

Quiz 5 (Solution Manual)

Parallel Distributed Computing

Name: _____

Roll No: _____

Section: 7

Q1) Define mutex lock and explain its type. (5)

Mutex lock is a binary nature variable that provides code wise functionality for (to) ensure mutual Exclusion.

Types

- ① Normal
- ② Recursive
- ③ Errorcheck.

Q2) Describe Alleviating Locking Overheads concept (10)

It is often possible to reduce idling overhead associated with locks using alternate function `pthread_mutex_trylock`.

This function attempts a lock on `mutex-lock` if successful function returns zero, if it is already locked by another thread, it returns value `EBUSY`.

```
int pthread_mutex_trylock(pthread_mutex_t *mutex-lock);
```

Q3) Consider the following and avoid deadlock (10)

```
int a[10], b[10], myrank;  
MPI_Status status;  
MPI_Comm_rank(MPI_COMM_WORLD, &myrank);  
MPI_Send(a, 10, MPI_INT, 1, 1, MPI_COMM_WORLD);  
MPI_Send(b, 10, MPI_INT, 1, 2, MPI_COMM_WORLD);  
MPI_Recv(b, 10, MPI_INT, 0, 2, MPI_COMM_WORLD);  
MPI_Recv(a, 10, MPI_INT, 0, 1, MPI_COMM_WORLD);
```


Solution Q(3)

```
int a[10], b[10], myrank;
```

```
MPI_Status status;
```

```
MPI_Comm_rank(MPI_COMM_WORLD, &myrank);
```

```
if (myrank == 0) {
```

```
    MPI_Send(a, 10, MPI_INT, 1, 1, MPI_COMM_WORLD);
```

```
    MPI_Send(b, 10, MPI_INT, 1, 2, MPI_COMM_WORLD);
```

```
}
```

```
else if (myrank == 1) {
```

```
    MPI_Recv(b, 10, MPI_INT, 0, 2, MPI_COMM_WORLD);
```

```
    MPI_Recv(a, 10, MPI_INT, 0, 1, MPI_COMM_WORLD);
```

```
}
```