###### Bài 1:

*Cho p=2 147 483 647; W=8*

1. *Hãy nhập số nguyên a* ∈*Fp từ bàn phím và biểu diễn a thành dạng mảng của các từ W-bit.*
2. *Hãy nhập một mảng của 4 từ W-bit và tính giá trị của mảng đó trong Fp.*

*Test:*

*<a>*

* *a= 673654876 => mảng A=*
* *a= 2354712 => mảng A=*
* *a= 873546401=> mảng A=*
* *a= 2147483647 => mảng A=*
* *a= 2147483645 => mảng A=*

*<b>*

* *mảng A=[* *120 235 65 72] => a=*
* *mảng A=[* 127 255 255 255*] => a=*
* *mảng A=[* *0 32 14 251] => a=*
* *mảng A=[* *65 132 250 97] => a=*
* *mảng A=[* *124 160 82 231] => a=*
* *mảng A= [251 39 101 219] =>a=*

###### **Bài 2:**

*Cho p=2147483647; W=8. Với t là độ dài từ của p, hãy nhập các giá trị a, b ∈ [0, ) từ bàn phím và AD thuật toán cộng chính xác bội để thực hiện tính giá trị c, với c = a + b mod .*

*Test:*

* *a=63245210; b=2354125* 
  + *=> (e,c)=*
* *a=2147483646; b=4147483647*
  + *=> (e,c)=*
* *a=2147483696; b=4147483645*
  + *=> (e,c)=*
* *a=3654254; b= 12546721*
  + *=> (e,c)=*
* *a=36215467; b=54210320*
  + *=> (e,c)=*

###### **Bài 3:**

*Cho p=2147483647; W=8. Với t là độ dài từ của p, hãy nhập các giá trị a, b ∈ [0, ) từ bàn phím và AD thuật toán trừ chính xác bội để thực hiện tính giá trị c, với c = a - b mod .*

*Test:*

* a=65324842; b=2354
  + => (e,c)=
* a=3216; b=562489
  + => (e,c)=
* a= 8654; b=365412
  + => (e,c)=

###### Bài 4: cài đặt phép cộng trên Fp

Cho w=8, p là một số nguyên tố lớn hãy nhập 2 số a và b ∈ [0, và tính c = a + b mod theo thuật toán cộng trên Fp.

Yêu cầu:

* In biểu diễn ma trận của p
* In kết quả a+b theo tt cộng chính xác bội dạng ma trận và giá trị bit nhớ
* In kết quả cộng trên trường Fp

Test:

* VD1: *P=2147483647; a=2147483646; b= 2147483643*
  + P=
  + Cộng chính xác bội dạng ma trận: (e;C=A+B)=
  + Cộng trên Fp: c=
* VD2: p=479001599; a= 347483646 ; b= 474836419
  + P=
  + Cộng chính xác bội dạng ma trận: (e;C=A+B)=
  + Cộng trên Fp: c=
* VD3: p=2971215073; a=2147483645; b= 2971215070
  + KQ:
    - P=
    - Cộng chính xác bội (e;C=A+B)=
    - Cộng trên Fp: c=

###### Bài 5: cài đặt thuật toán trừ trên Fp

Cho w=8, p là một số nguyên tố lớn hãy nhập 2 số a và b ∈ [0, và tính c = a - b mod theo thuật toán cộng trên Fp.

Yêu cầu:

* In kết quả a+b theo tt trừ chính xác bội dạng ma trận và giá trị bit nhớ
* In kết quả trừ trên trường Fp

*Test: Sub\_Fp*

* *Ví dụ1: p=2147483647; a=1387624979; b= 1568424364*
  + *Trừ chính xác bội: (e;C=A-B)=*
  + *Trừ trên Fp: c=*
* *VD2:p=433494437, a= 333294897, b= 183494999*
  + *Trừ chính xác bội: (e;C=A-B)=*
  + *Trừ trên Fp: c=*
* *VD3: p= 433494437, a=233294897, b= 383494999*
  + *Trừ chính xác bội: (e;C=A-B)=*
  + *Trừ trên Fp: c=*

###### Bài 6:Cài đặt chương trình tính phép nhân

*Cho w=8, p là một số nguyên tố lớn, nhập từ bàn phím hai số nguyên a và b thuộc [0,p-1). Sử dụng thuật toán Integer multiplication (operand scanning form) để tính c = a.b.*

