

Programación Funcional y Lógica.

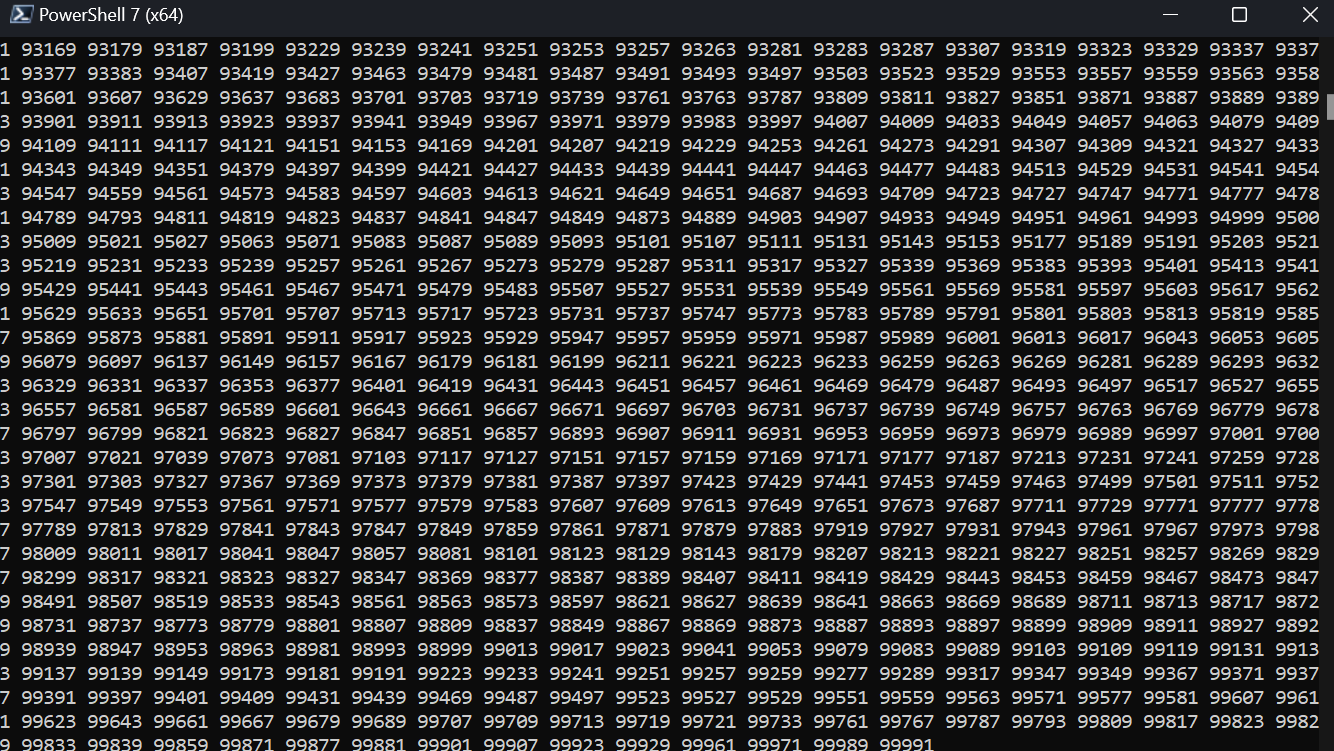
17/10/2022

