

Faculdade de Ciências Exatas e da Engenharia

2019/2020

Arquitetura de Computadores

Licenciatura em Engenharia Informática

2º Projeto – Posto de Carregamento Elétrico



Trabalho realizado por:

Diego Briceño (nº 2043818)

Rúben Rodrigues (nº 2046018)

Docentes:

Dionísio Barros, Nuno Ferreira

Sofia Inácio, Pedro Camacho

Índice

1. Introdução.....	3
2. Objetivos.....	3
3. Desenvolvimento.....	3
3.1. Periféricos de Entrada	3
3.1.1. Ligar.....	3
3.1.2. Botão OK	3
3.1.3. ID do cliente.....	3
3.1.4. Código de Segurança do cliente.....	3
3.1.5. Saldo do cliente.....	3
3.1.6. Bateria do Veículo do Cliente.....	4
3.1.7. Opções escolhidas	4
3.1.8. Tempo de Carregamento escolhido	4
3.1.9. Carregamento do Posto	4
3.2. Utilização do Programa.....	4
4. Discussão de Resultados.....	5
4.1. Criação de um Cliente	5
4.2. Alteração de dados	6
4.3. Carregamento do Posto	6
4.4. Carregamento da bateria do veículo.....	6
5. Conclusão	7
6. Bibliografia.....	7
7. Anexo A.....	8
7.1. Fluxograma do Programa Geral	8
7.2. Fluxograma da Alteração da Base de Dados.....	9
7.3. Fluxograma da Verificação do Utilizador.....	10
7.4. Fluxograma do Calcula Tempo	11
7.5. Fluxograma do Descarregamento do Posto.....	12
7.6. Fluxograma da Verificação do Saldo	13
7.7. Fluxograma do Fornecimento de Energia	14
7.8. Fluxograma do Débito.....	15
7.9. Fluxograma da Atualização do Posto.....	16
8. Anexo B	17

1. Introdução

Este relatório apresentará os objetivos relacionados ao segundo trabalho prático da unidade curricular de Arquitetura de Computadores, assim como o seu desenvolvimento, discussão de resultados e a conclusão a que os alunos chegaram.

A linguagem de programação utilizada para o desenvolvimento do trabalho foi o Assembly. Esta linguagem é específica de cada processador e permite codificar uma longa lista de valores de sinais num só número, o opcode. O assembler converte o programa assembly em números binários, permitindo que o processador consiga executar as instruções diretamente.

O programa desenvolvido foi criado em linguagem Assembly do processador PEPE e, para testá-lo utilizou-se o simulador JAVA.

2. Objetivos

O objetivo deste trabalho é criar um programa que simula um posto de carregamento para veículos elétricos, em linguagem Assembly, como também especificar e elaborar fluxogramas.

O posto de carregamento só está disponível aos detentores de um cartão fornecido pelos serviços da Unidade.

O processo de abastecimento é realizado pelos condutores, sendo selecionado o tipo de carregamento e o tempo de carregamento desejado e, de seguida, é efetuado o pagamento do carregamento. O posto de carregamento tem a possibilidade de inserção de novos utilizadores, como também da alteração de dados de detentores do cartão universitário.

A interface de comando do posto de carregamento foi feita através de um display de dimensões 7x16 (7 linhas de 16 caracteres – bytes), que funciona como periférico de saída.

3. Desenvolvimento

3.1. Periféricos de Entrada

3.1.1. Ligar

Este botão é responsável por ligar e desligar a máquina do posto que permite o carregamento dos veículos. Os utilizadores só podem utilizar o posto quando o periférico **Ligar** estiver diferente de 0. Caso esteja a “0”, a máquina do posto é desligada. Quando a máquina está desligada o display fica em branco até o utilizador a ligar.

3.1.2. Botão OK

Este botão é responsável pela confirmação das opções do utilizador. As opções inseridas pelo utilizador só serão válidas após o utilizador ter colocado o valor “1” no periférico **OK**. O utilizador só poderá avançar quando as suas opções forem validadas.

Caso seja inserido uma opção incorreta e o utilizador confirmar a opção, irá aparecer no display a informação de que a opção é inválida e o utilizador terá de inserir uma nova opção.

3.1.3. ID do cliente

O periférico **InputID** é onde o utilizador insere o ID do cliente que pretende realizar um carregamento, alterar os seus dados ou do novo cliente que se pretende adicionar à base de dados.

3.1.4. Código de Segurança do cliente

O periférico **InputCodSeguranca** é onde o utilizador insere o código de segurança do cliente que pretende realizar o carregamento, alterar os seus dados ou do novo cliente que se pretende adicionar à base de dados.

