# Question

1. Give the flynn's classification of computers.
   classified along the two independent dimensions of **Instruction** and **Data**
   SISD (Serial), SIMD, MISD, MIMD

2. Use an example to explain the differences between the *SIMD* and *MIMD* computers.
   In SIMD processors share the same instruction stream they execute the same code each on different fragments of data. Each processor has a data stream.
   In MIMD processors share nothing, each processor has his own instruction and data streams.

3. Explain the main differences between the **Blocking non-buffered** and the **Non-Blocking nonbuffered** send/receive operations of the message passing paradigm.
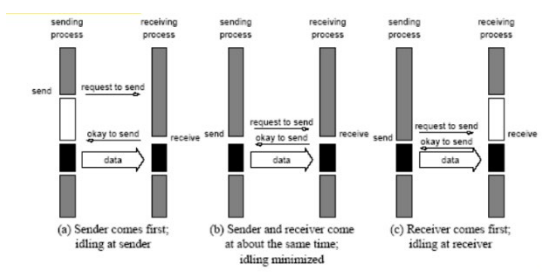
   Blocking non-buffered   non-blocking non-buffered
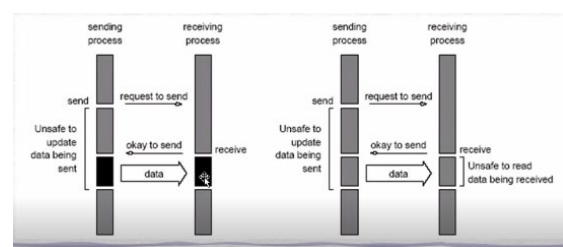
Blocking non-buffered: The sender issues a send operation and cannot proceed until a matching receive at the receiver's side is encountered and the operation is complete.

Blocking Buffered: When the sender process reaches a send operation it copies the data into the buffer on the receiver side and can proceed without waiting. When the receiver process encounters a receive operations it checks the buffer for data.

Non Blocking non buffered: The sender process needs not to be idle but instead can do useful computations while waiting for the send / receive operation to complete.

Non bocking buffered: The sender issues a direct memory access operation (DMA) to the buffer. The sender can proceed with its computations. At the receiver side, when a receive operation is encountered the data is transferred from the buffer to memory location.

4. Let's consider that a root process has N child processes. Let's consider that the root process has an array called **Data** of size N. Explain the following operations using the array Data.

    a. The root executes the operation **broadcast** of the message passing paradigm.

    Pi ← data

    b. The root executes the operation **scatter** of the message passing paradigm.

    Pi ← data[i]

    c. The root executes the operation **gather** of the message passing paradigm.

    P0 : data[i] ← Pi

    P0: data[0] ← R data[i] ← Pi   (Reduction)

5. Describe the **Task Farming** and the **Divide-and-Conquer** programming models and explain the main differences between them.

    In task Farming the master will decompose the problem into sub tasks and submit them to the workers. The workers will process the sub tasks simultaneously and and the master will gather the result from workers. The sub tasks my have different natures.

    In divide and conquer we decompose the problem into sub problems. The sub problems have the same nature as the original one and could be solved by recursively applying the same algorithm until we reach base case in which sub problems are solved directly.

6. Describe the **Task Farming** and the **Data Flow** programming models and explain the main differences between the

    In Task Farming the sub tasks are independent where in data flow they have dependencies among them

    Task Farming Each task simultaneously. Data flow parallelism tasks should be in the same level without dependencies

$$\sum_{i=0}^{2^n-1} = \sum_{i=0}^{\frac{2^n}{2}-1} + \sum_{\frac{2^n}{2}-1}^{2^n-1} \quad \sum_{i=0}^{7}$$

$$\sum_{i=0}^{3}$$

$$\sum_{i=4}^{7}$$

$$\sum_{4}^{5} - \text{⑤}$$

$$4 - ④$$

$$\sum \frac{7}{6} - ⑦$$

$$6 - ⑥$$