4.12 Using Amdahl's Law, calculate the speedup gain of an application that has a 60 percent parallel component for (a) two processing cores and (b) four processing cores.

The Amdahl's Law is

$$speedup <= 1/(S+(1-S)/N)$$

S is the portion of the application that must be performed serially on a system
N is the number of processing cores

(a) S is 0.4
N is 2
Speedup is 1/(0.4+0.6/2)=1.4286
(b) S is 0.4
N is 4
Speedup is 1/(0.4+0.6/4)=1.82

4.15 Consider the following code segment:

```
Pid_t pid;
pid = fork();
if (pid == 0) {/* child process */
fork();
thread_create( . . .);
}fork();
```

a. How many unique processes are created?
6
b. How many unique threads are created?
8

First fork() : T1 creates T2
Second fork(): T2 creates T3
thread_create: T2 creates a thread T2.2
So T2 can be considered as the parent thread T2.1 and the thread T2.2
T3 creates a thread T3.2
So T3 can be considered as the parent thread T3.1 and the thread T3.2
Third fork(): T1 creates T4
T2 creates T5
T3 creates T6

4.17 The program shown in Figure 4.16 uses the Pthreads API. What would be the output from the program at `LINE C` and `LINE P`?

LINE C: CHILD: value = 5

LINE P: PARENT: value = 0

Child process is a copy of parent process. So in the child process, it has the variable value whose value is 0. And the variable value in the child process is separate from the variable value in the parent process.

Because in the child process, the thread is created. And the thread can call the function called runner, which will set value (value is a variable) to 5.

So LINEC is CHILD: value = 5

Because in the parent process, no thread is created. The variable value is still 0.

So LINE P: PARENT: value = 0