CSIT 345 SU 21 Homework 2

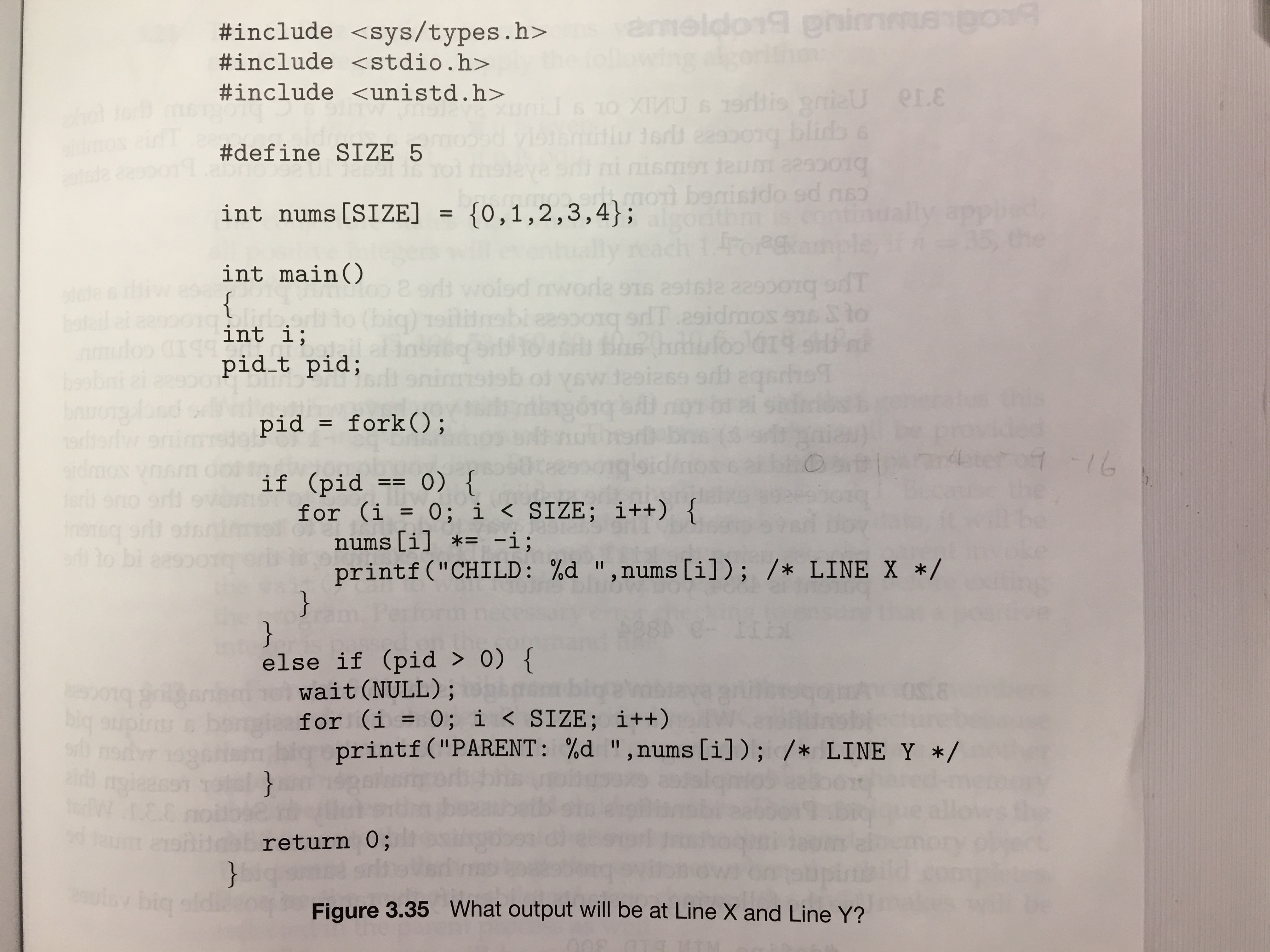
1. Describe the differences among short-term, medium-term, and long-term scheduling. (20pts)

Short term decides which of the in-memory process is to be executed.

Medium-term is a part of swapping and removes processes from memory.

Long term controls the number of processes being executed.

