



UNIVERSIDADE da MADEIRA

Arquitetura de Computadores

Projeto 2 – Pizzaria

Docentes:

- Dionísio Barros
- Pedro Camacho
- Nuno Ferreira
- Sofia Isabel Silva Inácio

Alunos:

- Juan Silva, nº 2082516
- Joel Camacho, nº 2083516

Índice

Índice.....	2
Introdução.....	3
Resultados.....	4
<i>Login</i>	4
Menu de Cliente	4
Menu de Pizzas	4
Menu de Tamanho	4
Menu de Pagamento	4
Bibliografia	5
Conclusão.....	6
Anexo A – Fluxogramas Gerais	7
Anexo B – Código em Linguagem <i>Assembly</i>	13

Introdução

Neste projeto foi desenvolvido um programa em *assembly* que permite controlar os pedidos de comida online de uma pizzeria que faz entregas. A interface gráfica (ou display) simula a página web onde os clientes efetuam os seus pedidos online.

Resultados

Ao iniciar a simulação teremos um menu onde permite-nos escolher entre fazer um *login* ou fazer um novo registo.

Login

Ao efetuar o *login* o cliente deverá introduzir um *username* e uma *password*, caso estes dados não correspondam com algum dos dados na base de dados é apresentada uma mensagem de erro. Caso os dados introduzidos nos periféricos coincidam com aos dados na base de dados o cliente é redirecionado para o menu de cliente.

Menu de Cliente

No menu de cliente são apresentadas duas escolhas, uma para efetuar uma encomenda e outra para fazer *logout*, caso o cliente escolha fazer *logout* é redirecionado para o menu inicial, caso escolha encomendar uma pizza é apresentado um menu com cinco pizzas.

Menu de Pizzas

No menu de pizzas estão disponíveis cinco pizzas para escolha, em qualquer uma das cinco escolhas o cliente é redirecionado para o menu de tamanho.

Menu de Tamanho

No menu de tamanho o cliente pode escolher se pretende uma pizza de tamanho grande, com o custo de oito euros, ou se pretende uma pizza de tamanho pequeno, com o custo de cinco euros. Após a escolha do tamanho o cliente é redirecionado para o menu de pagamento.

Menu de Pagamento

Neste menu pretendia-se mostrar ao cliente o valor total gasto na pizzeria e, caso este valor seja maior que cinquenta euros, mostrar que teve desconto de dois euros caso seja uma pizza pequena, ou desconto de quatro euros no caso de ter encomendado uma pizza grande. Infelizmente, este menu não foi bem sucedido.

Bibliografia

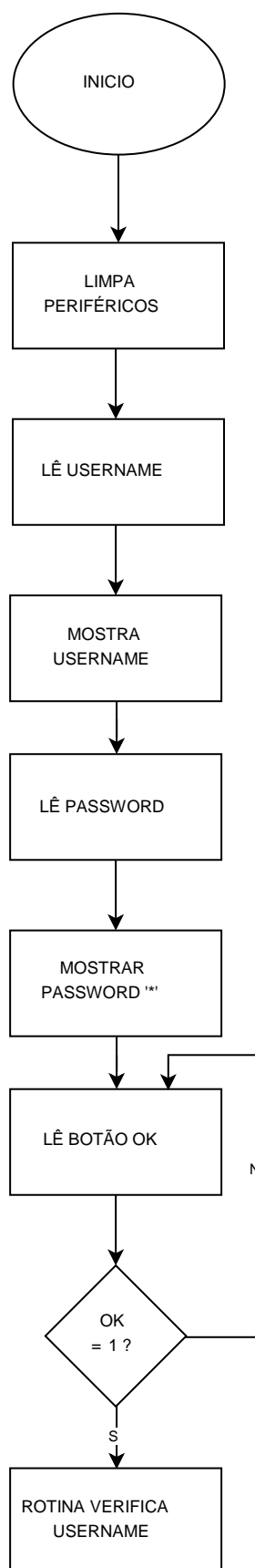
J. Delgado e C. Ribeiro, Arquitetura de Computadores, FCA, 2007.