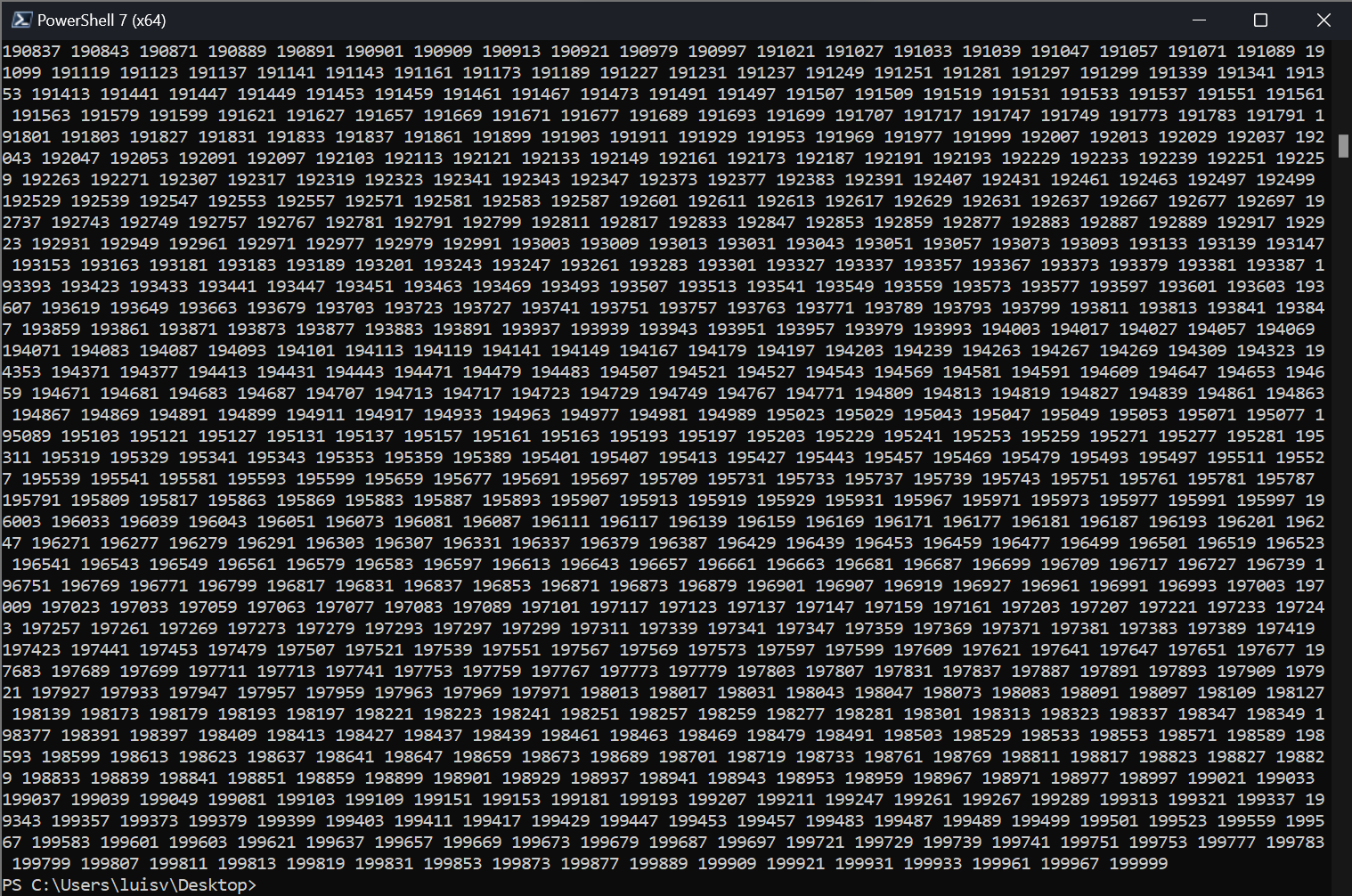
Luis Alberto Vargas González.

Practica 2.4 Programación en Go Lang.