3.1.5. Saldo do cliente

O periférico **InputSaldo** é onde, caso o cliente já esteja na base de dados, é inserido o novo valor do saldo do cliente escolhido, ou, o valor do saldo do cliente que será criado.

Se o valor inserido pelo utilizador for inferior a 0, é mostrado no display a informação de que o saldo inserido é inválido pois não pode ser negativo.

3.1.6. Bateria do Veículo do Cliente

O periférico *InputBateria* é onde, caso o cliente já esteja na base de dados, é inserido o novo valor da bateria do veículo do cliente escolhido, ou, caso não esteja, o valor da bateria do veículo do cliente que será criado.

Caso o valor inserido pelo utilizador for inferior a 0 ou superior a 100, é mostrado no display que o valor inserido é inválido e que deve ser inserido um valor entre 0 e 100.

3.1.7. Opções escolhidas

O periférico *InputOpcao* é onde o utilizar irá inserir a opção escolhida para os diferentes menus.

Se o utilizador escolher uma opção diferente das opções que aparecem no display (e for confirmada), é informado de que a opção escolhida é inválida.

3.1.8. Tempo de Carregamento escolhido

O periférico *InputTempo* é onde o utilizador insere a duração do carregamento que pretende realizar.

Caso o tempo seja inferior a 1, é mostrado no display que o tempo inserido tem de ser superior a 0.

E, caso o tempo escolhido seja superior ao tempo necessário para a bateria do veículo ficar completamente carregada, e não descarregue o posto, o utilizador é informado no display e, é realizado o fornecimento de energia apenas durante o tempo necessário para a bateria do veículo ficar 100% carregada, sendo feito apenas o débito do tempo que realmente foi necessário para efetuar o carregamento.

Por exemplo, se o utilizador escolher o tipo de carregamento rápido e uma duração de 3 horas, será apenas feito o débito de 3€ e não 9€, pois é apenas necessária uma hora para que a bateria do veículo fique completamente carregada.

3.1.9. Carregamento do Posto

O periférico *InputIncrementoBateria* é onde o utilizador insere a quantidade de energia que pretende carregar na bateria do posto escolhida anteriormente. O carregamento da bateria do posto só é realizado se o valor inserido pelo utilizador for superior a 0. Se inserir um valor que cause overflow, o utilizador é informado e a bateria fica com o valor que tinha antes da adição.

3.2. Utilização do Programa

Inicialmente a máquina encontra-se desligada. Após ser ligada, são lidos os endereços referentes às baterias do posto para guardar os valores em registos, evitando inúmeros acessos à memória (R0 – Normal, R1 – Semi-rápido e R2 – Rápido), e, de seguida, é iniciado o programa.

Para facilitar os cálculos, a bateria dos carros dos clientes está em percentagem, mas a bateria dos postos não, cada percentagem que se carrega no carro é uma unidade que retiramos ao valor da bateria do posto correspondente.

3.2.1. Criar/Alterar Dados

No início do programa é perguntado ao utilizador se pretende alterar ou criar dados na base de dados.

```
BASE DE DADOS
Pretende fazer
alteracoes na
base de dados?
1 - Sim
2 - Nao
OK - continuar
```

Figura 1 Início do programa

```
BASE DE DADOS
Pretende criar
um cliente ou
alterar um?
1 - Criar
2 - Alterar Dados
OK - continuar
```

Figura 2 Alterar ou Criar dados

Para alterar dados, é pedido ao utilizador para inserir o ID e código de segurança do cliente cujo utilizador pretende alterar os dados. Sendo depois pedido para ser inserido o novo ID, código de segurança, saldo e valor da bateria do veículo do cliente, caso este exista na base de dados.

Para criar dados, é pedido apenas para inserir os dados referentes ao novo cliente, sendo verificado se o ID inserido é igual ao ID de um cliente já existente na base de dados, se tal acontecer, o cliente não é criado e o utilizador é informado. Se o utilizador não pretender fazer nenhuma alteração, avança-se para a próxima secção.

3.2.2. Inserção de Energia

Para inserir energia no posto, o utilizador deve escolher uma das 3 baterias existentes no posto, cada uma referente a um tipo de carregamento diferente. Após a escolha, é feita a inserção do valor de energia que se pretende carregar na bateria desejada, com as limitações indicadas anteriormente no periférico referente ao carregamento do posto.

