1. A memory hierarchy includes two levels of cache, a main memory, a disk and a TLB for virtual to physical address translation. The level 1 (L1) instruction and data cache units have a hit time of $\text{Hit\_Time\_L1} = 1$ (CPU) clock cycle, the unified L2 cache has a hit time of $\text{Hit\_Time\_L2} = 8$ clock cycles, the TLB has a hit time of $\text{Hit\_Time\_TLB} = 1$ clock cycle, and the miss penalty from L2 to memory is $\text{MissPenalty\_L2} = 96$ clock cycles. Both L1 units have a Write Through policy and use write buffers. During the execution of a certain benchmark it has been observed that on the average, for every 100 memory references made by the CPU, 75 were directed to the instruction cache and have experienced $n_I=1.5$ misses, and the remaining 25 were directed to the data cache and have experienced $n_D=1$ miss. Out of 100 accesses to L2 there were (on average) $n_L2=3$ misses. (a) What is the % of Load and Store instructions that were executed?
(b) Calculate the miss rates of the instruction and data cache units, \( m_I \) and \( m_D \), respectively.

(c) Calculate the local and global miss rates of the L2 cache, \( m_{L2}^l \) and \( m_{L2}^g \), respectively.

(d) What is the average memory access time (AMAT) during the execution of the benchmark? Assume that there are no page faults and the TLB miss rate is \( m_{TLB}=0 \).

(e) What is the CPI of the processor if the base CPI (assuming L1 caches with a 100% hit ratio) is \( CPI_{base}=2.6 \)?

(f) To reduce the AMAT two design modifications were considered: (1) Changing the organization of L2 from direct mapped to 2-way set associative resulting in a higher hit time of \( Hit\_Time_{L2}=9 \) clock cycles but a lower average number of misses in L2 of \( n_{L2}=2.5 \) (out of 100 accesses to L2), or (2) Splitting the unified L2 into two equal-sized (and direct mapped) Instruction L2 cache and Data L2 cache resulting in a lower hit time of \( Hit\_Time_{L2}=7 \) clock cycles but a higher average number of misses in L2 of \( n_{L2}=4.0 \) (out of 100 accesses to L2). Explain the reasons for these increases/reductions in the hit time and number of misses.

(g) (Bonus) Calculate the new AMAT for the two alternatives in (f). Which one of these design modification would you recommend?
2. A transaction processing system includes a large number of identical disks. Each disk has a capacity of 2 Tera Bytes ($10^{12}$ bytes), an average seek time of 3.3msec, rotates at 10,000 RPM, transfers data at a rate of 90MByte/sec and experiences a 0.1msec controller delay.

(a) Calculate the time needed to read 30KBytes assuming that the data resides on 60 contiguous sectors (0.5 KBytes each) within one track.

(b) Repeat (a) for the case where the first 30 sectors are on one track while the remaining 30 sectors are in contiguous positions along a second track within the same cylinder.

(c) The system also includes a 1 GHz CPU, 2GByte memory and two IOPs with each controlling 1 disk of the above type. Each transaction starts with reading a 30KByte file (along one track) from the disk controlled by IOP1, continues with processing in the CPU that generates a 15KByte output file that is then written onto the other disk (controlled by IOP2) along one track. The input, processing and output steps for one transaction cannot overlap. Processing a transaction takes approximately 720,000 instructions, requires 250 MBytes in memory, and the CPI experienced by the CPU is 2.6. Calculate $t_c$, the time needed to process the transaction by the CPU once its input data is in memory.

(d) If consecutive transactions are overlapped would the system be CPU or I/O bound? Explain.
(e) Calculate the maximum rate of transactions per second that the system can support. Is this rate sustainable? Explain.

(f) What would be the average response time of the system to incoming transactions if their rate is 100 transactions per second?

(g) (Bonus) Each of the two IOPs is now connected to two disks (of the above type) and the incoming transactions use one of the disks randomly. By how much can the system overall rate be increased (from 100) and still keep the response time at no more than 10msec? What percentage of CPU time will then be available for other programs?

---

3. The instruction mix and average number of clock cycles per instruction for a certain benchmark executing on a given processor are shown below. (Note: a cycle count of 2 cycles, for example, means that the next instruction will be stalled, on the average, by 1 cycle.) (a) Calculate the average CPI (cycles per instruction) for the above benchmark suite.