**Programa 1.**

Impresión numeros primos con go rutinas; en este caso numeros que sean primos del rango del 1 al 200,000.





package main

import "fmt"

import "time"

func Primo(numero int){

var contador int

for i:=1;i<=numero;i++{

if numero % i ==0{

contador++

}

}

if contador==2 {

fmt.Print(numero," ")

}

}

func Buscar(inicio, fin int){

for i:=inicio; i<=fin; i++{

Primo(i)

}

}

func main () {

go Buscar(1,100000)

time.Sleep(100\* time.Second)

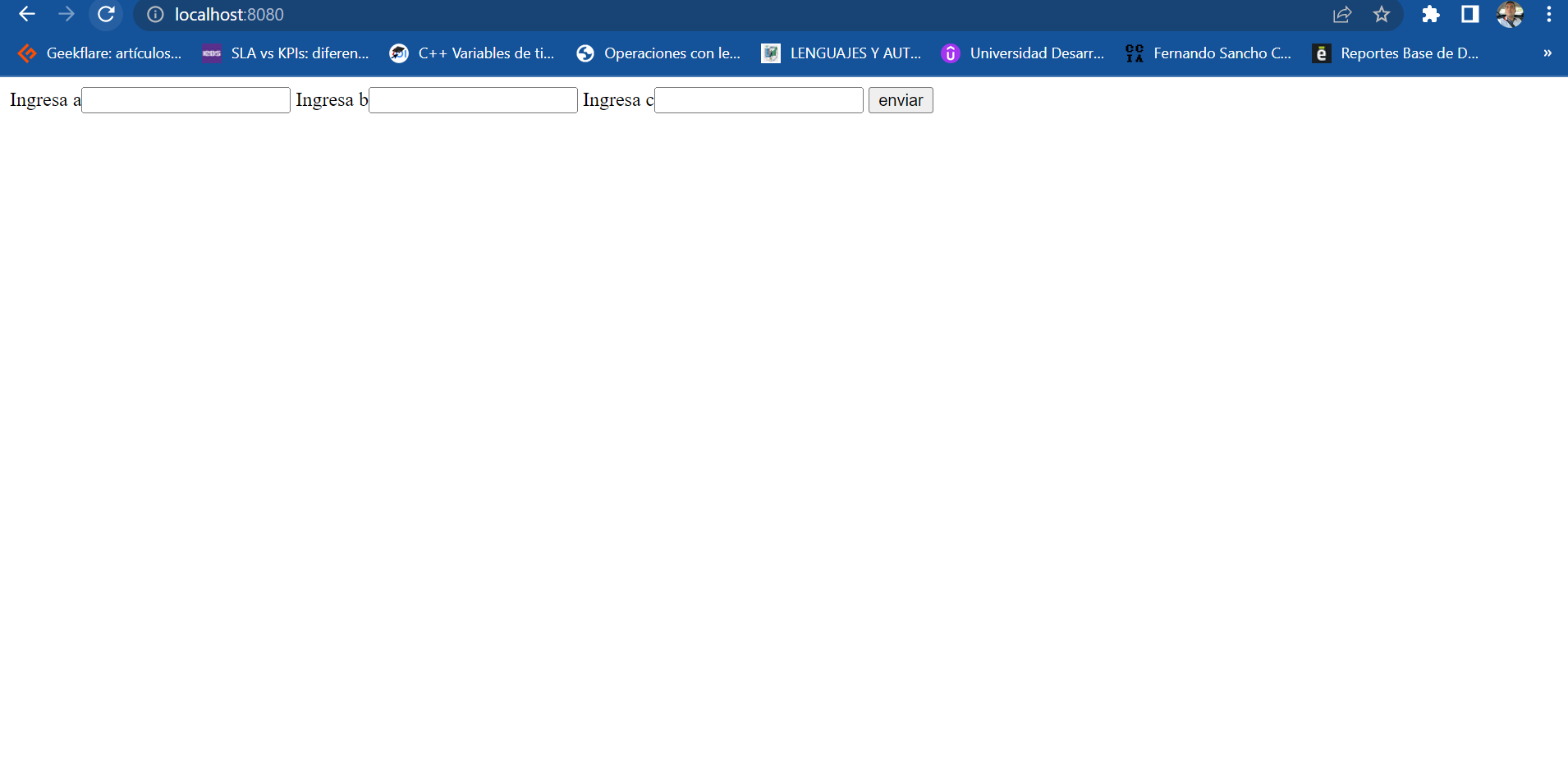
go Buscar(100001, 200000)

