|  |  |  |  |
| --- | --- | --- | --- |
| Instructor |  | Due Date |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Part | **1** | **2** | **3** | **4** | Total |
| *Maximum Points* | **25** points | **25** points | **25** points | **25** points | **100**G101010 pointsG |
| ***Your Score*** |  |  |  |  |  |

**Textbook Reading Assignment**

Thoroughly read Chapter(s) 11 in your Computer Architecture and Organization textbook.

**Part 1 Glossary Terms - Performance Measurement and Analysis**

Define, in detail, each of these glossary terms from the realm of computer architecture and computer topics, in general. If applicable, use examples to support your definitions. Consult your notes

or course textbook(s) as references or the Internet by visiting Web sites such as:

[**http://www.bing.com**](http://www.bing.com) or [**http://www.webopedia.com**](http://www.webopedia.com/)

**(a) Delayed Branching**

|  |
| --- |
| Delayed branching is a program / branching optimization technique wherein the sequence of instructions around a branch are re-ordered in such a way that a wasted instruction slot is filled with another instruction that would be executed anyway. This optimization takes place at compile time. “The idea is to utilize otherwise wasted cycles following a branch,” (Null et al. 637).  E.g., if I had an instruction that came after an unlinked branch statement that does not return, and which therefore was fetched after the branch statement but never executed, I could replace it with the instruction statement that came directly before the branch statement, and thereby utilize that unused fetch-decode-execute cycle. |

**(b) Elevator Algorithm**

|  |
| --- |
| To avoid the starvation (the constant delaying of accessing a remote track on a disk in favor of a track closer to the arm) that may occur in a Shortest Seek Time First (SSTF) disk scheduling algorithm, the elevator algorithm (so named after how elevators are scheduled in a skyscraper) can be used instead.  The elevator algorithm has the arm continually sweep over the disk, stopping when it reaches a track that is in its service queue (Null et al. 648), much like how elevators constantly go up and down during peak service times in a skyscraper. |

**(c) FLOPS**

|  |
| --- |
| Floating Point Operations Per Second (FLOPS) is an acronym for a metric used to measure computer system performance. |

**(d) Loop Peeling**

|  |
| --- |
| Another program optimization technique that takes place at compile time, Loop Peeling removes the beginning or ending statements from a loop to remove branching and lead to increased instruction-level parallelism (Null et al. 643) |

**(e) SSTF**

|  |
| --- |
| A disk scheduling algorithm, Shortest Seek Time First (SSTF) schedules tracks that are closest to the disk arm with higher priority in a service queue. |

**Part 2 Exercises - Performance Measurement and Analysis**

For each of the following, enter True or False.

**TRUE (1)** Mathematical and statistical tools give us many ways to rate the overall performance of a system and its components.

**FALSE (2)** A CPU running at double the clock speed of another is likely to give better CPU throughput.

**FALSE (3)** Branch prediction is the process of attempting to guess the next instruction in the instruction stream.

**TRUE (4)** Disk scheduling can be a function of either the disk controller or the host operating system.

**TRUE (5)** The elevator algorithm works much similar to how skyscraper elevators service their passengers.

**TRUE (6)** When throughput is more important than reliability, a system may employ the write - back cache policy.

**TRUE (7)** More time - consuming programs have greater influence on the harmonic mean.

**TRUE (8)** Clock speed, MIPS and FLOPS are the metrics in comparing relative performance across a line of similar computers offered by the same vendor.

**TRUE (9)** Simulation is very useful for estimating the performance of systems or system configurations that only exist.

**TRUE (10)** Disk utilization, the measure of the percentage of time that the disk is busy servicing I / O requests.

**Part 3 Exercises - Performance Measurement and Analysis**

**(1)** **( Mean Values )**

Some types of averages include the arithmetic mean, the geometric mean and the harmonic mean.

The arithmetic mean is defined as the sum of the data values divided by their count.

The geometric mean of *n* positive numbers is the *n* - th root of their product.

The harmonic mean of *n* numbers is the reciprocal of the sum of the reciprocals of the numbers.

The execution times for a system running four benchmarks is shown in the table below. Compute the arithmetic, geometric and harmonic means of this data.

Show your computations.

|  |  |
| --- | --- |
| *program* | System A Execution Time |
| W | 60 |
| X | 85 |
| Y | 70 |
| Z | 90 |

|  |
| --- |
| **Arithmetic Mean:** (60 + 85 + 70 + 90) / 4 = 76.25  **Geometric Mean:** (60 x 85 x 70 x 90) ^ (1/4) = 75.29  **Harmonic Mean:** 4 / ((1/60) + (1/85) + (1/70) + (1/90)) = 74.31 |

**(2)** **( Benchmarks and the Geometric Mean )**

The execution times for three systems running five benchmarks are shown in the table below. Compare the relative performance of each of these systems ( i.e., A to B , B to C

and A to C ) using the arithmetic and geometric means. Are there any surprises? Explain your analysis.

