

# Arithmetic

● Graded

## Student

SRUTHI SUBRAMANIAN

## Total Points

180 / 180 pts

## Autograder Score

180.0 / 180.0

## Passed Tests

Test 1 (20/20)

Test 2 (20/20)

Test 3 (20/20)

Test 4 (20/20)

Test 5 (20/20)

Test 6 (20/20)

## Autograder Results

Test 1 (20/20)

Test 2 (20/20)

Test 3 (20/20)

Test 4 (20/20)

Test 5 (20/20)

Test 6 (20/20)

## Submitted Files

```
1  #include <stdio.h>
2
3  int main() {
4      int n;
5      scanf("%d",&n);
6      int GCD(int X,int Y){
7          //to return gcd(X,Y)
8          if (Y==0){
9              return X;
10         }
11         return GCD(Y,X%Y);
12     }
13     int* Euc(int X, int Y, int d){
14         int a=Y;
15         int b=X%Y;
16         if (a==d && b==0){
17             int final[2];
18             final[0]=1;
19             final[1]=1-X/Y;
20             int *finpt;
21             finpt=final;
22             return finpt;
23         }
24         int p,s,q,r;
25         int *P=Euc(a,b,d);
26         p=P[0];
27         s=P[1];
28         q=X/Y;
29         r=X%Y;
30         int Final[2];
31         Final[0]=s;
32         Final[1]=p-q*s;
33         int *Finpt;
34         Finpt=Final;
35         return Finpt;
36     }
37     if (n==0){
38         //Modular exponentiation
39         int A,B,C;
40         scanf("%d %d %d", &A,&B,&C);
41         //output A^B mod C
42         //first we find the binary expansion of B:
43         int k=1;
44         int l=0;
45         while (B/k!=0){
46             k=k*2;
47             l=l+1;
48         }
49         k=k/2;
```

```

50     int binB[l];
51     int q,r;
52     for (int i=0;i<l;i++){
53         q=B/k;
54         k=k/2;
55         binB[l-i-1]=q;
56         B=B-k*q*2;
57     }
58     //binB stores the binary expansion of B, with the lsb at the start of the array
59     //length of binB is l
60     long int Ap=A;
61     long int Af=1;
62     if (binB[0]==1){
63         Af=A;
64     }
65     for (int i=1;i<l;i++){
66         //compute A^(powers of 2) and multiply the needed powers
67         Ap=(Ap*Ap)%C;
68         if (binB[i]==1){
69             Af=(Af*Ap)%C;
70         }
71     }
72     printf("%ld",Af);
73 }
74 else {
75     //Euclid's algorithm
76     int x,y;
77     scanf("%d %d",&x,&y);
78     int g=GCD(x,y);
79     int *L=Euc(x,y,g);
80     int a=L[0];
81     if (a>0){
82         a=a%y;
83     }
84     else{
85         while (a<0){
86             a=a+y;
87         }
88     }
89     if (g>1){
90         printf("%d 0",g);
91     }
92     else{
93         printf("%d %d",g,a);
94     }
95 }
96 return 0;
97 }

```