rNumDigits

Write a **recursive** function that counts the number of digits for a non-negative integer. For example, 1234 has 4 digits. Write two versions of the function. The function `rNumDigits1()` returns the result. The function `rNumDigits2()` returns the result through the parameter *`result`*. The function prototypes are given as follows:

```
int rNumDigits1(int num);
void rNumDigits2(int num, int *result);
```

For separate program testing: The following sample program template is given for testing the functions:

```
#include <stdio.h>
int rNumDigits1(int num);
void rNumDigits2(int num, int *result);
int main()
{
   int number, result=0;

   printf("Enter the number: \n");
   scanf("%d", &number);
   printf("rNumDigits1(): %d\n", rNumDigits1(number));
   rNumDigits2(number, &result);
   printf("rNumDigits2(): %d\n", result);
   return 0;
}
int rNumDigits1(int num)
{
   /* Write your code here */
}
void rNumDigits2(int num, int *result)
{
   /* Write your code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:

```
Enter the number:
5
rNumDigits1(): 1
rNumDigits2(): 1
```

(2) Test Case 2:

```
Enter the number:
13579
rNumDigits1(): 5
rNumDigits2(): 5
```

(3) Test Case 3:

```
Enter the number:
12
rNumDigits1(): 2
rNumDigits2(): 2
```

(4) Test Case 4:

```
Enter the number:
2468
rNumDigits1(): 4
rNumDigits2(): 4
```