Homework 1 (5 points)
Due 9/12 by 2 pm

1. (1 point) We are considering an enhancement to a processor that is used for encryption/decryption by adding special cryptographic hardware to it. When an encryption operation is run on the cryptographic hardware it is 20 times faster than running it on the normal hardware. A decryption operation is 10 times faster on the cryptographic hardware than on the normal hardware. Let us call the percentage of time spent doing encryption on normal hardware $e$, and percentage of time spent doing decryption on normal hardware $d$.
   a. (0.25 points) Draw a graph that plots the overall speedup when cryptographic hardware is introduced on $y$-axis versus the percentage of time spent doing encryption $e$ on $x$-axis. Hint: start by writing a formula for overall speedup and how it depends on $e$.
   b. (0.25 points) Draw a similar graph that shows how the overall speedup depends on $d$.
   c. (0.25 points) What percentage of time is spent doing decryption after cryptographic hardware is installed, if the overall speedup was 2.
   d. (0.25 points) What is the maximum overall speedup we can achieve, and what type of workload would give us such speedup? How big should $e$ be for the maximum speedup? How about $d$?

2. (1 point) Assume that we can improve the speed of multiplication 40 times. After the improvement, CPU running a typical workload spends 25% of time doing multiplication.
   a. (0.5 points) Calculate the overall speedup
   b. (0.5 points) What percentage of time was spent in multiplication before we introduced the improvement?

3. (2 points) The table below shows the time in seconds spent by three machines running the programs listed in the first column of the table. Compare the performance of these machines using:
   a. (0.25 point) Arithmetic mean
   b. (0.5 points) Weighted arithmetic mean with machine A being the reference machine
   c. (0.5 points) Weighted arithmetic mean with machine B being the reference machine
   d. (0.5 points) Geometric mean with machine A being the reference
   e. (0.25 points) Discuss advantages and disadvantages of arithmetic, weighted arithmetic and geometric means for performance measurement.

Note: To receive full credit for parts a-d, you must:
• Show formulas for all the calculations (you needn’t show formulas that are identical in everything except the index value, e.g. it is enough to show a formula for one weight)
• Provide a numerical value for performance of A, B, C and if there are weights involved show values of appropriate weights
• Rank the performance of machines A, B, C in a decreasing order
• Provide a conclusion in a form “Machine 1 is X times faster than machine 2, and machine 2 is Y times faster than machine 3”, where machines 1-3 are the first, second and third ranked machines.

<table>
<thead>
<tr>
<th>Program</th>
<th>Machine A</th>
<th>Machine B</th>
<th>Machine C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>45</td>
<td>37</td>
<td>33</td>
</tr>
<tr>
<td>2</td>
<td>109</td>
<td>267</td>
<td>406</td>
</tr>
<tr>
<td>3</td>
<td>5215</td>
<td>2874</td>
<td>2646</td>
</tr>
<tr>
<td>4</td>
<td>386</td>
<td>311</td>
<td>244</td>
</tr>
<tr>
<td>5</td>
<td>522</td>
<td>273</td>
<td>515</td>
</tr>
<tr>
<td>6</td>
<td>495</td>
<td>387</td>
<td>295</td>
</tr>
<tr>
<td>7</td>
<td>1838</td>
<td>924</td>
<td>379</td>
</tr>
</tbody>
</table>

4. (1 points) A machine runs an instruction mix with 25% instructions being ADD, 20% MUL, 10% DIV and the rest are MOV instructions. The CPI for ADD instructions is 2, for MUL it is 7, for DIV 13 and for MOV 3. There is an optimization that can speed up DIV operation on floating point numbers 10 times. About 30% of DIV instructions are performed on floating point numbers.

a. (0.25 points) Calculate the overall speedup if the optimization is implemented. (Hint: calculate the current and the new CPI and compare them)

b. (0.25 points) What percentage of time is spent doing division when the optimization is implemented? What percentage of time was spent without the optimization?

c. (0.5 points) In the original architecture (without the DIV optimization) identify the instruction that is executed most of the time (this may or may not be the most frequent instruction in the code). Imagine that we can design hardware that can speed up all instructions of this kind. How much would the instruction’s speedup have to be so that the overall speedup matches the speedup achieved in part a. with DIV optimization?