**Home Assignment- Thread**

**ANS-1.**

child sees i=0/nparent sees i=0?nparent sees i=1?nparent sees i=2?nparent sees i=0?nchild sees i=1/nparent sees i=0?nparent sees i=1?nchild sees i=2/njanki@janki

**ANS-2.**

parent saysa:1

id:1 a:2 b:1

id:2 a:3 b:1

**ANS-3.**

In thread pool, we use N kernel threads to execute M tasks, where M can be much higher than N. The advantage over using kernel only threading is that we consume less resources, like memory (both virtual and physical) and kernel objects. We also get less context switches, which increases performance

The advantage over user only threading is that we can take advantage of multiple CPUs or multiple CPU cores. And if one task blocks, we can create another kernel thread to use the available CPU more efficiently.

So, we get the advantages of both approaches, at the expense of some additional user-mode scheduling. When we say "user-level threads map to kernel threads" we mean that the abstraction of threads presented to user-space is implemented using threads in kernel-space, with each user thread being represented by a kernel-implemented thread.

**ANS-4.**

m>>n relative values of m and n is the mapping a possibility.

m>>n this values or possibility for the relative values of m and n.