HW1_P5 - Sum the digits in an integer

Student name :	Carter Hawks
Student email :	ckh170000@utdallas.edu
Class name :	2336.001_F18
Submitted on :	Sep 05, 2018 04:38 pm

solution.cpp

```cpp
/* Write your Analysis here

Given an input number, individually seperate and extract each digit of
    the number, and sum up all of the digits.

*/
/* Write your Design here

Each digit can be extracted using the following logic
Ex. The number 468 will be used as example.
468 % 100 = 68
468 - 68 = 400
400 / 100 = 4
Using those three operations, it can be determined that the first
    digit is 4.
This can be continued for each next digit.
468 - (100 * 4) = 68
68 % 10 = 8
68 - 8 = 60
60 / 10 = 6
Using these next set of operations, it can be determined that the next
    digit is 6. Finally, we can determine the final digit.
68 - (6 * 10) = 8
Now, all of the extracted digits can be summed.
4 + 6 + 8 = 18

Therefore, given an input of 468, the expected output is 18.

Loose Psuedo-code
1. Read in input number
2. Check if number is less than 1000 or greater than 99
3. Calculate first digit using operations listed above
4. Using the knowledge of the first digit, calculate second digit
5. Using the knowledge of the first & second digits, calculate the
    third digit.
6. Sum up the values, and print them out.


*/
// Write your code here
#include <iostream>
#include <stdio.h>
#include <string>
using namespace std;

int main(){
    int input;
    cin >> input;

    if(input < 1000 && input > 99){
```

```
        int dig1 = (input - (input % 100)) / 100;
        int dig2 = ((input - (dig1 * 100)) - (input - (dig1 * 100)) % 10) / 10;
        int dig3 = ((input - (dig1 * 100) - (dig2 * 10)) - (input - (dig1 * 100) - (dig2 * 10)) % 1);

        cout << dig1 + dig2 + dig3;
    } else {
        cout << -1;
    }

    // safe exit code
    return 0;
}
```

Name

Custom test case

Input

789

Output (Lines:2)

24

Expected Output (Lines:0)

Status

NA

Name

Custom test case

Input

346

Output (Lines:2)

13

Expected Output (Lines:0)

Status

NA

Name

Custom test case

Input

468

Output (Lines:2)

18

Expected Output (Lines:0)

Status

NA

Name

Default

Input

999

Output (Lines:2)

27

Expected Output (Lines:1)

27

Status

Pass