Documentatie

* Mai intai am verificat daca numarul introdus este prim
  + daca numarul este mai mica decat doi programul se tarmina
  + am mers prin toate numerele de la 2 pana la sqrt(p) si am verificat daca p se imparte la unul dintre ele
    - am folosit conditia i\*i<p (unde i este numarul la care am ajuns in iteratie)
  + daca nu se imparte niciunul, programul trece la cautarea generatorului
* Caut generatorul
  + Iau fiecare numar de la 2 pana la p-1(g)
  + Ii calculez resturile puterilor in modulo p(puterile de la 1,la g-1)
    - Pe prima pozitie pun 0, deoarece orice numar ridicat la 0 va da 0
    - Pentru urmaotarele daca o putere este 1, trebuie sa trec la uramtorul numar deoarece cel actual nu este generator pentru ca eu nu ajung niciodata la puterea g iar daca o putere inainte de aceea este 1 inseamna ca puterile se vor repeta
    - Daca am precut prin toate puterile si nu am gasit un 1, , inseamna ca puterile acopera tot Zg, pe care le stochez intr-un vector numit powers care are pe pozitia 0 pe g^0, in 1 pe g^1 … etc.
* Citesc mesajele, cel decodat si cel codat
* Codez primul mesaj
  + Iau fiecare caracter si il caut in sirul de carectere alfabet care contine toate literele alfabetului(scrise cu majuscula)
  + Cand o gasesc ii retin pozitia
  + Dupa aceea caut valoarea care se afla pe acea pozitie in vectorul powers
  + Dupa ce am valoare din powers, afisez alfabet(valoare), caracterul care se afla pe pozitia valorii din powers
  + Fac asta pana se termina lungimea mesajului necodat
* Decriptez al doilea mesaj
  + Iau fiecare caracter pe rand din mesaj
  + Ii caut pozitia din alfabet printre valorile din vectorul powers
  + Odata ce am gasit valoarea doar afisez pozitia pe care se afla pe aceeas pozitie in alfabet pe care se afla valoarea in powers

Cod sursa: (https://tinyurl.com/rplynmv)

.data

alfabet: .asciiz "ABCDEFGHIJKLMNOPQRSTUVWXYZ" #un string cu char din alfabet de care ne folosim pt a evita lucrul cu ASCII

p : .word 4 #spatiu pentru p-ul citit

powers: .space 400

mesajPrim: .asciiz "Numarul nu este prim"

newLine: .asciiz "\n"

mesaj1: .space 400

mesaj2: .space 400

mesajGenerator: .asciiz "Generatorul este: "

.text

main:

li $v0,5

syscall

sw $v0,p #citesc un p si il slavez intr-o variabila

li $t0,2 #de la 2

lw $t1,p #la p

blt $t1,$t0,exitPrim #daca este mai mic decat 2

prim: #verific daca numarul este prim

mul $t2,$t0,$t0 #calculez $t0\*$t0 a.k.a i\*i

bge $t2,$t1,ePrim # verific daca i\*i < p

rem $t3,$t1,$t0 #calcuelz restul

beqz $t3,exitPrim #daca restul e 0, numarul are divizori, deci nu e prim

addi $t0,$t0,1 #adaug la iterator

j prim # jump inapui in loop

exitPrim:

la $a0,mesajPrim #afisez mesajul din memorie pt numerele care nu sunt prime

li $v0,4 # codul pt afisarea unei adrese

syscall #execut

j exit # jump in exit-ul default

ePrim:

lw $t0,p # in incarc pe p in t0

subu $t1,$t0,1 # g-ul (aka p-1) e in $t1

li $t2,1 #t2 e g care initial e 2 , adaug 1 in loop

li $t3,1 # incep de la puterea 0

li $t6,0 #pozitia in vectorul de puteri

li $t5,1 # t5 e 1 pentru ca primul element din orice vector de puteri (element^0)

li $t7,1 # verific daca restul la orice iteratie e 1, daca e, numarul testat pt a vedea daca este generator, nu e generator

genPuteri:

addi $t2,$t2,1 #elementul pe care il testez acum

sw $t5,powers($t6) #salvez pe prima pozitie 1

move $t4,$t2 # fac o copie la $t2 (care e incercare pt generator, ca mai tarziu sa-l inmultesc pe $t4 cu $t2)

genPuteri2:

bge $t3,$t1,amGasitGenerator #inseamna ca am gasit un vector ( cu elementele de la 0 la g)

rem $t5,$t4,$t0 #calculez restul si il salvez in t5

beq $t5,$t7, nuEGenerator #daca t5(restul) e 1, inseamna ca numarul nu e generator