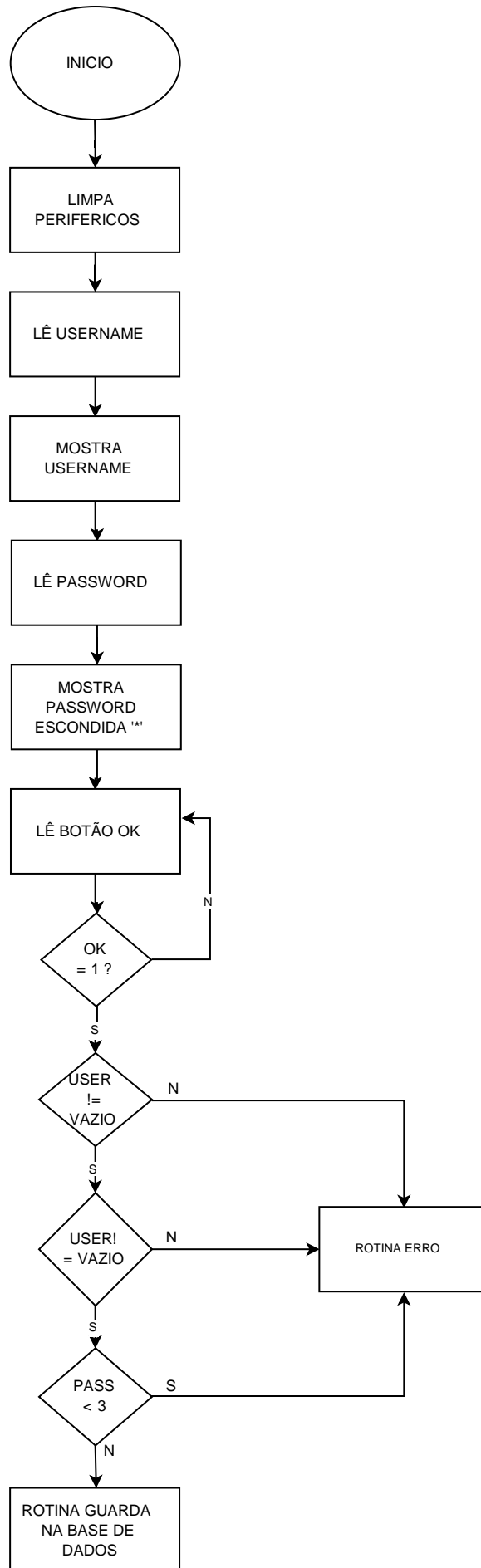
Conclusão

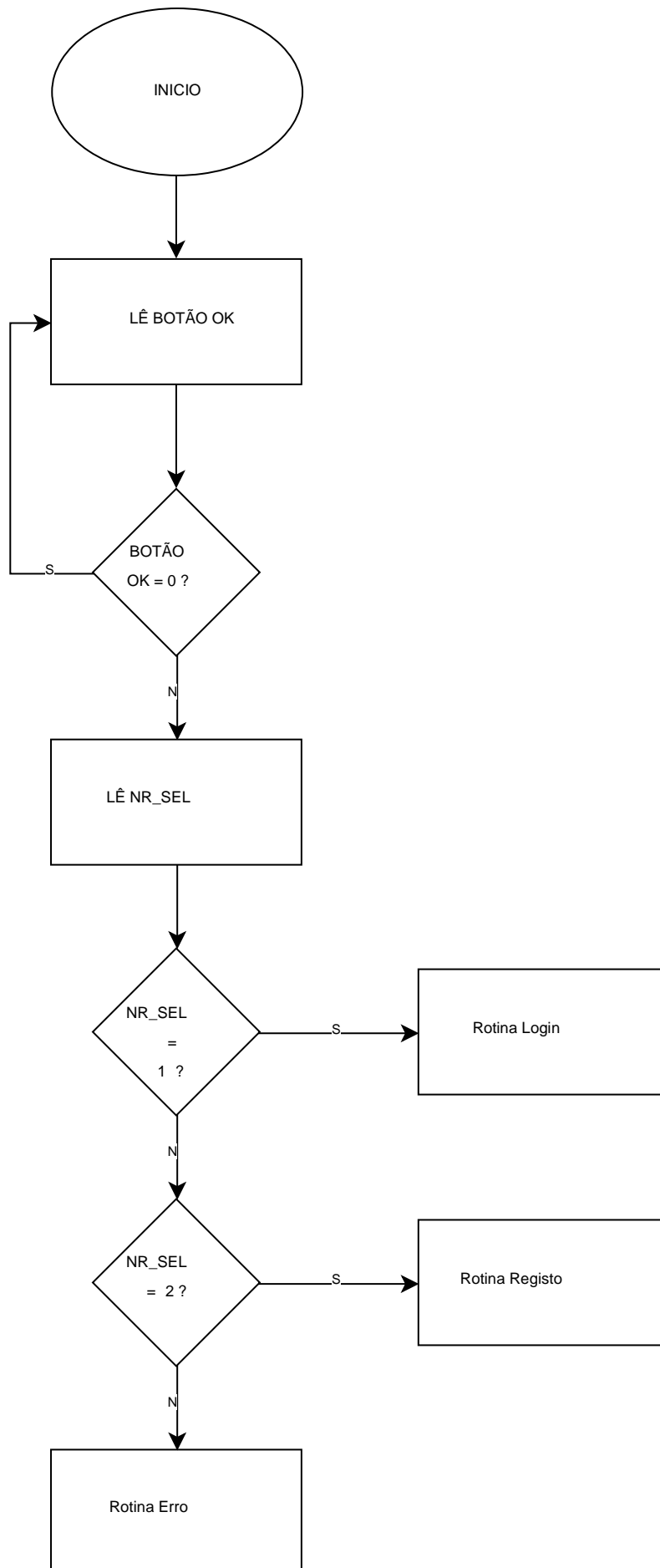
Infelizmente o projeto não está totalmente funcional, mas a maior parte deste funciona como esperado. Devido à falta de tempo, não nos foi possível terminar o pagamento e o desconto, mas o restante funciona devidamente.

