A well-known mathematical property states that a natural number is a multiple of three if and only if the sum of its digits is also a multiple of three. For instance, the sum of the digits of 8472 is $8 + 4 + 7 + 2 = 21$, which is a multiple of three. Therefore, 8472 is also a multiple of three.

Implement a recursive function that tells if a strictly positive natural number $n$ is a multiple of three or not.

```cpp
bool is_multiple_3 (int n);
```

**Interface**

- **C++**
  ```cpp
  bool is_multiple_3 (int n);
  ```
- **C**
  ```c
  int is_multiple_3 (int n);
  ```
- **Java**
  ```java
  public static boolean isMultiple3(int n);
  ```
- **Python**
  ```python
  is_multiple_3 (n) # returns bool
  is_multiple_3 (n: int) → bool
  ```

Solve this problem using a recursive function to return the sum of the digits of a natural number $n$.

```cpp
int sum_of_digits (int n);
```

**Interface**

- **C++**
  ```cpp
  int sum_of_digits (int n);
  ```
- **C**
  ```c
  int sum_of_digits (int n);
  ```
- **Java**
  ```java
  public static int sumOfDigits(int n);
  ```
- **Python**
  ```python
  sum_of_digits (n) # returns int
  sum_of_digits (n: int) → int
  ```

**Observation**

Here, you are allowed to use the operations of division and integer remainder only with the number 10. Otherwise, this exercise would be totally trivial!

**Problem information**

Author: Salvador Roura  
Translator: Carlos Molina  
Generation: 2023-07-14 18:10:00

https://jutge.org