|  |  |  |  |
| --- | --- | --- | --- |
| *program* | System A  Execution Time | System B  Execution Time | System C Execution Time |
| V | 150 | 200 | 80 |
| W | 200 | 250 | 150 |
| X | 275 | 170 | 200 |
| Y | 400 | 750 | 500 |
| Z | 900 | 1100 | 1200 |

|  |
| --- |
| **A vs. B:**   * **Arithmetic: 385 vs. 494** * **Geometric: 312 vs. 370**   **A vs. C:**   * **Arithmetic: 385 vs. 426** * **Geometric: 312 vs. 270**   **B vs. C:**   * **Arithmetic: 494 vs. 426** * **Geometric: 370 vs. 270**   **Analysis:** My first surprise is that the large difference between geometric and arithmetic means. My second surprise is that while A is faster than C when doing an arithmetic mean, it is slower than C when doing a geometric mean. I think because we have a great deal of variance between data, a geometric mean would be more accurate, and is said to be more accurate when comparing systems in Null et al. 618. |

**(3)** **( Benchmarks and the Geometric Mean )**

The execution times for three systems running five benchmarks are shown in the table below. Compare the relative performance of each of these systems ( i.e., A to B , B to C

and A to C ) using the arithmetic and geometric means. Are there any surprises? Explain your analysis.

|  |  |  |  |
| --- | --- | --- | --- |
| *program* | System A  Execution Time | System B  Execution Time | System B  Execution Time |
| V | 40 | 100 | 70 |
| W | 325 | 275 | 350 |
| X | 275 | 100 | 300 |
| Y | 400 | 200 | 400 |
| Z | 800 | 1000 | 600 |

|  |
| --- |
| **A vs. B:**   * **Arithmetic: 368 vs. 335** * **Geometric: 258 vs. 222**   **A vs. C:**   * **Arithmetic: 368 vs. 344** * **Geometric: 258 vs. 281**   **B vs. C:**   * **Arithmetic: 335 vs. 344** * **Geometric: 222 vs. 281**   **Analysis:** Data is a bit less variable than the previous exercise, and arithmetic and geometric means are closer in similarity. However, System A surpasses system C in performance arithmetically, but not geometrically. |

**(4)** **( Synthetic versus Real - World Benchmarks )**

Comment on any effectiveness of synthetic benchmarking versus real - world benchmarking.

|  |
| --- |
| Synthetic benchmarking uses tools (programs) to measure the performance of a system using metrics that are not dependent on a particular system or its ISA, and therefore can be used to compare different systems more accurately. |

**(5)** **( Performance Factors )**

Discuss some of the factors that affect the performance of processors, programs and magnetic disk storage.

|  |
| --- |
| Various factors affect the performance of systems, including how programs are compiled and optimized, what algorithms are in place for managing disk queues and caching accessed data, and whether a disk has been defragmented (re-organized) lately. |

**Part 4 Exercises - Performance Measurement and Analysis**

Write a complete answer for each of these.

**(1) ( Benchmarks )**

What would you say to a vendor that tells you that his system runs 50 % of the SPEC benchmark kernel programs twice as fast as the leading competitive system? Which statistical fallacy is at work here?

|  |
| --- |
| I would ask whether their competitors run the other 50% of the SPEC kernel programs twice as fast as well.  The vendor is cherry-picking their benchmark data. |

**(2) ( Synthetic Benchmarks )**

What are the limitations of synthetic benchmarks such as Whetstone and Dhrystone?   
 Do you think that the concept of a synthetic benchmark could be extended do overcome these limitations? Explain your answer.

|  |
| --- |
| Whetstone and Dhrystone are simple enough to be duped by manufacturers, who can equip their products with compilation switches that will output special code that is optimized for these benchmarks. Manufacturers will do whatever they can to make their numbers look good, so called “bench-marketing” (Null et al. 627-628).  I suppose a synthetic benchmark program could be extended to include both the Whetstone and Dhrystone libraries and run each within the same program, or perhaps attempt to obfuscate the benchmark, or implement some other method to try to detect and dupe a “compilation switch.” |

**(3) ( The Retail Business Sector )**

Besides the retail business sector, what other organizations would need good performance from a transaction - processing system. Justify your answer.

|  |
| --- |
| Banking / financial systems, since there would be many transactions that would need to be processed. |

**(4) ( Comparing Computer Models )**

Suppose a friend has asked you to help him to make a choice as to what kind of computer he should buy for his personal use at home. What would you look for in comparing various makes and models? How does your line of thinking differ in this situation than if you were to help your employer purchase a Web server to accept customers’ orders over the Internet?

|  |
| --- |
| I would look for hardware specs as well as benchmark test data for each situation, and consult different forums (Reddit etc.). |

**(5) ( Branching Predictors )**

In reference to branching, static prediction and fixed prediction are the same. Explain why this is so.

|  |
| --- |
| Static and fixed predications are the same in that they both involve black box testing of a branch, and their predictions will both be the same (either the branch is taken or not). They differ from dynamic prediction which looks at branch history to make a prediction. |