**S1Q1**

package main

import "fmt"

func main() {

    var choice, num1, num2 int

    fmt.Println("Choose an arithmetic operation:")

    fmt.Println("1. Addition (+)")

    fmt.Println("2. Subtraction (-)")

    fmt.Println("3. Multiplication (\*)")

    fmt.Println("4. Division (/)")

    fmt.Print("Enter your choice:")

    fmt.Scanln(&choice)

    fmt.Print("Enter the two numbers: ")

    fmt.Scanln(&num1, &num2)

    switch choice {

    case 1:

        fmt.Printf("%d + %d = %d\n", num1, num2, num1+num2)

    case 2:

        fmt.Printf("%d - %d = %d\n", num1, num2, num1-num2)

    case 3:

        fmt.Printf("%d \* %d = %d\n", num1, num2, num1\*num2)

    case 4:

        if num2 != 0 {

            fmt.Printf("%d / %d = %d\n", num1, num2, num1/num2)

        } else {

            fmt.Println("Cannot divide by zero!")

        }

    default:

        fmt.Println("Invalid choice")

    }

}

**S2Q1**

package main

import "fmt"

func main() {

    var f1, f2, f3, n int

    f1 = 1

    f2 = 2

    f3 = f1 + f2

    fmt.Print("Enter the number of terms to print:")

    fmt.Scanln(&n)

    fmt.Print("Fibonacci Series:")

    fmt.Print(f1, " ", f2)

    for i := 3; i <= n; i++ {

        fmt.Print(" ", f3)

        f1 = f2

        f2 = f3

        f3 = f1 + f2

    }

}

**S3Q1**

package main

import "fmt"

func isPalindrome(n int) int {

    var rev, rem, org int

    rev = 0

    org = n

    for n != 0 {

        rem = n % 10

        rev = rev\*10 + rem

        n /= 10

    }

    if org == rev {

        return 1

    } else {

        return 0

    }

}

func main() {

    var n, r int

    fmt.Print("Enter the number:")

    fmt.Scanln(&n)

    r = isPalindrome(n)

    if r == 1 {

        fmt.Printf("%d is Palidrome number.\n", n)

    } else {

        fmt.Printf("%d is not a Palindrome number.\n", n)

    }

}

**S4Q1**

package main

import "fmt"

func main() {

    var input int

    fmt.Print("Enter a number:")

    fmt.Scanln(&input)

    sum := sumDig(input)

    fmt.Println("Sum of digits:", sum)

}

func sumDig(n int) int {

    if n < 10 {

        return n

    }

    return n%10 + sumDig(n/10)

}

**S5Q1**

package main

import (

    "fmt"

    "os"

)

func main() {

    text := "Hello, this is NIRAJ BENDSURE.\nThis is our LAST PRACTICAL."

    file, err := os.Create("sample.txt")

    if err != nil {

        fmt.Println("Error creating file:", err)

        return

    }

    defer file.Close()

    \_, err = file.WriteString(text)

    if err != nil {

        fmt.Println("Error writing to file:", err)

        return

    }

    fmt.Println("Text file created successfully!")

}

**S6Q2**

package main

import "fmt"

func CopyArray(org []int, dst []int) {

    for i := 0; i < len(org); i++ {

        dst[i] = org[i]

    }

}

func main() {

    org := []int{1, 2, 3, 4, 5, 7, 6, 9}

    dst := make([]int, len(org))

    CopyArray(org, dst)

    fmt.Println("Original Array:", org)

    fmt.Println("Copied Array:", dst)

}

**S7Q2**

package main

import "fmt"

type student struct {

    name, class  string

    age   int

    per float32

}

func (s \*student) show() {

    fmt.Printf("Name: %s\n", s.name)

    fmt.Printf("Age: %d\n", s.age)

    fmt.Printf("Class: %s\n", s.class)

    fmt.Printf("Percentage: %.2f\n", s.per)

}

func main() {

    s := &student{

        name:  "NSB",

        age:   21,

        class: "TY",

        per: 88.99,

    }

    s.show()

}

**S8Q1**

package main

import "fmt"

type book struct {

    bid     int

    bname   string

    bauthor string

    price   float32

}

func main() {

    var b [10]book

    var n int

    fmt.Print("Enter the number of books: ")

    fmt.Scan(&n)

    for i := 0; i < n; i++ {

        fmt.Print("\nEnter book ID: ")

        fmt.Scan(&b[i].bid)

        fmt.Print("Enter book name: ")

        fmt.Scan(&b[i].bname)

        fmt.Print("Enter book author: ")

        fmt.Scan(&b[i].bauthor)

        fmt.Print("Enter book price: ")

        fmt.Scan(&b[i].price)

    }

    fmt.Println("\n\nBook Information")

    fmt.Println("\nBOOK ID  TITLE       AUTHOR      PRICE")

    fmt.Println("---------------------------------------")

    for i := 0; i < n; i++ {

        fmt.Println(b[i].bid, "\t\t", b[i].bname, "\t\t", b[i].bauthor, "\t\t", b[i].price)

    }

}

**S9Q1**

package main

import "fmt"

func isPalindrome(n int) int {

    var rev, rem, org int

    rev = 0

    org = n

    for(n != 0) {

        rem = n % 10

        rev = rev \* 10 + rem

        n /= 10

    }

    if (org == rev) {

        return 1

    } else {

        return 0

    }

}

func main(){

    var n, r int

    fmt.Print("Enter the number: ")

    fmt.Scanln(&n)

    r = isPalindrome(n)

    if (r == 1) {

        fmt.Printf("%d is a Palidrome number.\n", n)

