

Student Name: **KEY SOLUTION** ID:

A file named "Data.txt" contains positive and negative integer numbers. Write a code that reads each number from the file and for each positive number the code writes in a new file the sum, product and number of digits of each number.

Your code should contain a function *splitFun()* that takes the number as input and if this number is positive, returns 1 and the sum and the number of digits, and 0 otherwise. The test for positivity should be made inside the function. The output is saved to a new file named "Results.txt", with the format shown below.

Note: Sample program run

```
345
-154
2315
33
0
-343
4355
0
```

number	# of dig.	sum	product
345	3	12	60
2315	4	11	30
33	2	6	9
4355	4	17	300

```
#include <stdio.h>
#include <stdlib.h>
int splitFun(int x, int *digits, int *sum, int *prod);

int main(){
    int n, s, d, p;
    FILE *inFile, *outFile;
    inFile = fopen("Integers.txt", "r");
    if (inFile == NULL){
        printf("Error opening file !");
        exit(1);
    }
    outFile = fopen("Result.txt", "w");
    fprintf(outFile, "number # of dig. sum product\n", n,s,d,p);
    fprintf(stdout, "number # of dig. sum product\n", n,s,d,p);
    while(fscanf(inFile, "%d", &n) != EOF){
        if(splitFun(n, &d, &s, &p)){
            fprintf(outFile, "%6d \t %5d \t %5d \t %5d\n", n,d,s,p);
            fprintf(stdout, "%6d \t %5d \t %5d \t %5d\n", n,d,s,p);
        }
    }
    fclose(inFile);
    fclose(outFile);
    return 0;
}
// -----
int splitFun(int x, int *digits, int *sum, int *prod){
    int r;
    *sum = 0;
    *digits = 0;
    *prod = 1;
    if(x <= 0)           // Is number <= 0
        return (0);
    while(x != 0){
        *digits = *digits + 1;           // Number of digits
        r = x % 10;                     // Remainder
        *sum = *sum + r;                 // Sum of digits
        *prod = *prod * r;               // Product of digits
        x = x / 10;
    }
    return (1);
}
```