Se o utilizador não pretender inserir energia no posto, é mostrado no display a informação das baterias do posto que estão funcionais. O utilizador só poderá realizar um carregamento de um tipo de carregamento que se encontre funcional. Se os 3 postos não se encontrarem funcionais, o utilizador é informado que precisa carregar um posto e volta-se ao início do programa.

```
CARREGAR POSTO
Escolha bateria:
1) - Normal
2) - Semi - Rápido
3) - Rápido
4) - Avançar
OK - continuar
```

Figura 3 Escolha da bateria do posto

```
CARREGAR POSTO
Quanto quer
carregar?

OK - continuar
```

Figura 4 Escolha energia a carregar

```
ESTADO DO POSTO
Normal: Funcional
Semi Rápido:
Funcional
Rápido:
Funcional
OK - continuar
```

Figura 5 Estado das baterias do posto

3.2.3. Carregamento do veículo

Para realizar um carregamento, é pedido ao utilizador para inserir o ID e código de segurança do cliente que pretende realizar o carregamento. Após a verificação do cliente, caso seja autorizado, é pedido para escolher o tipo de carregamento desejado, e de seguida, a duração do mesmo. Depois, é mostrado no display o tipo de carregamento escolhido e a duração do fornecimento de energia que será feito (tendo em conta os diferentes casos possíveis mencionados anteriormente). Se o utilizador continuar, irá visualizar no display o saldo que terá no final do carregamento, tal como o custo da operação.

```
ESCOLHA O TIPO.
DE CARREGAMENTO:
1 - Normal
2 - Semi - Rápido
3 - Rápido

OK - continuar
```

Figura 6 Escolha do tipo de carregamento

```
CARREGAMENTO
Por quanto tempo
quer carregar o
seu carro?

OK - continuar
```

Figura 7 Escolha do tempo de carregamento

Após ser feito o débito, é realizada a atualização da bateria do posto referente ao tipo de carregamento escolhido pelo utilizador, sendo subtraído o valor da energia fornecida ao veículo à respetiva bateria, e aparece no display uma informação de que o carregamento foi concluído. Gravam-se os valores das baterias na memória e volta-se ao início do programa.

4. Discussão de Resultados

4.1. Criação de um Cliente

Para exemplificar a criação de um cliente, criou-se um cliente com o ID igual a 258 (decimal, 0102 em hexadecimal), código de segurança 772 (decimal, 0304 em hexadecimal), saldo 15€ e com a bateria do veículo a 0%.

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
10E0	A	87	1	F8	A	FF	0	0	0	0	0	0	0	0	0	0
10F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
1100	89	C2	29	4	F	44	0	0	7D	A3	B7	17	2	64	0	0
1110	C8	99	DF	37	E	DE	0	0	0	0	0	0	0	0	0	0

Figura 8 Base de Dados antes da criação

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
10E0	A	87	1	F8	A	FF	0	0	0	0	0	0	0	0	0	0
10F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
1100	89	C2	29	4	F	44	0	0	7D	A3	B7	17	2	64	0	0
1110	C8	99	DF	37	E	DE	0	0	0	0	0	0	0	0	0	0

Figura 9 Base de Dados depois da criação

Como se pode ver na Figura 8, a base de dados tem três clientes e após a criação do novo cliente, que é inserido no fim da base de dados, o tamanho da base de dados aumentou por 1 (Figura 9).

4.2. Alteração de dados

Para exemplificar esta parte do programa, decidiu-se alterar os dados do cliente criado anteriormente. Alterou-se o seu ID e código de segurança (0A0B e 0C0D, respetivamente), reduzimos o seu saldo para 9€ e “carregou-se” o seu veículo a 10%. Pode-se observar isto na *Figura 10*.

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
10E0	A	87	1	F8	A	FF	0	0	0	0	0	0	0	0	0	0
10F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
1100	89	C2	29	4	F	44	0	0	7D	A3	B7	17	2	64	0	0
1110	C8	99	DF	37	E	DE	0	A	B	C	D	0	9	0	A	

Figura 10 Base de Dados após a alteração

4.3. Carregamento do Posto

Para demonstrar o carregamento do posto, assumimos a situação inicial da *Figura 11*. Foi feito um carregamento da bateria do posto rápido no valor de FF, como se pode ver na *Figura 13*. Após avançarmos para a demonstração dos níveis de energia, pode-se verificar a alteração do valor na *Figura 12*.

PC	603C	R0	0A87
R1	01F8	R2	0AFF

Figura 11 Valores das baterias antes do carregamento

PC	603E	R0	0A87
R1	01F8	R2	0BFE

Figura 12 Valores das baterias após o carregamento

0020	20	43	41	52	52	45	47	41	52	20	50	4F	53	54	4F	20	CARREGAR POSTO
0030	20	20	51	75	61	6E	74	6F	20	71	75	65	72	20	20	20	Quanto quer
0040	20	20	20	63	61	72	72	65	67	61	72	3F	20	20	20	20	carregar?
0050	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0060	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0070	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0080	20	4F	4B	20	2D	20	63	6F	6E	74	69	6E	75	61	72	20	OK - continuar
0090	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00A0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figura 13 Escolha da energia a carregar

4.4. Carregamento da bateria do veículo

Para demonstrar o processo de verificação de um cliente e carregamento do seu veículo, tendo como base de dados a da *Figura 8*, na verificação introduziu-se o ID e código de segurança do primeiro cliente (*Figura 14*) e foi autorizado para carregar o seu veículo (*Figura 15*).

