Parent program

PSEUDOCODE:

READ argNum

IF argNum <> 4

WRITE error message

Do exit

ENDIF

READ maxSleepTime, firstInteger, secondInteger

IF maxSleepTime < 0 AND maxSleepTime > 50

WRITE “maximum sleep time is out of range.”

Do Exit

ENDIF

WRITE parentProcess, maxSleepTime, firstInteger, secondInteger

seed = processID + firstInteger

DO InitializeRandom Number Generator (using seed value)

sleepTime = rand() % maxSleepTime

DO sleep (using the sleepTime)

count = 0

WHILE count < 4

ADD 1 to count

If childProcess == true

DO ChildProgram

ELSE

IF parentProcess == true

WRITE “Forked child”, childpid

ELSE

WRITE “Fork failure”

ENDIF

ENDIF

ENDWHILE

Do Exit

Child program

PSEUDOCODE:

READ maxSleepTime, firstInteger, secondInteger, childNumber

seed = processID + childNumber

DO InitializeRandom Number Generator (using seed value)

sleepTime = rand() % maxSleepTime

DO sleep (using the sleepTime)

IF childNumber == 0

sum = firstInteger + secondInteger

WRITE childNumber, processID, sum

ELSE

IF childNumber == 1

difference = firstInteger – secondInteger

WRITE childNumber, processID, difference

ELSE

IF childNumber == 2

product = firstInteger \* secondInteger

WRITE childNumber, processID, product

ELSE

IF childNumber == 3

IF secondInteger <> 0

quotient = firstInteger/secondInteger

remainder = firstInteger%secondInteger

WRITE childNumber, processID, quotient, remainder

ELSE

WRITE “cannot divide by zero”, childNumber, processed

ENDIF

ENDIF

ENDIF

ENDIF

ENDIF

Do exit