**Date-**

**Assignment No. :**

**Problem Statement:**

Program in C to multiply two numbers by Russian Peasant Multiplication.

**Theory:**

In mathematics The Russian peasant method one of two multiplication methods used by scribes, was a systematic method for multiplying two numbers that does not require the [multiplication table](https://en.wikipedia.org/wiki/Multiplication_table), only the ability to multiply and [divide by 2](https://en.wikipedia.org/wiki/Division_by_2), and to [add](https://en.wikipedia.org/wiki/Addition).

It decomposes one of the [multiplicands](https://en.wikipedia.org/wiki/Multiplicand) (preferably the smaller) into a sum of [powers of two](https://en.wikipedia.org/wiki/Powers_of_two) and creates a table of doublings of the second multiplicand. This method maybe called mediation and duplation, where [mediation](https://en.wikipedia.org/wiki/Division_by_two) means halving one number and duplation means doubling the other number. It is still used in some areas.

In actual figures, the multiplication of 26 and 278 is given below. When the number in the left column is even the entire row is crossed off (marked as ‘’). Except these rows, the numbers from the right columns of the other rows are selected for addition for the result of the multiplication.

|  |  |  |
| --- | --- | --- |
| **** | 26 | 278 |
| **** | 13 | 556 |
| **** | 6 | 1112 |
| **** | 3 | 2224 |
| **** | 1 | 4448 |

Now, the result of the multiplication is : 556+2224+4448=7228

**Algorithm:**

**Input specification:** The numbers which needs to be multiplied, say a and b.

**Output specification:** Multiplication result of the numbers a and b.

**Steps:**

1. Set p=x
2. Set q=y
3. Print p, q
4. If (p%2!=0) Then
   * 1. Set s=q
     2. Set a[0]=q
     3. i=i+1
5. Else
6. Print "X(STRUCK OFF)”

[End of If-else structure]

1. Repeat through Step i to Step x For p=x to p>1
2. Set p=p/2
3. Set q=q\*2
4. Print p, q
5. If (p%2!=0)
6. Set s=s+q
7. Set a[i]=q
8. i=i+1
9. count=count+1
10. Else
11. Print "X(STRUCK OFF)" [End of For loop]
12. If (x%2=0)
    * 1. a[count]=0

[End of If]

1. Print a[0]
2. Repeat step 9 For i=1 to count
3. Print “+” a[i]
4. i=i+1 [End of For loop]
5. Print s

**Source Code:**

#include<stdio.h>

#include<conio.h>

int main()

{

int x,y,p,q,s,a[50],i,count; char ch;

do{

i=0;

s=0; //initialization of the sum

count=0;

printf("Enter the numbers:");

scanf("%d %d",&x,&y); //scanning of the numbers

printf("\nThe multiplication method of %d & %d\nusing Russian

Peasant Method is as follows:\n\n",x,y);

p=x; //setting the values of the inputs to two temporary variables

q=y;

printf("%d\t\t%d",p,q);

if(p%2!=0) //checking if the variable in the left column is odd or not

{

s=q; //setting sum to q when left column has odd number

a[0]=q;

i++;

}

else

printf("X(STRUCK OFF)"); //when left column has even number

while(p>1) //until left column has not reached 1

{

p=p/2; //dividing the left column by 2

q=q\*2; //multiplying the right column by 2

printf("\n%d\t\t%d",p,q);

if(p%2!=0)

{

s=s+q; //adding q to s when left column is odd

a[i]=q; //taking the values of q for printing

i++;

count++;

}

else

printf("X(STRUCK OFF)");

}

count--; /\*decrementing count to avoid unnecessary garbage value

storing\*/

printf("\n\nThe answer is:\n");

for(i=0;i<=count;i++)

{

printf("%d+",a[i]); //printing the values of qs

}

printf("\b=%d",s);

printf("\n\nWhich satisfies the statement %d\*%d=%d",x,y,x\*y);

loop: printf("\nDo you want to continue?(Y/N)");

scanf(" %c",&ch);

if(ch=='n'||ch=='N')

return 0;

}while(ch=='y'||ch=='Y');

if(ch!='y'||ch!='Y')

{

printf("Wrong choice given");

goto loop; //shifting the control to the continuity part

}

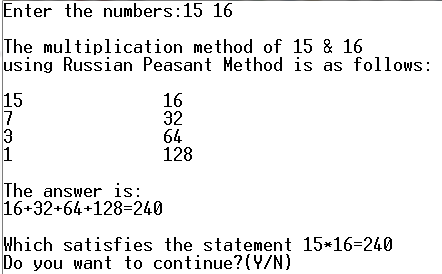
getch();

return 0;

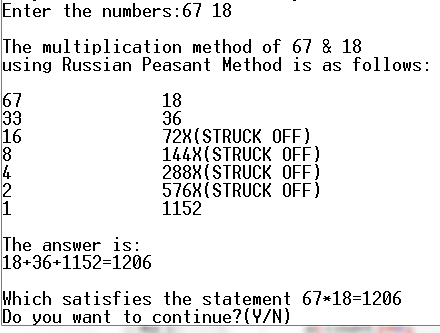
}

**Input & Output:**

Set 1:



Set 2:



**Discussion:**

1. This method is useful as it does not require to remember the multiplication tables except for 2.
2. But for negative numbers and floating point numbers this method will not work.
3. Also this program uses iterative approach to fulfill its purpose. Hence it can be problematic for larger numbers. Again, when multiplying with 0, if we set the 0 to the right column and a larger number to the left column then a repetitious 0 will come on the right column.