

## power

Write a function that computes the power  $p$  of a positive number  $num$ . The power may be any integer value. Write two iterative versions of the function. The function **power1()** returns the computed result, while **power2()** passes the result through the pointer parameter **result**. In this question, you should not use any functions from the standard math library. The function prototypes are given below:

```
float power1(float num, int p);
void power2(float num, int p, float *result);
```

A sample program template is given below to test the functions:

```
#include <stdio.h>
float power1(float num, int p);
void power2(float num, int p, float *result);
int main()
{
    int power;
    float number, result=-1;

    printf("Enter the number and power: \n");
    scanf("%f %d", &number, &power);
    printf("power1(): %.2f\n", power1(number, power));
    power2(number, power, &result);
    printf("power2(): %.2f\n", result);
    return 0;
}
float power1(float num, int p)
{
    /* Write your code here */
}
void power2(float num, int p, float *result)
{
    /* Write your code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:  
Enter the number and power:

2 3

power1(): 8.00

power2(): 8.00

(2) Test Case 2:  
Enter the number and power:

2 -4

power1(): 0.06

power2(): 0.06

```
float power1(float num, int p)
{
    int i;
    while(num>0)
    {if(p>0)
    {
        p--;
        i = num;
        while(p!=0)
        {
            num = num*i;
            p--;
        }
        break;
    }
    else if(p==0)
    {
        num = 1.00;
        break;
    }
    else if(p<0)
    {
        i = num;
        while(p!=0)
        {
            num = num/i;
            p++;
        }
        num = num/i;
        break;
    }
    }
}
```

(3) Test Case 3:

Enter the number and power:

2 0

power1(): 1.00

power2(): 1.00

```
void power2(float num, int p, float *result)
{
    int i;
    while(num>0)
    {if(p>0)
    {
        p--;
        i = num;
        while(p!=0)
        {
            num = num*i;
            p--;
        }
        break;
    }
    else if(p==0)
    {
        num = 1.00;
        break;
    }
    else if(p<0)
    {
        i = num;
        while(p!=0)
        {
            num = num/i;
            p++;
        }
        num = num/i;
        break;
    }
    }
    *result = num;
}
```