

SUM LINES PROGRAM

Write a program that reads integer values from a file, and outputs the sum of the integers in each line and the sum of all the numbers to another file. The input file is named "numbers.dat" and has a title line containing the words "A Bunch of Integers". Each subsequent line has exactly 5 integers on it.

Example Input File:

```
A Bunch of Integers
1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5
```

The output file is named "sums.dat" and has the programmer's name on the first line. The second line is blank. After that, it shows each line total on a separate line, with the words "Line X Total: " (where X is the line number), and then the line total. Once each line is completed, a line of separators is displayed. The total sum of the entire file is displayed as the last line of the output file, with the words "File Total: " preceding the total.

Corresponding Output File:

Pamela J Vermeer

```
Line 1: 5
Line 2: 10
Line 3: 15
Line 4: 20
Line 5: 25
+++++
File Total: 75
```

Functional Decomposition: Top-Level Design

1. Setup the input and output files
2. Initialize variables
3. Read the first value in the file
4. While the last read was successful
 - 4.1 Process one line of data file
 - 4.2 Output results for that line
 - 4.3 Prepare for next time through loop
5. Output separator line and the file total
6. Close the files

Variables and Constants Identified From Specification and Top-Level Design

VARIABLES

lineTotal	sum of numbers on single line
fileTotal	sum of numbers in whole file
lineNumber	number of line just processed
curNum	most recent number read in from file

CONSTANTS

NUMBERS_ON_LINE 5 known number of integers on each line

Expand Top-Level Design

1. Setup the input and output files
 - 1.1 Open the input stream, connecting it to "numbers.dat"
 - 1.2 Read past the first line of input (title line in file)
 - 1.3 Open the output stream, connecting it to "sums.dat".
 - 1.4 Output the header information (programmer's name)
2. Initialize variables
 - 2.1 lineTotal = 0 (total of numbers on this line)
 - 2.2 lineNumber = 0 (current line just read in)
 - 2.3 fileTotal = 0 (total of numbers in whole file)
3. Read the first value in the file into curNum

Expand Top-Level Design

4. While the last read was successful
 - 4.1 Process one line of data file
 - 4.1.1 Initialize count to 0
 - 4.1.2 While count is less than NUMBERS_ON_LINE
 - 4.1.2.1 process this number
 - 4.1.2.1.1 add curNum to lineTotal
 - 4.1.2.1.2 add curNum to fileTotal
 - 4.1.2.2 increment count
 - 4.1.2.3 get another number
 - 4.2 Output results for that line
 - 4.2.1 Increment lineNumber
 - 4.2.2 Output lineNumber and lineTotal
 - 4.3 Prepare for next time through loop
 - 4.3.1 Reset lineTotal to 0

Expand Top-Level Design

5. Output separator line and the file total
 - 5.1 Initialize count to 0
 - 5.2 While count is less than SEPARATOR_COUNT (new constant)
 - 5.2.1 Output SEPARATOR
 - 5.2.2 Increment count
 - 5.3 Output file total
6. Close the files