1. Using the program shown in the following figure, explain what the output will be at lines X and Y. (15pts)



Output:

CHILD: 0 CHILD: -1 CHILD: -4 CHILD: -9 CHILD: -16 PARENT: 0 PARENT: 1 PARENT: 2 PARENT: 3 PARENT: 4

1. Which of the following components of program state are shared across threads in a multithreaded process? (15 pts)

a. Register values

b. Heap memory

c. Global variables

d. Stack memory

b) heap memory and c) global variables

1. 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. (15 pts)
2. Two processing cores with 60 percent parallel component –

1/((serial fraction) + (parallel fraction / number of processors))

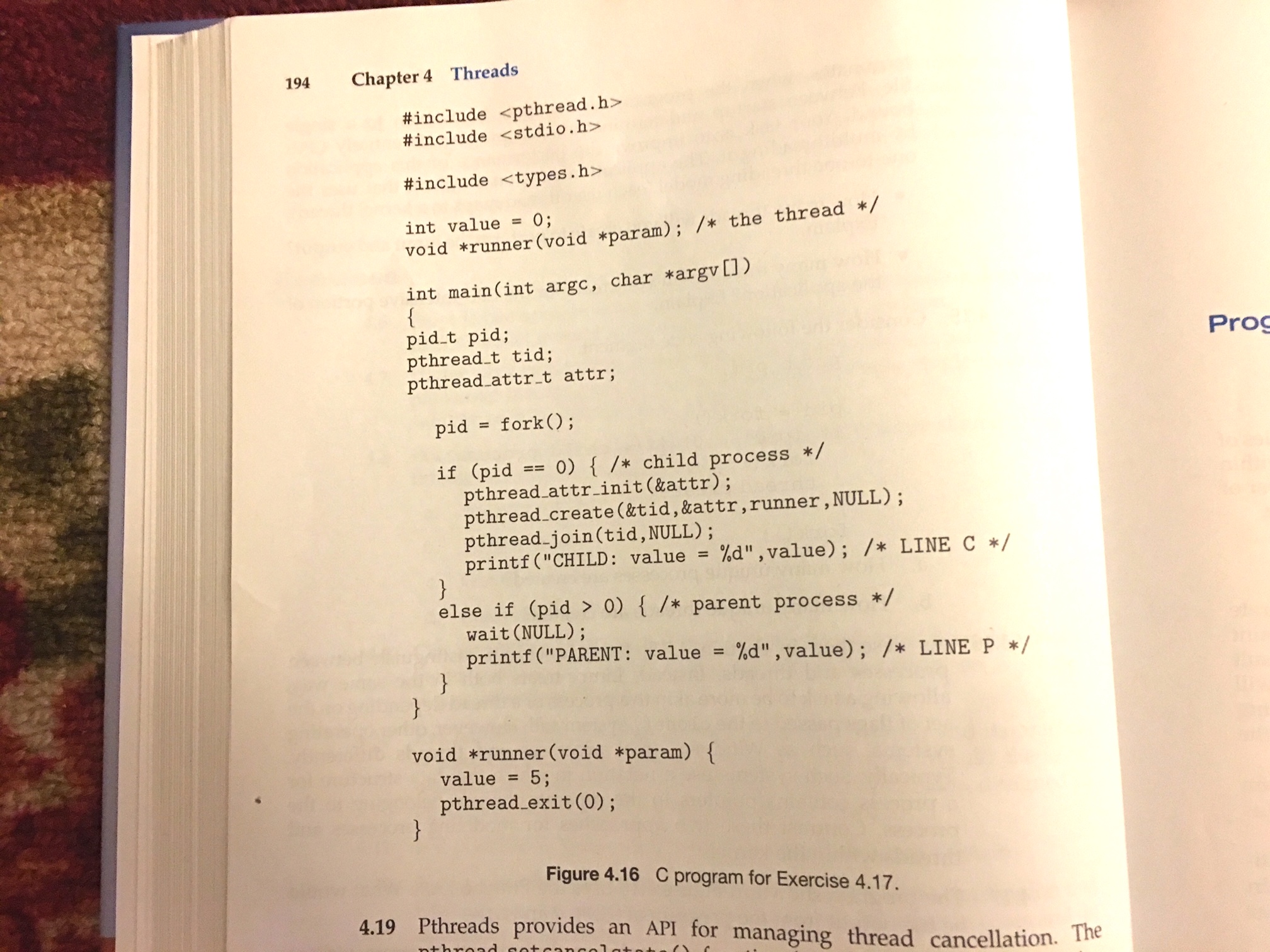
1/(0.4 + 0.6 / 2) = 1.4285

1. Four processing cores with 60 percent parallel-

1/((serial fraction) + (parallel fraction / number of processors))

1/ (0.4 + 0.6 / 4) = 1.818

1. The program shown in the following figure uses the Pthreads API. What would be the output from the program at LINE C and LINE P? (15 pts)



Line C output – 5

Line P output - 0

1. Consider a multiprocessor system and a multithreaded program written using the many-to-many threading model. Let the number of user-level threads in the program be more than the number of processors in the system. Discuss the performance implications of the following scenarios. (20 pts)

a. The number of kernel threads allocated to the program is less than the number of processors.

b. The number of kernel threads allocated to the program is equal to the number of processors.

c. The number of kernel threads allocated to the program is greater than the number of processors but less than the number of user level threads.

1. The number of kernels is less than the number of cores so some of the cores won’t be utilized.
2. Being that they are equal that means that we have the potential to use all processors.
3. This is most optimal due to the possibility of a kernel thread getting blocked which will now be redirected to another kernel thread.