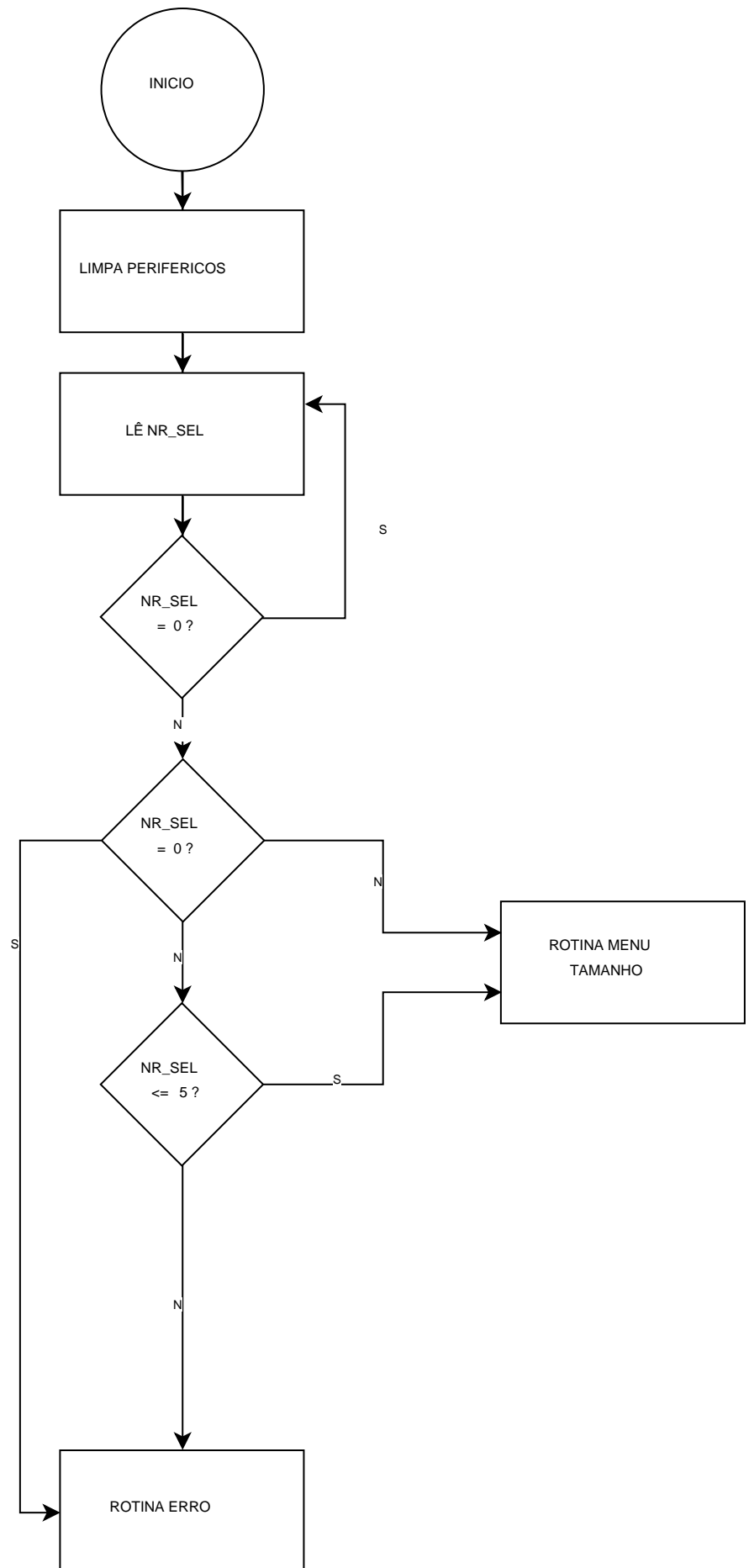
Com este projeto concluiu-se que é possível, e complexo, criar aplicações ou programas apenas em linguagem *assembly* pois esta é de baixo nível, concluiu-se também que o projeto foi gratificante pois adquirimos conhecimentos da linguagem *assembly*.