    } else {

        fmt.Printf("%d is not a Palindrome number.\n", n)

    }

 }

**S10Q2**

package main

import "fmt"

func fibonacci(n int, c chan int) {

    a, b := 0, 1

    for i := 0; i < n; i++ {

        c <- a

        a, b = b, a+b

    }

    close(c)

}

func main() {

    var n int

    fmt.Print("Enter the number of fibonacci series: ")

    fmt.Scan(&n)

    fmt.Printf("Fibonacci Series of %d numbers:\n", n)

    c := make(chan int)

    go fibonacci(n, c)

    for num := range c {

        fmt.Println(num)

    }

}

**S11Q1**

package main

import "fmt"

func main() {

    var num int

    fmt.Print("Enter a number: ")

    fmt.Scanf("%d", &num)

    if num >= 10 && num <= 99 {

        fmt.Println(num, "is a two-digit number.")

    } else {

        fmt.Println(num, "is not a two-digit number.")

    }

}

**S12Q1**

package main

import (

    "fmt"

    "log"

    "os"

)

func main() {

    if len(os.Args) < 2 {

        fmt.Println("Usage: go run main.go <filename>")

        return

    }

    filename := os.Args[1]

    file, err := os.Stat(filename)

    if err != nil {

        log.Fatal(err)

    }

    fmt.Println("File Name:", file.Name())

    fmt.Println("Size (in bytes):", file.Size())

    fmt.Println("Permissions:", file.Mode())

    fmt.Println("Last Modified:", file.ModTime())

}

**S13Q1**

package main

import "fmt"

func main() {

    evenSum := 0

    oddSum := 0

    for i := 1; i <= 100; i++ {

        if i%2 == 0 {

            evenSum += i

        } else {

            oddSum += i

        }

    }

    fmt.Println("Sum of even numbers between 1 to 100:", evenSum)

    fmt.Println("Sum of odd numbers between 1 to 100:", oddSum)

}

**S14Q1**

package main

import "fmt"

func main() {

    slice := []int{1, 2, 3, 4, 5}

    fmt.Println("Original Slice:", slice)

    slice = append(slice, 6)

    fmt.Println("After Append:", slice)

    rem := 2

    slice = append(slice[:rem], slice[rem+1:]...)

    fmt.Println("After Remove:", slice)

    copys := make([]int, len(slice))

    copy(copys, slice)

    fmt.Println("Copied Slice:", copys)

}

**S15Q1**

package main

import "fmt"

func addSub(a, b int) (int, int, int) {

    sum := a + b

    diff := a - b

    prod := a \* b

    return sum, diff, prod

}

func main() {

    var n1, n2 int

    fmt.Print("Enter the two numbers to add, subtract and multiply:")

    fmt.Scanln(&n1, &n2)

    sum, diff, prod := addSub(n1, n2)

    fmt.Printf("Sum: %d\n", sum)

    fmt.Printf("Difference: %d\n", diff)

    fmt.Printf("Product: %d\n", prod)

}

**S16Q1**

package main

import (

    rectangle "GOLang/rect"

    "fmt"

)

func main() {

    var l, b int

    fmt.Print("Enter the length and width of rectangle: ")

    fmt.Scanln(&l, &b)

    area := rectangle.Area(l, b)

    fmt.Printf("Area of rectangle with length %d and width %d is %.d\n", l, b, area)

}

**S17Q2**

package main

import (

    "fmt"

    "os"

)

func main() {

    file, err := os.OpenFile("example.txt", os.O\_APPEND|os.O\_WRONLY|os.O\_CREATE, 0644)

    if err != nil {

        fmt.Println("Error:", err)

        return

    }

    defer file.Close()

    content := "Helllo this is NSB from TYBCA"

    if \_, err := file.WriteString(content); err != nil {

        fmt.Println("Error:", err)

        return

    }

    fmt.Println("Content appended successfully.")

}

**S18Q1**

package main

import "fmt"

func multi(num int) {

    for i := 1; i <= 10; i++ {

        fmt.Printf("%d x %d = %d\n", num, i, num\*i)

    }

}

func main() {

    var n int

    fmt.Print("Enter the number to print multiplication table: ")

    fmt.Scanln(&n)

    multi(n)

}

**S19Q1**

package main

import "fmt"

func addSub(a, b int) (int, int) {

    sum := a + b

    diff := a - b

    return sum, diff

}

func main() {

    var n1, n2 int

    fmt.Print("Enter the two numbers to add and subtract:")

    fmt.Scanln(&n1, &n2)

    sum, diff := addSub(n1, n2)

    fmt.Printf("Sum: %d\n", sum)

    fmt.Printf("Difference: %d\n", diff)

}

**S20Q1**

package main

import (

    "fmt"

    "os"

)

func main() {

    file, err := os.OpenFile("example.txt", os.O\_APPEND|os.O\_WRONLY|os.O\_CREATE, 0644)

    if err != nil {

        fmt.Println("Error:", err)

        return

    }

    defer file.Close()

    content := "Helllo this is NSB from TYBCA"

    if \_, err := file.WriteString(content); err != nil {

        fmt.Println("Error:", err)

        return

    }

    fmt.Println("Content appended successfully.")

}