hexStrToDec

Write a C function hexStrToDec() that takes in a string ***hex*** which contains a hexadecimal number as parameter, converts the hexadecimal string into its equivalent decimal number (i.e. converts the number (in ***str***) with base value 16 to base value 10) and returns the converted decimal number to the calling function.

A sample program template is given below:

```
#include <stdio.h>
#include <math.h>
#include <string.h>
int hexStrToDec(char *hex);
int main()
{
   int num;
   char hex[100];

   printf("Enter a hexadecimal number: \n");
   scanf("%s",&hex);
   num=hexStrToDec(hex);
   printf("hexStrToDec(): %d\n", num);
   return 0;
}
int hexStrToDec(char *hex)
{
   /* Write your code here */
}
```

Some test input and output sessions are given below:

(1) Test Case 1

```
Enter a hexadecimal number:
5
hexStrToDec(): 5
```

(2) Test Case 2

```
Enter a hexadecimal number:
2A
hexStrToDec(): 42
```

(3) Test Case 3

```
Enter a hexadecimal number:
45
hexStrToDec(): 69
```

(4) Test Case 4

```
Enter a hexadecimal number:
10E
hexStrToDec(): 270
```

```
#include <stdio.h>
#include <math.h>
#include <string.h>
int hexStrToDec(char *hex);
int main()
{
   int num;
   char hex[100];

   printf("Enter a hexadecimal number:
\n");
   scanf("%s",&hex);
   num=hexStrToDec(hex);
   printf("hexStrToDec(): %d\n", num);
   return 0;
}
int hexStrToDec(char *hex)
{
    int i, num=0;
    int len;
    char hexdigit;

    len = strlen(hex);

    for(i=0;i<len;i++)
    {
       hexdigit = hex[len-i-1];

       if (hexdigit >= '0' &&hexdigit<='9')
          num+= (hexdigit - '0') *pow(16,i);
       else
          num+= (hexdigit - 'A'+10) *pow
(16,i);
    }
    return num;
}
```

from the last digit

when hexdigit is in a range

can do subtraction for char and get int