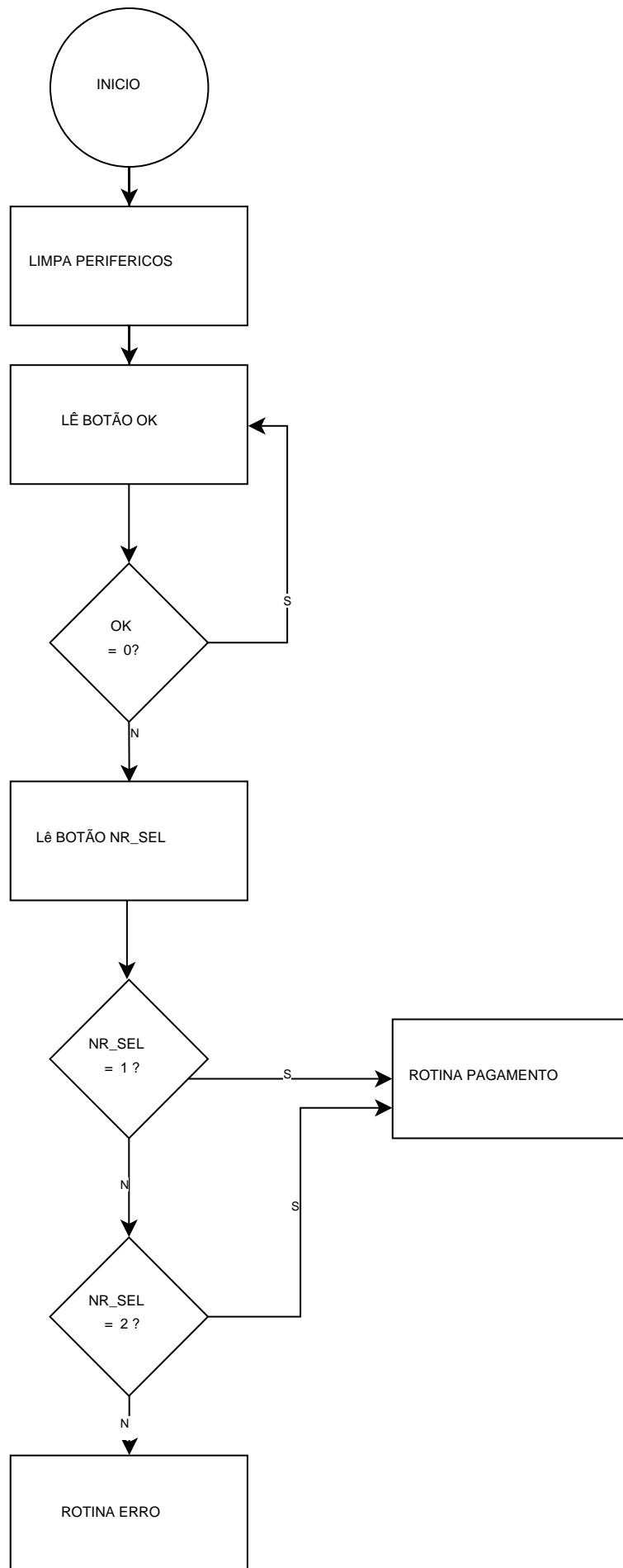
Anexo A – Fluxogramas Gerais

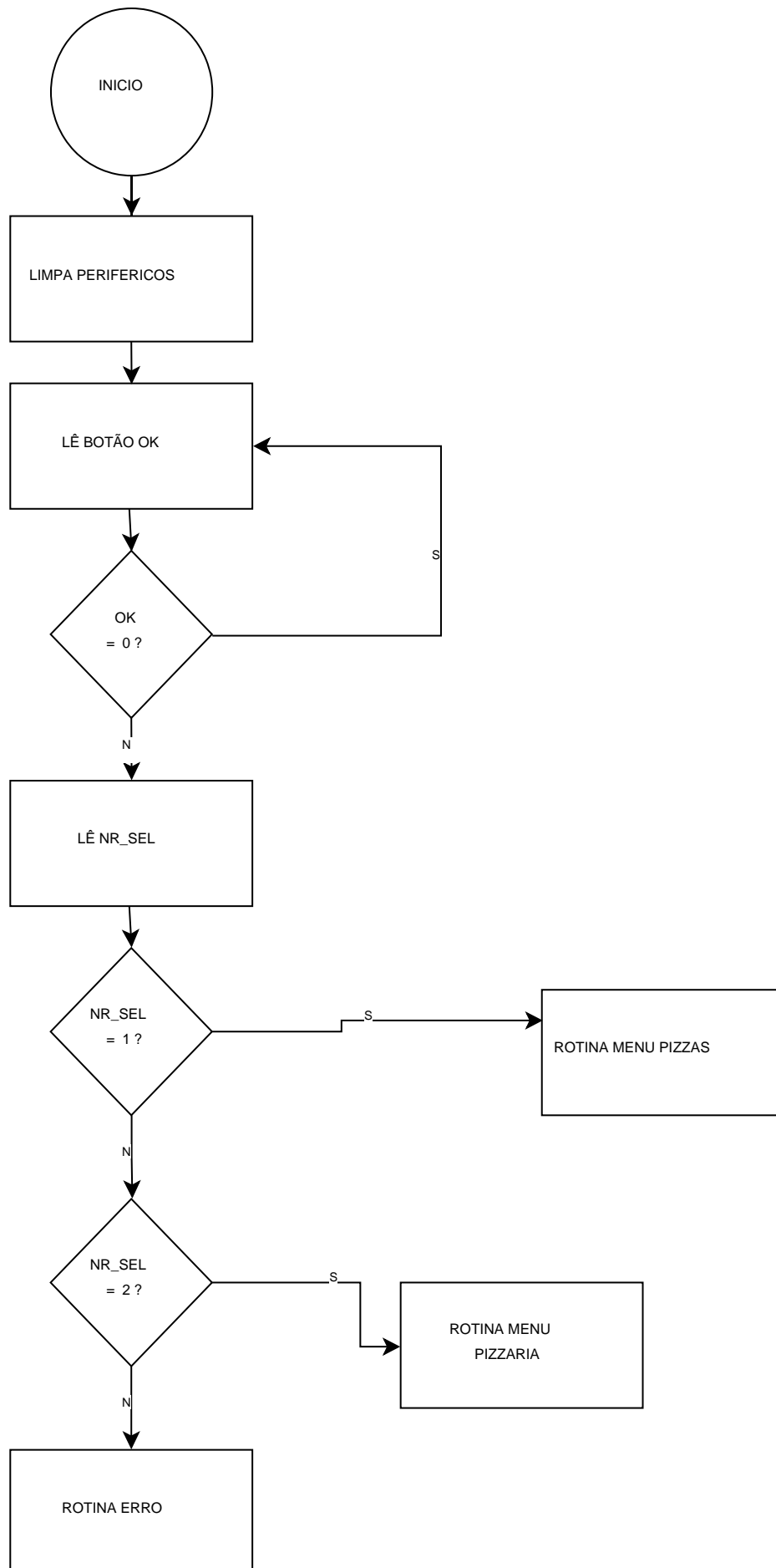












[illegible]

Place 2000H ;A PARTIR DO ENDEREÇO 2000 DA MEMORIA VAI ESTAR O MENU ABAIXO

MenuPizzaria:

```
String " MENU PIZZARIA "
String "
String "1. LOGIN "
String "2. REGISTAR "
String "
String "
String "
```

Place 2080H

MenuLogin:

```
String " MENU LOGIN "
String "
String " USERNAME: "
String "
String " PASSWORD: "
String "
String "
```

Place 2160H

MenuRegisto:

```
String " MENU REGISTO "
String "
String " USERNAME: "
String "
String " PASSWORD: "
String "
String "
```

Place 2240H

MenuCliente:

```
String " MENU CLIENTE "
String "
String "1. PIZZAS "
String "2. SAIR "
String "
String "
String "
```

Place 2320H

MenuPizzas:

```
String " MENU PIZZAS "
String "
String "1. PIZZA 1 "
String "2. PIZZA 2 "
String "3. PIZZA 3 "
String "4. PIZZA 4 "
String "5. PIZZA 5 "
```

Place 2400H

MenuTamanho:

```
String " MENU TAMANHO "
String "
String "1. PEQUENA 5 EUR"
String "2. GRANDE 8 EUR "
String "
String "
String "
```

Place 2480H

MenuErro:

