

[t e c h a c e s s . a l g]

Angela Akemi Hiroishi - Gabriel Miranda - Geovana da Costa Andrade - Guilherme Henrique Moura Neves - Guilherme Henrique Santos Rodrigues

O algoritmo descrito tem por objetivo demonstrar/apresentar de forma caricata as funções de login e cadastro de usuários no aplicativo de acessibilidade desenvolvido por este grupo/startup. Como determinado, o código traz apenas algo abstrato que exemplifique a usabilidade das funções. Foram levantados os requisitos de funcionamento do sistema e reunidas as funções iniciais de UI – controle do usuário, como segue:

1. Menu inicial
2. Painel de login
3. Painel de cadastro

1. A tela/menu inicial traz as opções disponíveis para escolha do usuário:

<1> procedimento login
<2> procedimento cadastro
<0> sair / fechar

A decisão é tratada e a decisão tomada, informando caso o número/opção digitada não possa ser reconhecida ou não exista

2. O painel de login possibilita a entrada de usuário/senha por meio da entrada de dados, após inseridos os dados são conferidos com o vetor de memória dos dados cadastrados, caso exista e seja válido o usuário/senha informado, retorna ao menu principal mostrando que o usuário <nome> está autenticado. Caso o vetor de memória esteja em branco (inicializado porém nenhum usuário cadastrado), redireciona ao procedimento de cadastro, após cadastro retorna ao login.

3. Painel de cadastro fornece a opção de registro em memória de até 10 entradas usuário/senha em vetor de memória – iniciado nulo a princípio.

Nos anexos seguintes é possível identificar casos de testes e demonstração das funcionalidades:

```
C:\ Console simulando o modo texto do MS-DOS

1 - entrar
2 - registrar
0 - sair
informe a opcao desejada: <0-2>
1

nenhum usuario cadastrado na base
favor cadastrar:

[cadastrar]
informe um usuario:
user
informe uma senha:
123
usuario <user> cadastrado com sucesso!
aperte <enter> para retornar

[login]
informe o usuario:
user
informe a senha:
123
usuario autenticado: user
aperte <enter> para retornar

C:\ Console simulando o modo texto do MS-DOS

usuario: user
1 - entrar
2 - registrar
0 - sair
informe a opcao desejada: <0-2>
```

GLOBAL	TUSER	C	"user"
GLOBAL	TPW	C	"123"
GLOBAL	USER[0]	C	"user"
GLOBAL	USER[1]	C	"0"
GLOBAL	USER[2]	C	"0"
GLOBAL	USER[3]	C	"0"
GLOBAL	USER[4]	C	"0"
GLOBAL	USER[5]	C	"0"
GLOBAL	USER[6]	C	"0"
GLOBAL	USER[7]	C	"0"
GLOBAL	USER[8]	C	"0"
GLOBAL	USER[9]	C	"0"
GLOBAL	USER[10]	C	"0"
GLOBAL	PW[0]	C	"123"
GLOBAL	PW[1]	C	"0"
GLOBAL	PW[2]	C	"0"
GLOBAL	PW[3]	C	"0"

Usuário tenta logar sem cadastrar, efetua cadastro com os dados: <user> <123>

Após, tela inicial informando o usuário autenticado.

Ao lado direito, tela da memória com os dados salvos

```
C:\ Console simulando o modo texto do MS-DOS

1 - entrar
2 - registrar
0 - sair
informe a opcao desejada: <0-2>
2

[cadastrar]
informe um usuario:
user
informe uma senha:
123
usuario <user> cadastrado com sucesso!
aperte <enter> para retornar
```

Usuário efetua cadastro com os dados: <user> <123> Dados são salvos na memória.

```
C:\ Console simulando o modo texto do MS-DOS

usuario: user
1 - entrar
2 - registrar
0 - sair
informe a opcao desejada: <0-2>
10
opcao invalida!
aperte <enter> para retornar
```

Usuário informa uma opção inválida <10>

```
C:\> Console simulando o modo texto do MS-DOS

usuario: user
1 - entrar
2 - registrar
0 - sair
informe a opcao desejada: <0-2>
1

[login]
informe o usuario:
user
informe a senha:
111
usuario ou senha incorretos!
aperte <enter> para retornar
```

Usuário já cadastrado tenta efetuar login com dados incorretos.

Memória <user> <123>

Informado <user> <111>

Resultado: Login não autorizado,
procedimento reinicia.

```
procedimento login
início

    aut<-falso

    escreval()

    se (user[0]<>"0") então
    senão
        escreval("nenhum usuario cadastrado na base")
        escreval("favor cadastrar:")
        cadastrar
    fimse

    escreval("[login]")
    escreval("informe o usuario:")
    leia(tuser)
    escreval("informe a senha:")
    leia(tpw)

    para i de 0 ate 10 faça
        se (tpw=pw[i]) e (tuser=user[i]) então
            aut<-verdadeiro
        fimse
    fimpara

    se (aut=falso) então
        escreval("usuario ou senha incorretos!")
        escreval("aperte <enter> para retornar")
        leia(r)
        login
    senão
        escreval("usuario autenticado: ",tuser)
        escreval("aperte <enter> para retornar")
        leia(r)
    fimse

fimprocedimento
```

```

procedimento cadastrar
  inicio

    t<-falso

    escreval()

    escreval("[cadastrar]")
    escreval("informe um usuario:")
    leia(tuser)

    para i de 0 ate 10 faça
      se (tuser=user[i]) entao
        escreval("usuario ja cadastrado: ", tuser)
        interrompa
      fimse
    fimpara

    escreval("informe uma senha:")
    leia(tpw)

    i2<-0
    enquanto (t=falso) faça
      se (user[i2]="0") entao
        user[i2]<-tuser
        pw[i2]<-tpw
        t<-verdadeiro
      fimse
      se (i2+1=11) entao
        escreval("limite de usuarios atingido")
        interrompa
      fimse
      i2<-i2+1
    fimenquanto

    escreval("usuario <",tuser,> cadastrado com sucesso!")
    escreval("aperte <enter> para retornar")
    leia(r)

  fimprocedimento

```

inicio

```
loop<-verdadeiro

init

escreval()

enquanto (loop=verdadeiro) faca

    limpatela

    se (tuser="") entao
    senao
        escreval("usuario: ",tuser)
    fimse

    escreval("1 - entrar")
    escreval("2 - registrar")
    escreval("0 - sair")

    escreval("informe a opcao desejada: <0-2>")
    leia(r)

    escolha r
    caso 0
        escreval("..finalizando")
        loop<-falso
    caso 1
        login
    caso 2
        cadastrar
    outrocaso
        escreval("opcao invalida!")
        escreval("aperte <enter> para retornar")
        leia(r)
    fimsecolha

fimenquanto
```

fimalgoritmo