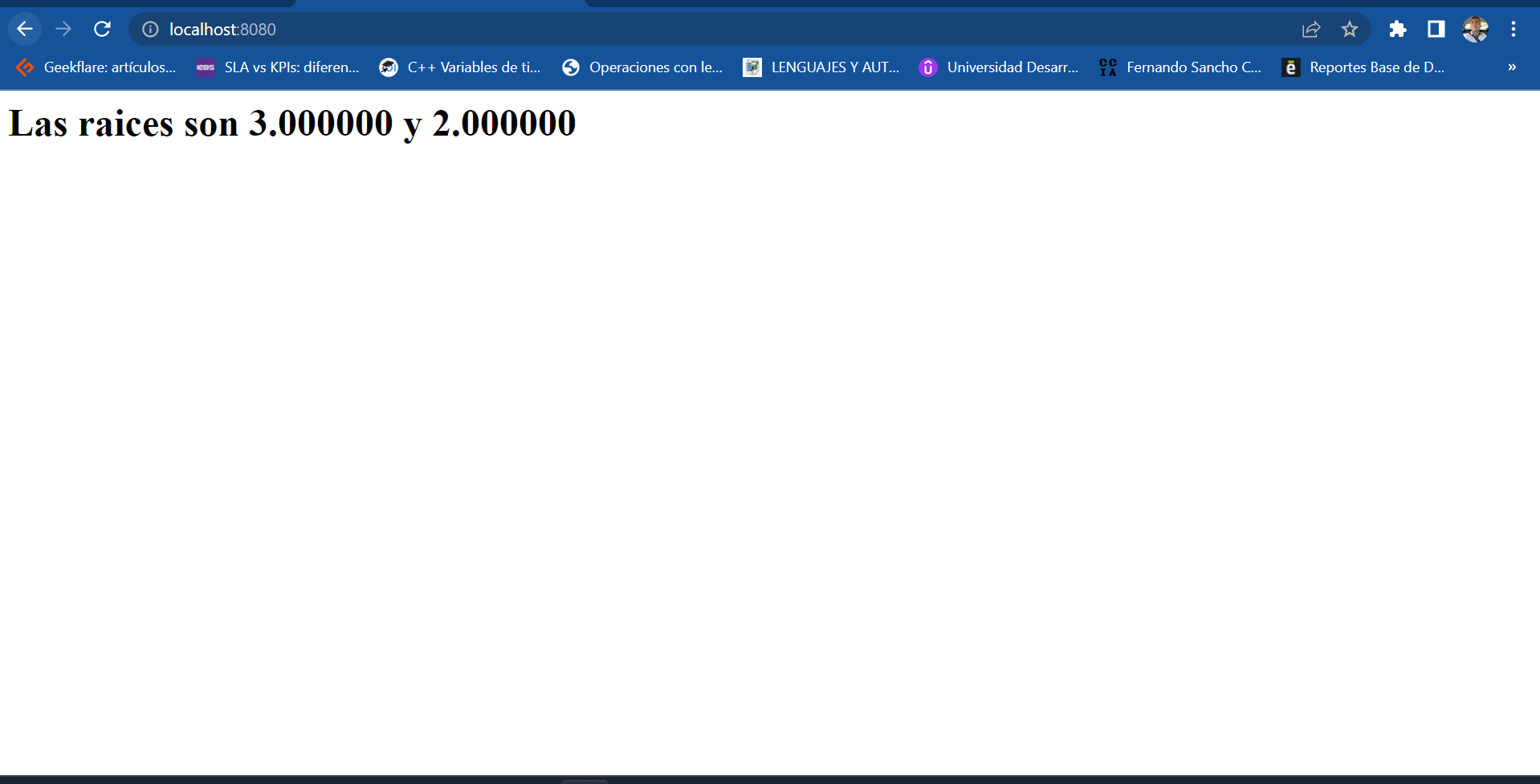
time.Sleep(100\* time.Second)