```
String " ATENCAO: "
```



```

    PUSH R2
    PUSH R3
    PUSH R4
    PUSH R5
    PUSH R6
    PUSH R7
    CALL LimpaPerifericos ;LIMPA PERIFÉRICOS
    MOV R2, MenuLogin ;COLOCA MENU DE LOGIN
    CALL MostraDisplay ;MOSTRA NO DISPLAY O MENU

login_Le_ok:
    CALL RotinaUser ;COLOCA O USERNAME DO PERIFERICO NO DISPLAY
    CHARACTER A CHARACTER
    CALL RotinaPass ;COLOCA A PASSWORD DO PERIFERICO NO DISPLAY
    CHARACTER A CHARACTER, CONVERTENDO PARA '*'
    MOV R0, Ok
    MOVB R1, [R0]
    CMP R1, 0
    JEQ login_Le_ok ;CICLO ATÉ O OK PASSAR A '1'

VerificarUserLogin:
    MOV R1, BasedeDados ;INICIALIZA REGISTO DE BASE DA 1ª FICHA
    MOV R2, numClientes ;NÚMERO DE FICHAS DE UTILIZADORES DA
    PIZZARIA
    MOV R3, tam ;NÚMERO DE BYTES DE CADA FICHA DE UTILIZADOR
    MOV R4, Username ;USERNAME DO PERIFERICO
    MOV R7, Nr_Caracteres ;NUMERO DE CARACTERES A VERIFICAR
    MOV R6, BasedeDados ;CÓPIA DO REGISTO DE BASE DA 1ª FICHA

login_verifica_prox_caracter_user:
    MOVB R5, [R4] ;CARACTER DO USERNAME DO PERIFERICO
    MOVB R0, [R1] ;CARACTER DO USERNAME DA BASE DE DADOS DE
