

|  |
| --- |
| **FORMULA** |
|  |

Datos:

Frecuencia de fallos de la cache de instrucciones

Frecuencia de fallos de la cache de datos

El procesador tiene un CPI igual a

Operaciones debido a las Instrucciones:

Operaciones debido a la Memoria:

El numero total de ciclos de parada por acceso a memoria es de

Esto quiere decir que hace 3 paradas por instrucción.

El procesador tiene un CPI igual a 2 sin incluir paradas debidas a la memoria. Tenemos que sumar al número total de ciclos de parada este CPI.

A continuación, hacemos una relación entre la CPI con paradas y la CPI perfecta, haciendo uso de la formula inicial:

Las prestaciones con la cache perfecta son mejores en un **2.72**

**Extensión:** Máximo 8 páginas.

**Rúbrica**

|  |  |  |  |
| --- | --- | --- | --- |
| Simulaciones en el WinDLXV | Descripción | Puntuación máxima  (puntos) | Peso  % |
| Desarrollo Ejercicio 1 | El ejercicio está desarrollado correctamente y está correctamente descrito. | 3 | 30% |
| Desarrollo Ejercicio 2 | El ejercicio está desarrollado correctamente y está correctamente descrito. | 3 | 30% |
| Desarrollo Ejercicio 3 | El ejercicio está desarrollado correctamente y está correctamente descrito. | 3 | 30% |
| Calidad del reporte | El documento está bien estructurado, carece de faltas de ortografía y sigue las normas APA | 1 | 10% |
|  |  | **10** | **100 %** |