}

**Programa 2.**

Raíces con formula general ; 1, -5, 6= 3,2 respectivamente.





package main

import "fmt"

import "math"

import "html/template"

import "net/http"

import "strconv"

func chicharronera(w http.ResponseWriter, r \*http.Request){

fmt.Println("method:", r.Method)

if r.Method == "GET" {

t, \_ := template.ParseFiles("raices.html")

t.Execute(w, nil)

} else {

r.ParseForm()

a, err := strconv.ParseFloat(r.FormValue("a"),64)

if err != nil {

// handle error

}

b,err := strconv.ParseFloat(r.FormValue("b"),64)

if err != nil {

// handle error

}

c, err := strconv.ParseFloat(r.FormValue("c"),64)

if err != nil {

//handle error

}

var x1 float64

var x2 float64

x1=(-b+(math.Sqrt(b\*b-4\*a\*c)))/(2\*a);

x2=(-b-(math.Sqrt(b\*b-4\*a\*c)))/(2\*a);

fmt.Fprintf(w,"<h1>Las raices son %f y %f </h1>", x1,x2)

}

}

func main(){

http.HandleFunc("/", chicharronera)

http.ListenAndServe(":8080", nil)

}

----------------------------------HTML-----------

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Raices con formula general </title>

</head>

<body>

<div>

<form action="/" method="POST" >

<label>Ingresa a<input type="text" name="a" value=""></label>

<label>Ingresa b<input type="text" name="b" value=""></label>

<label>Ingresa c<input type="text" name="c" value=""></label>

<input type="submit" value="enviar">

</form>

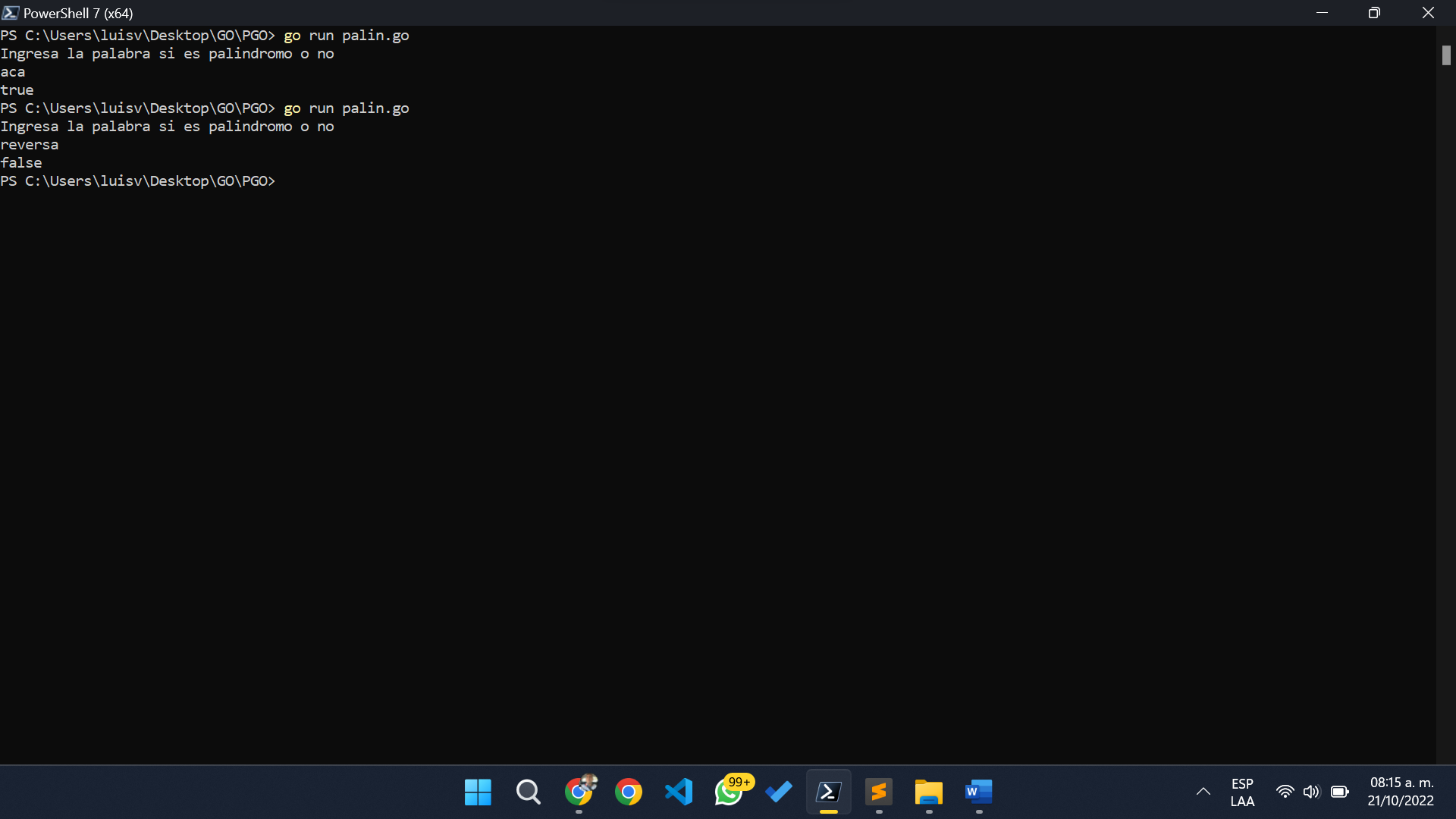
</div>

</body>

</html>

**Programa 3.**

Busca palabras que son Palíndromos ; en este caso la palabra “aca”.



package main

import "fmt"

func Palindromo(str string) bool {

reversedStr := ""

for i := len(str)-1; i >= 0; i-- {

reversedStr += string(str[i])

}

for i := range(str) {

if str[i] != reversedStr[i] {

return false

}

}

return true

}

func main () {

var a string

fmt.Println("Ingresa la palabra si es palindromo o no ")

fmt.Scan(&a)