    DETERMINADA FICHA
    CMP R5, R0 ;COMPARA CHARACTER DE AMBOS
    JNE login_procura_basededados_next ;PROXIMA FICHA DE CLIENTE
    ADD R4, 1 ;PASSA A APONTAR PARA O ENDEREÇO DO PROXIMO
    CHARACTER DO USERNAME DO PERIFERICO
    ADD R1, 1 ;PASSA A APONTAR PARA O ENDEREÇO DO PROXIMO
    CHARACTER DO USERNAME DA BASE DE DADOS DE DETERMINADA FICHA
    SUB R7, 1 ;MENOS UM CHARACTER A VERIFICAR
    CMP R7, 0 ;CHEGOU AO FIM DO USERNAME (8 caracteres)
    JEQ VerificarPassLogin ;TODOS OS CARACTERES IGUAIS, USERNAME
    EXISTE, VERIFICA A PASS
    JMP login_verifica_prox_caracter_user ;CICLO PARA VERIFICAR
    CHARACTER A CHARACTER

login_procura_basededados_next:
    ADD R6, R3 ;PASSA PARA A PROXIMA FICHA DE CLIENTE
    MOV R1, R6 ;COLOCA A APONTAR PARA A PROXIMA FICHA DE CLIENTE
    MOV R7, Nr_Caracteres ;REPOE NUMERO DE CARACTERES A VERIFICAR

    MOV R4, Username ;USERNAME DO PERIFERICO
    SUB R2, 1 ;MENOS UM CLIENTE A VERIFICAR
    CMP R2, 0 ;CHEGOU AO FIM DOS CLIENTES?
    JNE login_verifica_prox_caracter_user ;NAO, PASSA VOLTA A
    COMPARAR OS CARACTERES

