# Write you own Power without using multiplication(*) and division(/) operators

### Method 1 (Using Nested Loops)

We can calculate power by using repeated addition.

For example to calculate 5^6.
1) First 5 times add 5, we get 25. (5^2)
2) Then 5 times add 25, we get 125. (5^3)
3) Then 5 time add 125, we get 625 (5^4)
4) Then 5 times add 625, we get 3125 (5^5)
5) Then 5 times add 3125, we get 15625 (5^6)

```
/* Works only if a >= 0 and b >= 0  */
int pow(int a, int b)
{
  if (b == 0)
    return 1;
  int answer = a;
  int increment = a;
  int i, j;
  for(i = 1; i < b; i++)
  {
     for(j = 1; j < a; j++)
     {
        answer += increment;
     }
     increment = answer;
  }
  return answer;
}

/* driver program to test above function */
int main()
{
  printf("\n %d", pow(5, 3));
  getchar();
  return 0;
}
```

## Method 2 (Using Recursion)

Recursively add a to get the multiplication of two numbers. And recursively multiply to get *a* raise to the power *b*.

```
#include<stdio.h>
/* A recursive function to get a^b
  Works only if a >= 0 and b >= 0  */
int pow(int a, int b)
{
   if(b)
     return multiply(a, pow(a, b-1));
   else
    return 1;
}

/* A recursive function to get x*y */
int multiply(int x, int y)
{
   if(y)
     return (x + multiply(x, y-1));
   else
     return 0;
}

/* driver program to test above functions */
int main()
{
  printf("\n %d", pow(5, 3));
  getchar();
  return 0;
}
```