<table>
<thead>
<tr>
<th>Instruction type</th>
<th>Integer ALU ops.</th>
<th>Load</th>
<th>Store</th>
<th>Branch</th>
<th>FP Add/Sub</th>
<th>FP Mulit</th>
<th>FP Div</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>18%</td>
<td>20%</td>
<td>10%</td>
<td>12%</td>
<td>20%</td>
<td>12%</td>
<td>8%</td>
</tr>
<tr>
<td>Clock cycle count</td>
<td>1.3</td>
<td>2.5</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>20</td>
</tr>
</tbody>
</table>

(b) If the processor has a data cache and instruction cache, both with hit time of 1 cycle and miss penalty of 30 cycles. Estimate the miss rates of the instruction cache (m_I) and of the data cache (m_D). Clearly state your assumptions.
(c) Analysis of the benchmark has revealed that there is a high percentage of FP divisions with the constant divisor \(\pi\). Such an instruction, e.g., \(\text{divf F0,F1,F2}\) where \(F2\) contains \(\pi\), can be replaced by an FP multiply, e.g., \(\text{multf F0,F1,F9}\), where \(F9\) contains the reciprocal \(1/\pi\). What would be the CPI of the modified design if the compiler is successful in replacing 50% of the FP divides executed by the benchmark by FP multiplies? (Ignore the time needed to calculate once \(1/\pi\).)

(d) Repeat (b) for the case where the compiler is unable to determine in advance whether a \(\text{divf}\) can be replaced by a \(\text{multf}\). Instead, the compiler replaces each \(\text{divf F0,F1,F2}\) by a \(\text{subf F0,F2,F8}\) (where \(F8\) contains \(\pi\)) and a branch followed by either \(\text{divf F0,F1,F2}\) or \(\text{multf F0,F1,F9}\) (where \(F9\) contains \(1/\pi\)). Assume that the \(\text{multf}\) is executed 75% of the time instead of the \(\text{divf}\).

(e) Will the benchmark program execute faster with the modification in (d) than the one in (c)? What is the speedup of the faster alternative over the other one?

(f) (Bonus) Estimate the percentage of instructions that follow a Load instruction and need the loaded data. Assume an in-order pipeline with data forwarding. Clearly state your assumptions.
4. A computer system contains an IOP which may access the memory directly (DMA), and a 16-way interleaved main memory. Each memory bank has a capacity of 512K Bytes, reads (or writes) four bytes at once and has a total memory cycle of 400 nsec. (a) What is the length of the address register in the CPU assuming that the virtual address space is eight times the physical address space? 

(b) What is the bandwidth of each memory bank? 

(c) Estimate the average bandwidth of the memory system assuming that data is accessed in a random order. 

(d) The IOP collects data bytes from k I/O devices with data transfer rate of 5 bytes/µsec each and stores the data in consecutive locations in k separate buffers in the main memory. Estimate the actual memory data rate for k=8, k=20 and k=40. Explain. 

5. A certain computer system includes a CPU and two IOPs. IOP1 is connected to several disks; IOP2 is connected to a printer and several other IO devices. The CPU executes a program consisting of N steps where each step contains three non-overlapping phases: p1, p2, p3. In p1 a record of fixed length is read from a disk with a read time of t_i time units. In p2 the record is processed by the CPU for t_c time units. In this phase two output records are prepared. In p3, the first output record is sent to a disk with write time of t_{o_1} time units. The second output record is printed with print time of t_{o_2} time units. Each one of t_i, t_{o_1} and t_{o_2}, is at most 0.5 t_c, i.e., t_i, t_{o_1}, t_{o_2} ≤ 0.5 t_c 

(a) Show the timing chart of the above process and write an expression for the total time, T_N, required to execute all N steps. Assume that the size of the main memory is limited so that only one input record and its two associated output records can be stored simultaneously. A new input record can not be read before the previous two output records are disposed of.
(b) Repeat part (a) assuming that the size of the main memory is sufficient to store two input records and their associated output records.

(c) How much faster is the system in (b) than the one in (a) for $N \to \infty$? What is the maximum value of this speedup?

6. A 2 GHz processor with separate instruction and data cache has an ideal CPI of 1.6 when there are no cache misses. The application running executes 20% loads and 10% store operations. Both cache units have similar design: direct-access with a block size of 32 bytes, addressed using virtual addresses, have a hit time of 1 CPU clock cycle and use a write-back and write allocate policy. The I cache has a miss rate of 4% while the D cache has a miss rate of 8% and on the average, 32% of its blocks are “dirty.” The memory access time is 90 CPU clock cycles for the first 4 bytes, has a 4-byte memory bus and a 100% hit rate (i.e., there are no page faults). Consecutive bytes are transferred at a rate of 4 bytes per clock cycle. Assume further that there is no TLB unit, the entire page table is stored in the main memory and the virtual address is of size 32 bits. (a) Calculate $\tau_{\text{transl}}$ - the time (in CPU cycles) required to perform a virtual to physical address translation.

