Chapter 1
Parallel Computer Models

Problem 1.1

\[
CPI = \frac{45 \times 1 + 32 \times 2 + 15 \times 2 + 8 \times 2}{45 + 32 + 15 + 8} = \frac{155}{100} = 1.55 \text{ cycles/instruction.}
\]

MIPS rate = \(10^{-6} \times \frac{40 \times 10^6 \text{cycles/sec}}{1.55 \text{ cycles/instruction}} = 25.8 \text{MIPS.}\)

Execution time = \(\frac{(45000 \times 1 + 32000 \times 2 + 15000 \times 2 + 8000 \times 2) \text{cycles}}{(40 \times 10^6) \text{cycles/s}} = 3.875 \text{ ms.}\)

The execution time can also be obtained by dividing the total number of instructions by the MIPS rate:

\[
\frac{(45000 + 32000 + 15000 + 8000) \text{instructions}}{25.8 \times 10^6 \text{ instructions/s}} = 3.875 \text{ ms.}
\]

Problem 1.2 Instruction set and compiler technology affect the length of the executable code and the memory access frequency. CPU implementation and control determines the clock rate. Memory hierarchy impacts the effective memory access time. These factors together determine the effective CPI, as explained in Section 1.1.4.

Problem 1.3

(a) The effective CPI of the processor is calculated as

\[
CPI = \frac{15 \times 10^6 \text{ cycles/sec}}{10 \times 10^6 \text{ instructions/sec}} = 1.5 \text{ cycles/instruction.}
\]

(b) The effective CPI of the new processor is

\[
(1 + 0.3 \times 2 + 0.05 \times 4) = 1.8 \text{ cycles/instruction.}
\]
Parallel Computer Models

Therefore, the MIPS rate is
\[ 30 \times 10^6 \text{ cycles/sec} \div 1.8 \text{ cycles/instruction} = 16.7 \text{ MIPS}. \]

**Problem 1.4**
(a) Average CPI = \( 1 \times 0.6 + 2 \times 0.18 + 4 \times 0.12 + 8 \times 0.1 = 2.24 \text{ cycles/instruction}. \)
(b) MIPS rate = \( 40 \div 2.24 = 17.86 \text{ MIPS}. \)

**Problem 1.5**
(a) False. The fundamental idea of multiprogramming is to overlap the computations of some programs with the I/O operations of other programs.
(b) True. In an SIMD machine, all processors execute the same instruction at the same time. Hence it is easy to implement synchronization in hardware. In an MIMD machine, different processors may execute different instructions at the same time and it is difficult to support synchronization in hardware.
(c) True. Interprocessor communication is facilitated by sharing variables on a multiprocessor and by passing messages among nodes of a multicomputer. The multicomputer approach is usually more difficult to program since the programmer must pay attention to the actual distribution of data among the processors.
(d) False. In general, an MIMD machine executes different instruction streams on different processors.
(e) True. Contention among processors to access the shared memory may create hot spots, making multiprocessors less scalable than multicomputers.

**Problem 1.6** The MIPS rates for different machine-program combinations are shown in the following table:

<table>
<thead>
<tr>
<th>Program</th>
<th>Computer A</th>
<th>Computer B</th>
<th>Computer C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program 1</td>
<td>100</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Program 2</td>
<td>0.1</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>Program 3</td>
<td>0.2</td>
<td>0.1</td>
<td>2</td>
</tr>
<tr>
<td>Program 4</td>
<td>1</td>
<td>0.125</td>
<td>1</td>
</tr>
</tbody>
</table>

Various means of these values can be used to compare the relative performance of the computers. Definition of the means for a sequence of positive numbers \( a_1, a_2, \ldots, a_n \) are summarized below. (See also the discussion in Section 3.1.2.)
(a) Arithmetic mean: \( AM = (\sum_{i=1}^{n} a_i) / n. \)
(b) Geometric mean: \( GM = (\prod_{i=1}^{n} a_i)^{1/n}. \)

**Problem 1.7**
- An SIMD computer has a single control unit. The other processors are simple slave processors which accept instructions from the control unit and perform an identical operation at the same time on different data. Each processor in an MIMD computer has its own control unit and execution unit. At any moment, a processor can execute an instruction different from the other processors.
- Multiprocessors have a shared memory structure. The degree of resource sharing is high, and interprocessor communication is carried out via shared variables in the shared memory. In multicomputers, each node typically consists of a processor and local memory. The nodes are connected by communication channels which provide the mechanism for message interchanges among processors. Resource sharing is light among processors.
- In UMA architecture, each memory location in the system is equally accessible to all processors, and the access time is uniform. In NUMA architecture, the access time to a memory location depends on the proximity of a processor to the memory location. Therefore, the access time is nonuniform. In NORMA architecture, each processor has its own private memory; no memory is shared among processors. Each processor is allowed to access its private memory only. In COMA architecture, such as that adopted by KSR-1, each processor has its private cache, which together constitutes the global address space of the system. It is like a NUMA with cache in place of memory. A page of data can be migrated to a processor upon demand or be replicated on more than one processor.