wordEquivalent

Write a C function wordEquivalent() that takes in a non-negative integer ***num*** between 0 and 99 inclusively and returns a character string ***str*** that contains the word-equivalent of each input digit in ***reverse*** to the calling function via call by reference. The function should also check whether the input integer exceeds 99. If the input integer exceeds 99, then the function should return the error message **"Input exceeds 99"** through the string ***str***.

A sample program template is given below:

```
#include <stdio.h>
#include <string.h>
void wordEquivalent(int num, char *str);
int main()
{
   int num;
   char str[40]="";

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

Some test input and output sessions are given below:

(1) Test Case 1

```
Enter the number:
5
wordEquivalent(): five
```

(2) Test Case 2

```
Enter the number:
25
wordEquivalent(): five two
```

(3) Test Case 3

```
Enter the number:
60
wordEquivalent(): zero six
```

(4) Test Case 4

```
Enter the number:
999
wordEquivalent(): Input exceeds 99
```