(b) Calculate $\tau_{\text{block}}$ - the time required to read (or write) a cache block from memory.
(c) Calculate the CPI of the processor. Write first an expression as a function of $\tau_{\text{transl}}$ and $\tau_{\text{block}}$ and only then plug in your results from (b) and (c).

(d) The memory system has been modified as follows: the cache units are now addressed using physical addresses and a TLB unit has been added to the system. This TLB unit is searched in parallel to the cache access and has a miss rate of 0.6%. All cache parameters (e.g., hit time and miss rate) remain unchanged. Calculate the CPI of the processor. Write first an expression as a function of $\tau_{\text{transl}}$ and $\tau_{\text{block}}$ and only then plug in your results from (a) and (b).

---

7. State whether each of the following statements is true or false and briefly explain your answer. A correct answer with no explanation is worth only one point. A correct answer with an incorrect explanation is worth 0 points.

(a) When allocating disk sectors for a file, it is better to allocate sectors in consecutive tracks on one surface than sectors in different surfaces.

(b) All cache organizations can benefit from a separate victim cache.

(c) Floating-point benchmarks have a higher instruction-level parallelism than integer benchmarks since the execution time of floating-point instructions is higher than that of integer instructions.
(d) A sector write operation in RAID5 requires two writes (data sector and parity sector) which can be done in parallel but will still take more time than a sector write in a non-RAID disk.

(e) A loop that includes 4 instructions (that perform some computation) and 2 loop control instructions has been unrolled 3 times, i.e., 4 iterations of the computation are now executed in a single pass through the loop. The unrolled loop has then been scheduled to execute on a 4-instruction wide VLIW processor. The resulting number of VLIW instructions will be no more than 5.

(f) A direct-access cache includes $2^n$ bytes of data and uses m-bit tags. To replace this direct-access cache by a $2^k$-way set associative cache either the tag length should increase to $m+k$ or the data portion of the cache must increase to $2^{n+k}$.

---

8. A computer system uses 20 100GB disks that rotate at 10,000 RPM, have a data transfer rate of 10MByte/s (for each disk) and an average seek time of 8ms. The average size of an I/O operation is 32 KByte and the system’s data processing rate is limited by the disks. Each disk can handle only one request at a time but two (or more) disks can handle different requests.

(a) What is the average service time for an I/O request?

(b) What is the maximum number of I/Os per second (IOPS) for the system?
(c) Suppose now that you can replace the above 20 disks by 11 disks that have 190 GByte each, rotate at 12,000 RPM, transfer at 12 MByte/s, and have an average seek time of 6ms. What would be the average service time for an I/O request in the new system?

(d) What is the maximum number of IOPS in the new system?

(e) What is the disk utilization for both systems if they receive an average of 950 I/O requests per second?

(f) What would be the average response time for the two systems? Use the equation below for the disks as servers. Which system would have a lower response time?

\[
\text{Response\_time} = \text{Server\_time} \times (1 + \frac{\text{Server\_utilization}}{\text{Number\_of\_servers} \times (1 - \text{Server\_utilization})})
\]
9. A pipeline for a version of "MIPS" without delayed branches has the following average CPI penalties due to stalls:

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Hazard</th>
<th>Average Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loads</td>
<td>data</td>
<td>0.5</td>
</tr>
<tr>
<td>Branches</td>
<td>control</td>
<td>2</td>
</tr>
<tr>
<td>FP Mult</td>
<td>data</td>
<td>3</td>
</tr>
<tr>
<td>FP Add/Sub</td>
<td>data</td>
<td>1</td>
</tr>
<tr>
<td>FP Div</td>
<td>data</td>
<td>10</td>
</tr>
</tbody>
</table>

The CPI without pipeline stalls is 1.4 and the instruction mix is: Loads 14%, Branches 5%, FP Mult 5%, FP Add/Sub 7%, FP Div 3%, Stores 8%, Move 5%, Integer arithmetic & logical 28%.

(a) How much faster is the ideal pipelined machine versus the machine with these stalls?

(b) Suppose the average stall length for FP Mult can be reduced to 1.5 or the FP Divide average stall can be reduced to 5. How much performance will each scheme gain? Which is better?

(c) Assume the branches can be converted to delayed branches with the success in filling each of the 2 slots as shown below. How much faster is the machine with delayed branches? Assume none of the stall reductions in (b) have been implemented.

No slots filled 40%
1 slot filled 40%
2 slots filled 20%