1. Caches
   a) The average memory access time (AMAT) can be modeled using the following formula:

   \[ AMAT = Hit\ time + Miss\ rate \times Miss\ penalty \]

   Name and explain (briefly) one technique for each of the three components of the formula in order to decrease the average memory access time.

   Pick one of each block:

   Reducing Miss penalty:
   - Multilevel caches: 1st level small, but at the speed of the CPU, 2nd level larger but slower
   - Critical word first: don’t wait until the entire cache-block has been load, focus
   - on the required data item: ask for the required data item, Forward the data item to the processor, Fill up the rest of the cache block afterwards
   - Early restart: don’t wait until the entire cache-block has been load, focus
   - on the required data item: Fetch words of a cache block in normal order, Forward the requested data item to the processor as soon as available, Fill up the rest of the cache block afterwards
   - Giving priority to read misses over writes
   - Merging write buffer: Check in the write buffer whether multiple entries can be merged to a single one
   - Victim caches: fully associative cache between the ‘real’ cache and the memory keeping blocks that have been discarded from the cache
   - Nonblocking caches
   - Hardware prefetch of Instructions and Data
   - Compiler controlled prefetching

   Reducing Miss rate:
   - Larger cache block size
   - Larger caches
   - Higher associativity
   - Way prediction and pseudo-associative caches
   - Compiler optimization

   Reducing Hit time:
   - Small and simple caches
   - Avoiding address translation
   - Pipelined cache access
   - Trace caches

(3 Pts)
b) Consider two identical machines differing only in the cache organization. The first machine has a 2-way set-associative cache, the second machine has a 4-way set-associative cache. The 1st machine has a clock cycle time of 1.25ns and a miss rate of 1.0%, while the 2nd machine has a clock cycle time of 1.4ns. Assuming a CPI of 2.0 for perfect cache behavior, 1.5 memory references per instruction, a cache miss penalty of 75ns, and a cache hit time of 1 clock cycle, determine the miss rate of the second machine in order for the four-way set associative cache to have a lower average memory access time than the first machine.

\[ \text{AMAT}_1 = 1.25 + (0.01 \times 75) = 2.0\text{ns} \]
\[ \text{AMAT}_2 = 1.4 + (x \times 75) \]

\[ \text{AMAT}_2 < \text{AMAT}_1 \text{ if } 1.4 + 75x < 2.0 \]
\[ \rightarrow x < (2.0-1.4)/75 = 0.008 = 0.8\% \]

c) In the following, we would like to determine the set associativity leading to the minimal average memory access time. The miss penalty is still 75ns. A cache hit takes one clock cycle. However, the clock cycle time is depending on the set associativity \( n \), with \( n=1 \) being a direct mapped cache, \( n=2 \) being a 2-way set associative cache etc. The formula describing the dependence between the clock cycle time and the set associativity is:

\[ \text{Clock cycle time}(n) = 1.0 + 0.02n^2 \]

Similarly, the Miss rate depends on \( n \) and can be described by the formula

\[ \text{Miss rate} = 0.01 - 0.002n \]

1. Give the formula for the overall Average Memory Access Time (AMAT) depending on \( n \).

\[ \text{AMAT}(n) = 1.0 + 0.02n^2 + (0.01 - 0.002n) \times 75 \]
\[ = 1.0 + 0.02n^2 + 0.75 - 0.15n \]
\[ = 0.02n^2 - 0.15n + 1.75 \]
2. Determine the value of $n$ leading the minimal AMAT. Please note that you will have to round $n$ to closest integer value.

$$AMAT'(n) = 0.04n - 0.15$$
$$0 = 0.04n - 0.15$$
$$0.04n = 0.15$$
$$n = 0.15/0.04 = 3.75 \approx 4 \text{ (rounded)}$$

Thus, $n = 4$ would lead to the minimal AMAT.

2) Given the following code sequence to implement a matrix-vector multiply operation:

```c
for (i=0; i<96; i++) {
    for (j=0; j<96; j++) {
        c[i] = c[i] + A[i][j] * b[j];
    }
}
```

Determine the number of cache misses and the cache miss rate for the code sequence above, assuming that
- all data items are 8-byte double precision floating point numbers,
- C-style row major ordering for multi-dimensional matrices
- no limitation on the cache size,
- each cache line is 64 bytes,
- no data item of A, b and c are in the cache upon start of the execution of the code sequence above
- the array c has been initialized to zero much earlier in the code and has subsequently been discarded from the cache
- we only worry about compulsory misses, not about capacity or conflict misses.