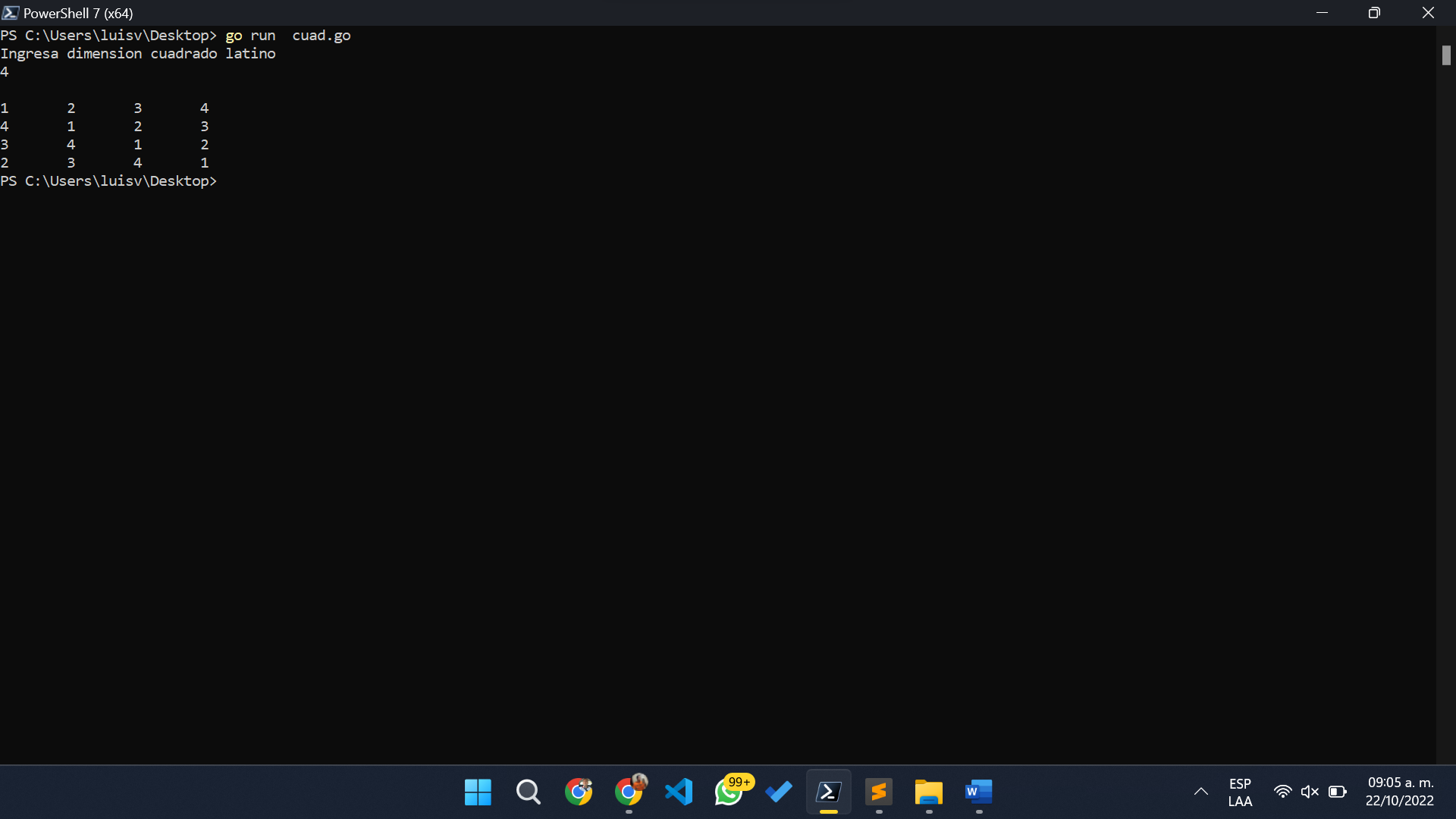
resultado := Palindromo(a)

fmt.Println(resultado)

}

**Programa 4.**

Imprime el cuadrado latino de orden n .



package main

import "fmt"

var m int

func main () {

var dim int

var k int

var p int

for {

fmt.Print("Ingresa dimension cuadrado latino")

fmt.Scan(&dim)

fmt.Print("\n")

if dim >=2 {

k= dim

p=0

m=0

break

}

}

for i :=1; i<=dim; i++{

if i==1{

for j:=1; j<=dim; j++{

fmt.Print(j, "\t")

}

}else{

for j:=k+1; j<=dim; j++{

fmt.Print(j, "\t")

}

for m := 1; m<=dim-p; m++{

fmt.Print(m,"\t")

}

}

k = k-1

p=p+1

fmt.Print("\n")

}

}

**Conclusiones:**

Como se pudo demostrar el hacer distintos programas que ejecuten distintas tareas, ayuda al programador en particular a poder ejercitar su lógica de programación, asi como a entrenar sus habilidades blandas de intercomunicación , de búsqueda e investigación, asi como de nuevas habilidades en el desarrollo de su perfil profesional.

Adicional a esto; se pudo constatar que el Lenguaje de programación GO es un lenguaje fuertemente tipado, hecho con robustez y que obliga al usuario del mismo a generar más y mejores prácticas de programación.