remDuplicates

Write a C function remDuplicates() that takes in an array ***a*** and array ***size*** as parameters. The array ***a*** contains integer numbers in **sorted ascending order**. The function removes duplicate values in the array and returns the new array size after removal. For example, if ***a***={1,2,3,3,5,6,6,6,9,9} with ***size***=10, then ***a***={1,2,3,5,6,9} and the new array size of 6 will be returned to the calling function after executing the function. There is no need to check user input errors in your program.

A sample program template is given below:

```
#include <stdio.h>
#define N 20
int remDuplicates(int a[], int size);
int main()
{
   int a[N],i,size;

   printf("Enter array size: \n");
   scanf("%d",&size);
   printf("Enter %d data:\n",size);
   for (i=0; i<size; i++)
      scanf("%d",&a[i]);
   size=remDuplicates(a,size);
   printf("remDuplicates(): \n");
   for (i=0; i<size; i++)
      printf("%d ",a[i]);
   printf("\n");
   return 0;
}
int remDuplicates(int a[], int size)
{
   /* Write your code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:

```
Enter array size:
10
Enter 10 data:
1 2 3 3 5 6 6 6 9 9
remDuplicates():
1 2 3 5 6 9
```

(2) Test Case 2:

```
Enter array size:
5
Enter 5 data:
1 2 3 3 5
remDuplicates():
1 2 3 5
```

(3) Test Case 3:

```
Enter array size:
```

```
2
Enter 2 data:
1 2
remDuplicates():
1 2
```