1. 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.
2. 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 \( t_{o_1} \) and 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., \( t_i, t_{o_1}, t_{o_2} \leq 0.5 t_c \)

(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?
3. 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).
4. 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 $2^{n+k}$.
5. 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) This processor’s pipeline was designed to provide a throughput of 1 when only ALU instructions are executed. Why is the observed Clock_cycle_count for these instructions larger than 1?
(b) Calculate the average CPI (cycles per instruction) for the above benchmark.

<table>
<thead>
<tr>
<th>Instruction type</th>
<th>Fixed-point</th>
<th>Load</th>
<th>Store</th>
<th>Branch</th>
<th>Floating-Point</th>
<th>FP</th>
<th>FP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ALU ops.</td>
<td></td>
<td></td>
<td></td>
<td>Add/Sub</td>
<td>Mult</td>
<td>Div</td>
</tr>
<tr>
<td>Frequency</td>
<td>20%</td>
<td>20%</td>
<td>10%</td>
<td>16%</td>
<td>18%</td>
<td>12%</td>
<td>4%</td>
</tr>
<tr>
<td>Clock cycle count</td>
<td>1.5</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>20</td>
</tr>
</tbody>
</table>

(c) The design of the floating-point unit has been modified and now includes a Multiply-Add unit which is capable of performing a multiply operation of two operands followed by an addition of a 3rd operand to the product. A corresponding instruction MulAdd R_i, R_j, R_k has been added to the instruction set. This new instruction calculates $R_i = R_j + R_k \times R_{k+1}$, has an average cycle count of 7 and can replace a multiply instruction and a consecutive add instruction that uses the product as one of its operands. Why are the multiplier and multiplicand of the MulAdd instruction restricted to be in two consecutive registers $R_k$ and $R_{(k+1)}$?
(d) What would be the CPI of the modified design if the compiler is successful in replacing 50% of the multiplications (together with the follow-up additions) by MulAdd instructions?

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

(f) (Bonus) If the processor has a data cache and instruction cache, both with hit time of 1 cycle and miss penalty of 50 cycles. Calculate the miss rate of the instruction cache and estimate the miss rate of the data cache. Clearly state your assumptions.

6. 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?

\[
\text{Response\_time} = \text{Server\_time} \times (1 + \frac{\text{Server\_utilization}}{\text{Number\_of\_servers} \times (1 - \text{Server\_utilization})})
\]