Assignment 2

Parallel and Distributed Computing

**Roll no.** 20L-1027

**Name:** Fatima Azfar

**Section:** BDS-6A

### Question 1:

**i. 4x4 two-dimensional mesh with wraparound links:**

Diameter: 8

Cost: 32

Bisection Width: 8

Arc-Connectivity: 4

Here is the network diagram:

o---o---o---o---o

| / | / | / | / |

o---o---o---o---o

| / | / | / | / |

o---o---o---o---o

| / | / | / | / |

o---o---o---o---o

| / | / | / | / |

o---o---o---o---o

**ii. 4x4 two-dimensional mesh with no wraparound links:**

Diameter: 6

Cost: 32

Bisection Width: 8

Arc-Connectivity: 4

Here is the network diagram:

o---o---o---o

| | | |

o---o---o---o

| | | |

o---o---o---o

| | | |

o---o---o---o

**iii. Three-dimensional hypercube (size=8 nodes):**

Diameter: 3

Cost: 24

Bisection Width: 4

Arc-Connectivity: 3

Here is the network diagram:

o---o---o---o

|\ |\ |\ |

| \ | \ | \ |

| \ | \ | \ |

o---o---o---o

|\ |\ |\ |

| \ | \ | \ |

| \ | \ | \ |

o---o---o---o

\ |\ \ |\ \

\ | \ | \ |

o---o---o

|\ |\ |

| \ | \ |

| \ | \ |

o---o---o

**iv. A complete binary tree of 3 levels:**

Diameter: 4

Cost: 15

Bisection Width: 4

Arc-Connectivity: 3

Here is the network diagram:

o

/ \

o o

/ \ / \

o oo o

### Question 2:

**What are the similarities and differences between Pthreads and OpenMP?**

Pthreads and OpenMP are two commonly used parallel programming models that support multithreading. While they share some similarities, they also have some key differences.

**Similarities:**

Both Pthreads and OpenMP are used to create multithreaded applications.

Both provide APIs for programmers to create and manage threads.

Both models can be used to parallelize code and increase performance.

**Differences:**

Pthreads is a low-level API, while OpenMP is a high-level API. Pthreads provide fine-grained control over threads and synchronization mechanisms, while OpenMP is designed to simplify parallel programming by providing constructs like parallel loops and sections.

Pthreads is platform-specific, while OpenMP is platform-independent. Pthreads require the programmer to write platform-specific code, while OpenMP can be used on any platform that supports it.

Pthreads can be used to create both shared and distributed memory programs, while OpenMP is primarily used for shared memory parallelism.

OpenMP is often easier to learn and use, particularly for parallelizing loops, while Pthreads can be more complex and require more programming effort to parallelize code.

Pthreads is suitable for systems programming, while OpenMP is more suited for scientific and engineering applications.

OpenMP provides more flexibility for task scheduling and load balancing, while Pthreads is more suitable for fine-grained parallelism.