

BOOTH ALGORITHM:

```
#include <stdio.h>
```

```
void booth_multiplication(int multiplicand, int multiplier, int *result) {
```

```
    *result = 0;
```

```
    int multiplier_bits = 0;
```

```
    int sign_bit = multiplier & 0x80000000;
```

```
    while (multiplier != 0) {
```

```
        int ls_bit = multiplier & 0x1;
```

```
        if (ls_bit != multiplier_bits) {
```

```
            if (ls_bit == 1) {
```

```
                *result += multiplicand;
```

```
            } else {
```

```
                *result -= multiplicand;
```

```
            }
```

```
        }
```

```
        multiplicand <<= 1;
```

```
        int msb = multiplicand & 0x80000000;
```

```
        if (msb != 0) {
```

```
            multiplicand |= 0xFFFFFFFF;
```

```
        }
```

```
        multiplier >>= 1;
```

```
        multiplier_bits = ls_bit;
```

```
    }
```

```
    if (sign_bit != 0) {
```

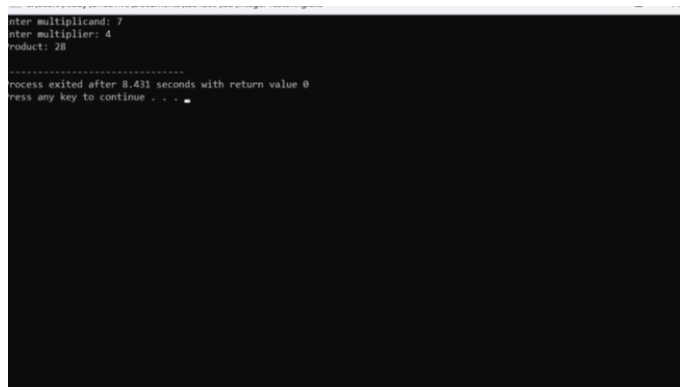
```
        *result = -*result;
```

```
    }
```

```
}
```

```
int main() {  
    int multiplicand, multiplier;  
    int product;  
  
    printf("Enter multiplicand: ");  
    scanf("%d", &multiplicand);  
  
    printf("Enter multiplier: ");  
    scanf("%d", &multiplier);  
  
    booth_multiplication(multiplicand, multiplier, &product);  
  
    printf("Product: %d\n", product);  
  
    return 0;  
}
```

Input&output:



```
Enter multiplicand: 7  
Enter multiplier: 4  
Product: 28  
-----  
Process exited after 0.431 seconds with return value 0  
Press any key to continue . . .
```