*Test:*

* *Ví dụ 1: p=2147483647; a=524647; b= 32549*
  + *c=*
* *Ví dụ 2: p=2147483647; a=2348762; b= 98637*
  + *c=*

###### Bài 7: Cài đặt chương trình tính ước chung lớn nhất của 2 số a và b: gcd(a,b) theo thuật toán Euclide

Yêu cầu: nhập hai số a và b luôn dương

In số a, b

In gcd(a,b)

Test: Euclide\_gcd.m

* VD1:  *a=45632454; b=23454563*
  + *=> gcd=*
* *VD2: a= 28150488 b= 10507620*
  + *=> gcd=*
* *a=864879; b=356478*
  + *=> gcd=*

###### Bài 8. Cài đặt thuật toán Euclide mở rộng tính ước chung lớn nhất của a và b

Yêu cầu:

* In gcd(a, b)
* Nếu gcd(a,b)=1 thì in a-1 mod b và b-1 mod a

Test:

* VD1: a=34568734; b=487345936
  + gcd=
  + a-1 mod b=
  + b-1 mod a=
* *a=45632454; b=23454563*
  + *=> gcd=*
  + a-1 mod b=
  + b-1 mod a=

###### Bài 8.1 Cài đặt thuật toán tính nghịch đảo trên Fp dùng Euclide mở rộng

*Test:*

* *Ví dụ 1: p= 489573857; a= 45682375*
  + *=> a-1 mod p=*
* p=*489573857;* a=5464144
  + *=> a-1 mod p=*

###### Bài 9: Cài đặt chương trình tính nhân bình phương có lặp a^k mod n

Test:

* VD1: n=6784; a=3453; k=2546
  + => KQ:
* VD2: n=658497624; a=658464; k=356482
  + =>KQ:

###### Bài 10: Viết chương trình tìm tất cả các số nguyên tố <=n với n nhập vào từ bàn phím

* Cách thông thường
* Sàng Erosthenes

Sàng phân đoạn

Test:

* VD1: n=50:
* VD2 n=100:
* VD3 n=150:

###### Bài 11. Viết chương trình tìm một thừa số không tầm thường của một số n nhập từ bàn phím

Test:

* VD1: n=2309832423
  + => d=
* VD2: n=3473657
  + => d=
* VD3: n=455459
  + => d=
* VD4: n= 69875624
  + => d=

###### Bài 12. Viết chương trình kiểm tra tính nguyên tố của một số n nhập vào từ bàn phím theo cách thông thường.

Test:

* n= 20863
  + =>
* n=99991
  + =>
* n= 5642581
  + =>
* n=99109
  + =>

###### Bài 13. Viết chương trình kiểm tra tính nguyên tố của một số n nhập vào từ bàn phím theo thuật toán Fermat

Test:

* n= 20863
  + =>
* n=99991
  + =>
* n= 5642581
  + =>
* n=99109
  + =>
* n=561 (với cơ sở a =2
  + =>

###### Bài 14. Viết chương trình kiểm tra tính nguyên tố của một số n nhập vào từ bàn phím theo thuật toán Miller-rabin

Test:

* n= 20863
  + =>
* n=99991
  + =>
* n= 5642581
  + =>
* n=99109
  + =>
* n=561 với cơ sở a=2
  + =>
* n=91 (với cơ sở a= 9, 10, 12, 16, 17, 22, 29, 38, 53, 62, 69, 74, 75, 79

###### Bài 15. Viết chương trình sinh số nguyên tố sử dụng kiểm tra Miller-Rabin

Test:

* k=20
* k=10
* k= 15
* k=12
* k= 25

###### Bài 16. Cài đặt thuật toán tìm kiếm mẫu P trong đoạn văn bản T kết quả trả về vị trí xuất hiện của mẫu P theo phương pháp vét cạn (không sử dụng hàm có sẵn)

###### Bài 17. Cài đặt thuật toán tìm kiếm mẫu P trong đoạn văn bản T kết quả trả về vị trí xuất hiện của mẫu P theo phương pháp Boyer-Moore

###### Bài 18. Cài đặt thuật toán tìm kiếm mẫu P trong đoạn văn bản T kết quả trả về vị trí xuất hiện của mẫu P theo phương pháp Knuth-Morris-Pratt