octStrToDec

Write a function octStrtodec() that accepts a character string ***str*** containing an octal number as parameter, converts the octal string into the equivalent decimal number (i.e. converts the number with base value 8 to base value 10), and returns the converted decimal number to the calling function.

A sample program template is given below:

```
#include <stdio.h>
int octStrTodec(char *str);
int main()
{
   char str[20],*sp;
   int num;

   printf("Enter an octal number: \n");
   scanf("%s",str);
   num=octStrTodec(str);
   printf("octStrTodec(): %d\n",num);
   return 0;
}
int octStrTodec(char *str)
{
   /* Write your code here */
}
```

Some test input and output sessions are given below:

(1) Test Case 1

```
Enter an octal number:
5
octStrTodec(): 5
```

(2) Test Case 2

```
Enter an octal number:
30
octStrTodec(): 24
```

(3) Test Case 3

```
Enter an octal number:
100
octStrTodec(): 64
```

(4) Test Case 4

```
Enter an octal number:
300
octStrTodec(): 192
```

```
#include <stdio.h>
int octStrTodec(char *str);
int main()
{
  char str[20],*sp;
  int num;

  printf("Enter an octal number: \n");
  scanf("%s",str);
  num=octStrTodec(str);
  printf("octStrTodec(): %d\n",num);
  return 0;
}
int octStrTodec(char *str)
{
   int remainder,dec = 0, n=0;

  int num = atoi(str);
  while(num!=0)
  {
     remainder = num%10;
     dec = dec+ remainder*pow(8,n);
     n++;
     num = num/10;
  }
  return dec;
}
```