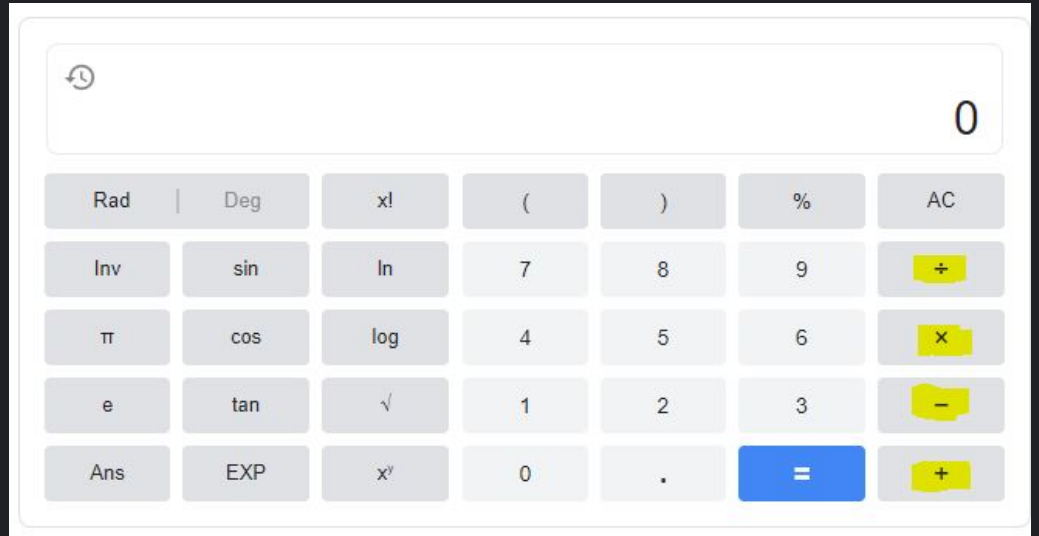


# Exercício 02

Lucas Alcantara e Victor Nicéas

# Definir o objetivo da avaliação

- Cliente-Servidor TCP e UDP
- Tempo de Resposta da Calc.



# Listar os serviços do sistema

```
package impl

import "shared"

type Calculadora struct{}

func (Calculadora) InvocaCalculadora(req shared.Request) int {
    var r int

    (...)

    func (Calculadora) Add(x int, y int) int {
        return x + y
    }

    func (Calculadora) Sub(x int, y int) int {
        return x - y
    }

    func (Calculadora) Mul(x int, y int) int {
        return x * y
    }

    func (Calculadora) Div(x int, y int) int {
        return x / y
    }
}
```

# Listar os serviços do sistema TCP

```
// Create request
request := shared.Request{Op: __, P1: i, P2: i}
// Serialise and send request
err = encoder.Encode(request)
if err != nil {
    fmt.Println(err)
    os.Exit(0)
}
// Receive response from server
err = decoder.Decode(&response)
if err != nil {
    fmt.Println(err)
    os.Exit(0)
}
```

# Listar os serviços do sistema UDP

```
var json = jsoniter.ConfigCompatibleWithStandardLibrary

n, addr, err = conn.ReadFromUDP(msgFromClient)
//Deserialize request from clientserver
err := json.Unmarshal(msgFromClient[:n], &request)
(...)
msgToClient, err = json.Marshal(rep)
(...)
_, err = conn.WriteTo(msgToClient, addr)
```

# Escolher as métricas de desempenho

- Tempo médio de resposta e desvio padrão, tanto no UDP quanto TCP. Assim como tempo total de execução.

# Escolher fatores

Fatores: Máquina local com um nível médio, serialização tem um nível dependendo do tamanho da mensagem e qual encoder. Por exemplo, o encoder escolhido foi o jsoniter pro udp e tivemos uma melhora considerável de desempenho.

# Carga de trabalho

Vamos invocar a requisição 10.000 vezes tanto no UDP quanto TCP, fazendo N (número de clientes) sendo 2, 5 e 10.



# Projetando o experimento

Iremos fazer o experimento com uma carga pequena no tamanho da informação a ser enviada e o processamento do dado. Depois, aumentaremos isso para verificar a influência de cada fator aumentando o nível.

# Analisar e interpretar os dados

Faremos a média do tempo de resposta e seu desvio padrão.