```



```

        CALL RotinaPass ;COLOCA A PASSWORD DO PERIFERICO NO
DISPLAY CHARACTER A CHARACTER, CONVERTENDO PARA '*'
        MOV R0, Ok
        MOVB R1, [R0]
        CMP R1, 0
        JEQ registo_Le_ok ;CICLO ATÉ O OK PASSAR A '1'

RotinaVerificarUserRegisto:
        MOV R1, BasedeDados ;INICIALIZA REGISTO DE BASE DA 1ª FICHA
        MOV R2, numClientes ;NÚMERO DE FICHAS DE UTILIZADORES DA
PIZZARIA
        MOV R3, tam ;NÚMERO DE BYTES DE CADA FICHA DE UTILIZADOR
        MOV R4, Username ;USERNAME DO PERIFERICO
        MOV R7, Nr_Caracteres
        MOV R6, R1 ;COPIA DO REGISTO DE BASE DA 1ª FICHA

registo_verifica_prox_caracter_user:
        MOVB R5, [R4] ;CARACTER DO USERNAME DO PERIFERICO
        MOVB R0, [R1] ;CARACTER DO USERNAME DA BASE DE DADOS DE
DETERMINADA FICHA
        CMP R5, R0 ;COMPARA CHARACTER DE AMBOS
        JNE registo_procura_basededados_next ;VERIFICA PROXIMO
CLIENTE
        ADD R4, 1 ;PASSA A APONTAR PARA O ENDEREÇO DO PROXIMO
CARACTER DO USERNAME DO PERIFERICO
        ADD R1, 1 ;PASSA A APONTAR PARA O ENDEREÇO DO PROXIMO
CARACTER DO USERNAME DA BASE DE DADOS DE DETERMINADA FICHA
        SUB R7, 1 ;MENOS UM CHARACTER A VERIFICAR
        CMP R7, 0
        JEQ registo_user_em_uso ;USERNAME TOTALMENTE IGUAL A BASE DE
DADOS, LOGO ESTA EM USO
        JMP registo_verifica_prox_caracter_user ;CICLO PARA VERIFICAR
SE HA UM USERNAME IGUAL

registo_user_em_uso:
        MOV R2, MenuUserEmUso
        CALL RotinaErro ;COLOCA NO DISPLAY A MENSAGEM DE UTILIZADOR
EM USO

registo_procura_basededados_next:
        MOV R4, Username ;VOLTAR A COLOCAR A APONTAR PARA O INICIO DO
USERNAME DO PERIFERICO
        MOV R7, Nr_Caracteres ;REPOE
        ADD R6, R3 ;PROXIMA FICHA DA BASE DE DADOS
        MOV R1, R6 ;COLOCA EM R1 PARA AVALIAR EM CIMA
        SUB R2, 1 ;MENOS UM CLIENTE A VERIFICAR
        CMP R2, 0
        JNE registo_verifica_prox_caracter_user ;CICLO DE VERIFICAR
CARACTER A CHARACTER

;VERIFICA PASSWORD
MOV R1, Password ;PASSWORD DO PERIFERICO
MOV R2, 0 ;CONTADOR DE CARACTERES

registo_verifica_prox_caracter_pass:
        MOVB R0, [R1] ;CARACTER DA PASSWORD DO PERIFERICO
        CMP R0, Terminador

