Write an iterative O(Log y) function for pow(x, y)

Given an integer x and a positive number y, write a function that computes x^y under following conditions.
a) Time complexity of the function should be O(Log y)
b) Extra Space is O(1)

Examples:

```
Input: x = 3, y = 5
Output: 243

Input: x = 2, y = 5
Output: 32
```

We strongly recommend to minimize your browser and try this yourself first.

We have discussed recursive O(Log y) solution for power. The recursive solutions are generally not preferred as they require space on call stack and they involve function call overhead.

Following is C function to compute x^y.

```c
#include <stdio.h>

/* Iterative Function to calculate x raised to the power
int power(int x, unsigned int y)
{
    // Initialize result
    int res = 1;

    while (y > 0)
    {
        // If y is even, simply do x square
        if (y%2 == 0)
        {
            y = y/2;
            x = x*x;
        }

        // Else multiply x with result.  Note that this
        // is always executed in the end when y becomes
        else
        {
            y--;
            res = res*x;
        }
    }
    return res;
}

// Driver program to test above functions
int main()
{
    int x = 3;
    unsigned int y = 5;

    printf("Power is %d", power(x, y));

    return 0;
}
```

Output:

```
Power is 243
```