mul $t6,$t3,4 # calculez pozitia in vector ,$t6 e pozitia($t3,care e puterea \*4(bytes pt un int))

sw $t5,powers($t6) #il salvez pe $t5(restul) in vectorul powers pe pozitia $t6

mul $t4,$t4,$t2 #inmultesc pe $t4 cu t2(care e elementul testat pt conditia de generator))

rem $t4,$t4,$t0 #salvez mereu in $t4 restul modulo p, ca sa nu ajungem la numere mai mari decat p

addi $t3,$t3,1 # adaug 1 la putere

j genPuteri2 # jump la generarea urmatorului rest

nuEGenerator:

li $t3,1 #daca nu e generator resetez puterea

j genPuteri #ma intorc sa testez unrmatorul element

amGasitGenerator:

#in mom asta am in $t2 generatorul

# si in "powers" puterile

la $a0,mesajGenerator

li $v0,4

syscall #afisez mesajul ca am gasit generatorul

move $a0,$t2

li $v0,1

syscall #afisez generatorul

la $a0,newLine

li $v0,4

syscall #afisez newLine

la $a0,mesaj1 #incarc adresa din memorie unde vreau sa fie stocat sirul de caractere

li $a1,99 # incarc lungimea sirului

li $v0,8

syscall #citesc sirul de caractere decriptat

la $a0,mesaj2

li $a1,99

li $v0,8

syscall #citesc sirul de caractere criptat

li $t0,0 #pozitia in mesaj

lb $t1,mesaj1($t0) # incarc in $t1 primul caracter

criptare:

li $t7,10 #incarc 10 care e codul pt newLine

beq $t1,$t7,decriptare #am trecut prin tot sirul deci mesajul a fost criptat, acum il tratam pe cel de-al doilea

li $t3,0 #pozitia in alfabet

cautInAlfabet: #caut caracter cu caracter in sirul alfabet

lb $t6, alfabet($t3) #incarc in t6 litera de pe pozitia t3

beq $t1,$t6,afisezPozInAlfabet # daca e egala cu litera din mesajul dat de la tastatura atunci afisez corespondenta lui conform generatorului

addi $t3,$t3,1 # trec la urmatoare pozitie in alfabet

j cautInAlfabet #mai incerc sa caut caracterul in alfabet

afisezPozInAlfabet: #am gasit pe ce poz e in alfabet

li $t5,4

mul $t4,$t3,$t5 # pozitia in powers e poz din alfabet \* 4

lw $t7,powers($t4) #incarc in t7 restul de pe pozitia t4

lb $a0,alfabet($t7) #incarc in a0 byte-ul din alfabet de pe pozitia t7

li $v0,11

syscall #afisez byteul

addi $t0,$t0,1

lb $t1,mesaj1($t0) #trec la urmatorul caracter din mesajul necriptat

j criptare #in criptez

decriptare:

la $a0,newLine

li $v0,4

syscall # afisez o linie noua

li $t0,0

lb $t1,mesaj2($t0) #incarc in t1 caracterul de pe pozitia t0

decriptare2:

li $t7,10 #10 e codul pt newLine (asta e pt mars mai mult, altfel ramane intr-un loop infinit)

beqz $t1,exit # la al doilea mesaj, in MARS trebuie beq $t1,$t7 pt ca imi consiera un newLine si la al doilea cuv

li $t3,0 #pozitia in powers

cautInPowers: #caut caracterul(pozitia lui in alfabet) din mesajul codat in powers

lb $t6, powers($t3) #incarc in t6 restul de pe pozitia t3

lb $t5, alfabet($t6) #incarc in t5 pozitia t6 din alfabet

beq $t1,$t5,afisezCaracterDecodat #daca caracterul din mesajul criptat de potriveste cu

addi $t3,$t3,4 #trec la urmatorul rest

j cautInPowers #incec iara

afisezCaracterDecodat: #am gasit corespondenta

# t1 e litera din mesaj criptat

# t3 e pozitia din powers(mult de 4)

# t6 e poz literei in alfabet

li $t5,4

div $t4,$t3,$t5 # aflu pozita in alfabet

lb $a0,alfabet($t4) #incarc in a0 caracterul din alfabet care corespunde caracterului din mesajul codat

li $v0,11

syscall # afisez un byte

addi $t0,$t0,1 #trec la urmatorul caracter in mesajul codat

lb $t1,mesaj2($t0) #il incarc in $t1

j decriptare2 # here we go again

exit:

li $v0,10 #termin programul

syscall

Exemple de date:

 