Note: please give some details for your calculations in order to get partial points in case of a mistake.

e.g. for the variable c: the first access to c[0] will be a cache miss. The according load operation will load a cache line consisting of 64 bytes = 8 elements in to the cache, such that access to c[1]...c[7] will be cache hits. c[8] will lead to cache miss again. Thus, we have

for c: 12 cache misses
COSC 6385 Computer Architecture

Exercises

Name:_______________________________________

for b: 12 cache misses:
for A: 96*12 cache misses
total: 98*12 cache misses = 1176

Total number of cache accesses:
for each iteration of the inner loop j we have 3 reads (c[i], b[j], A[i][j])
and one write (c[i]) = 4 memory accesses
Thus, total number of accesses: 96*96*4 = 36864

Cache miss rate = No. of cache misses / No. of cache access = 1176 / 36864 = 0.0319 ~ 3.19%

3) Magnetic hard drives are accounting for the largest number of failures in today’s
computer systems. In order to increase the reliability of hard drives, servers
typically use hard drives in a RAID 5 configuration, where four hard drives are
used to store the data, and a fifth hard drive is used to store parity bits. These
parity bits can be used to reconstruct the data of a failed hard drive. (Note, that a
RAID 5 configuration cannot tolerate the failure of two hard drives
simultaneously). For the following calculations, please assume the MTTF of each
individual hard drive is 250,000 hours

a. For the first step of the reliability calculation, let’s assume that it takes on
average 50 hours to get the replacement hard drive from the manufacturer of
the server. Assuming that MTTR is 50h, how large is the MTFF of a RAID 5
configuration consisting of five hard drives in which the disk containing the
parity bits is considered to be a ‘backup’ to any of the other four hard drives?

MTTF hard drive = 250,000h = 2.5 * 10^5h
MTTR = 50h

FIT w/o = 5/(2.5 * 10^5) => MTTF w/o = 5*10^4h
MTTF w/ = MTTF w/o / (MTTR/MTTF 4 hard drives)

FIT 4 hard drives = 4/(2.5 * 10^5) => MTTF 4 hard drives = 6.25 * 10^4

MTTF w/ = 5*10^4h / (50 / 6.25 * 10^4)
= 6.25 * 10^1h
b. In case a hard drive fails, data of the failed hard drive has to be reconstructed using the parity information bit-by-bit. Therefore, the MTTR depends not only on the time it takes to deliver the replacement hard drive, but also on the data reconstruction time. Thus, an improved formula for MTTR involves both components and depends on the size of the hard drives which is given in GB, and can be described by the formula

\[ \text{MTTR}(s) = 50 \text{ hours} + 0.1 \text{ hour} \times s \]

(i.e. it takes 0.1 hours to reconstruct 1 GB of data.). Using the new, improved formula for MTTR, determine the according MTTF as a function of the size of the hard drive s.

\[
\text{MTTF}_{w/}(s) = \frac{\text{MTTF}_{w/o}(s)}{(\frac{\text{MTTR}(s)}{\text{MTTF}_{4\text{hard drives}}})}
\]

\[
= \frac{5 \times 10^4 h}{((50 + 0.1 \times s) / (6.25 \times 10^4))}
\]

\[
= \frac{3.125 \times 10^9}{50 + 0.1s} h
\]

c. What is the formula for the MTTF(s) of a computer system having two such RAID 5 configurations using five hard drives each as a function of the disk capacity s? Note that the two RAID 5 systems are assumed to be completely independent sets of disks, and are not backing up each other.

\[ \text{FIT} = \frac{50 + 0.1s}{3.125 \times 10^9} \times 2 \]

\[ \text{MTTF}_{2\text{RAID}} = \frac{3.125 \times 10^9}{2(50 + 0.1s)} \]
d. What is the formula for the MTTF(s,k) of a computer system having k independent RAID 5 disk systems with five disks each? (1 Pts)

\[
MTTF_{k,RAID5} = \frac{3.125 \times 10^9}{k(50 + 0.1s)}
\]


e. In order to provide a customer a 10 Terabyte storage solution, a company has the option to use either 250 GB disks, which would require 10 RAID 5 systems utilizing five disks each. Alternatively, one could use 500GB disks and have 5 RAID 5 configurations with five disks each. Which one of the two systems (system 1: k=10, s=250GB; system 2: k=5, s=500GB) promises a higher MTTF according to the formula developed in part d? (2 Pts)

System 1: k = 10, s= 250GB

\[
MTTF_{\text{system1}} = \frac{3.125 \times 10^9}{10(50+0.1 \times 250)} = \frac{3.125 \times 10^9}{(500+250)} = \frac{3.125 \times 10^9}{750}
\]

\[= 4.16666 \times 10^6 \text{h}\]

System 2: k= 5, s = 500GB

\[
MTTF_{\text{system2}} = \frac{3.125 \times 10^9}{5(50+0.1 \times 500)} = \frac{3.125 \times 10^9}{(250+250)} = \frac{3.125 \times 10^9}{500}
\]

\[= 6.25 \times 10^6 \text{h}\]

⇒ the second system has a higher MTTF and should be preferred.