```

```

JEQ registo_pass_invalida
ADD R2, 1 ;INCREMENTA CONTADOR DE CARACTERES
ADD R1, 1 ;PASSA A APONTAR PARA O PROXIMO CARACTER
CMP R2, 3 ;VERIFICA SE A PASSWORD TEM 3 CARACTERES
JNE registo_verifica_prox_caracter_pass ;CICLO PARA VERIFICAR
SE TEM PELO MENOS 3 CARACTERES

```

AdicionaBasedeDados:

```

MOV R1, Username ;VOLTA A COLOCAR A APONTAR PARA O INICIO DO
USERNAME DO PERIFERICO
MOV R3, BasedeDados ;COLOCA NOVAMENTE NO INICIO DA BASE DE
DADOS
MOV R7, BasedeDados ;COPIA DO INICIO DA BASE DE DADOS
MOV R6, tam ;TAMANHO DE CADA FICHA

```

proxima_ficha_basededados:

```

MOVB R5, [R3]
CMP R5, Terminador ;VERIFICA SE ESTA FICHA NAO ESTA OCUPADA
JEQ add_prox_caracter_user ;ESTÁ LIVRE, PODE PREENCHER
ADD R3, R6 ;PASSA PARA A PROXIMA FICHA
ADD R7, R6 ;COPIA PARA USAR NO REGISTO
JMP proxima_ficha_basededados

```

add_prox_caracter_user:

```

MOVB R0, [R1]
CMP R0, Terminador ;VERIFICA SE O USERNAME ACABOU
JEQ AdicionaPassBasedeDados ;SE ACABOU O USERNAME, PASSA PARA
A PASSWORD
MOVB [R3], R0 ;COLOCA O PRIMEIRO CARACTER NA BASE DE DADOS
ADD R1, 1 ;PASSA AO PROXIMO CARACTER DO PERIFERICO
ADD R3, 1 ;PASSA A PROXIMA POSIÇÃO DA BASE DE DADOS
JMP add_prox_caracter_user

```

registo_pass_invalida:

```

MOV R2, MenuPassErrada ;PASSWORD INVALIDA
CALL RotinaErro ;APRESENTA MENSAGEM DE PASSWORD INVALIDA

```

AdicionaPassBasedeDados:

```

MOV R1, Password ;VOLTA A COLOCAR A APONTAR PARA O INICIO DA
PASSWORD DO PERIFERICO
MOV R3, bd_pass ;COLOCA EM R3 O INDICE DA PASSWORD DA BASE DE
DADOS
ADD R3, R7 ;COLOCA EM R3 O INDICE DA PASSWORD DA BASE DE
DADOS DA FICHA DE CLIENTE PRETENDIDA

```

add_prox_caracter_pass:

```

MOVB R0, [R1]
CMP R0, Terminador ;VERIFICA SE A PASS ACABOU
JEQ AdicionaSucces ;SE ACABOU A PASS, MENSAGEM DE
SUCESSO
MOVB [R3], R0 ;COLOCA O PRIMEIRO CARACTER DA PASS NA
BASE DE DADOS
ADD R1, 1 ;PASSA AO PROXIMO CARACTER DA PASS
ADD R3, 1 ;PASSA A PROXIMA POSIÇÃO DA PASS DA BASE DE
DADOS
JMP add_prox_caracter_pass ;CICLO DE INTRODUIR A
PASSWORD NA BASE DE DADOS

```


[illegible]