0020	20	20	56	45	52	49	46	49	43	41	43	41	4F	20	20	20	VERIFICACAO
0030	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0040	20	20	49	4E	54	52	4F	44	55	5A	41	20	49	44	20	20	INTRODUZA ID
0050	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0060	20	45	20	43	4F	44	49	47	4F	20	53	45	47	2E	20	20	E CODIGO SEG.
0070	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0080	20	4F	4B	20	2D	20	63	6F	6E	74	69	6E	75	61	72	20	OK - continuar
0090	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00A0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00C0	89	C2	29	04	0	0	0	0	0	0	0	0	0	0	0	0	

Figura 14 Introdução de dados do cliente

AUTORIZADO
COM SUCESSO

OK - continuar

Figura 15 Cliente autorizado

Para carregar o veículo, optou-se por um carregamento do tipo normal com duração de 10 horas (*Figura 16*). Como o carregamento normal carrega 20% da bateria do veículo por hora, só seriam necessárias 5 horas para carregar o veículo, por isso foi apresentado ao utilizador a mensagem da *Figura 17*. Na *Figura 18* especificou-se o tipo de carregamento e a duração deste. Por fim, para debitar o valor ao cliente, só foi debitado o custo de 5€ (no tipo normal cada hora é 1€) e foi apresentado ao cliente o custo do carregamento e o seu saldo atualizado após o débito (*Figura 19*).

0020	20	20	43	41	52	45	47	41	4D	45	4E	54	4F	20	20	20	CARREGAMENTO
0030	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0040	50	6F	72	20	71	75	61	6E	74	6F	20	74	65	6D	70	6F	Por quanto tempo
0050	71	75	65	72	20	63	61	72	72	65	67	61	72	20	6F	20	quer carregar o
0060	20	20	20	73	65	75	20	63	61	72	72	6F	3F	20	20	20	seu carro?
0070	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	
0080	20	4F	4B	20	2D	20	63	6F	6E	74	69	6E	75	61	72	20	OK - continuar
0090	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00A0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
00C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Figura 16 Escolha do tempo de carregamento

ATENCAO
O tempo
selecionado e
superior ao
necessario para
carregar bateria
OK - continuar

Figura 17 Informação sobre o tempo escolhido

```

CARREGAMENTO

Tipo:
Normal
Duracao:
5 horas
OK - continuar

```

Figura 18 Informação sobre o carregamento

```

CARREGAMENTO

Saldo:
3903 euros
Custo:
5 euros
OK - continuar

```

Figura 19 Informação sobre o saldo e custo

Verificando o estado da base de dados antes e após o carregamento pode-se observar que o veículo do cliente está carregado a 100% e que o saldo do mesmo diminuiu por 5€. Como após um carregamento os valores das baterias dos postos em memória são atualizados também se pode observar que a bateria do posto normal diminuiu por 100 unidades, equivalente aos 100% fornecidos.

Addr...	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
10E0	A	87	1	F8	A	FF	0	0	0	0	0	0	0	0	0	0
10F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
1100	89	C2	29	4	F	44	0	0	7D	A3	B7	17	2	64	0	0
1110	C8	99	DF	37	E	DE	0	0	0	0	0	0	0	0	0	0

Figura 20 Base de dados antes do carregamento

10E0	A	23	1	F8	B	FE	0	0	0	0	0	0	0	0	0	0
10F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
1100	89	C2	29	4	F	3F	0	64	7D	A3	B7	17	2	64	0	0
1110	C8	99	DF	37	E	DE	0	0	0	0	0	0	0	0	0	0

Figura 21 Base de dados após o carregamento

Se o cliente autorizado não tivesse saldo suficiente para efetuar o carregamento que desejava, seria informado como na Figura 22 e o seu veículo não seria carregado.

```

ATENCAO

SALDO
INSUFICIENTE

OK - continuar

```

Figura 22 Informação sobre saldo insuficiente

5. Conclusão

Concluindo, consideramos que os objetivos do trabalho foram atingidos e este tornou-se mais fácil quando se dividiu o problema em partes mais pequenas e mais acessíveis. A elaboração prévia dos fluxogramas, em relação ao desenvolvimento da solução, ajudou a facilitar o dito desenvolvimento.

A linguagem Assembly, embora seja mais complicada de perceber do que uma linguagem de alto nível, é mais fácil para codificar um dispositivo do que o código máquina.

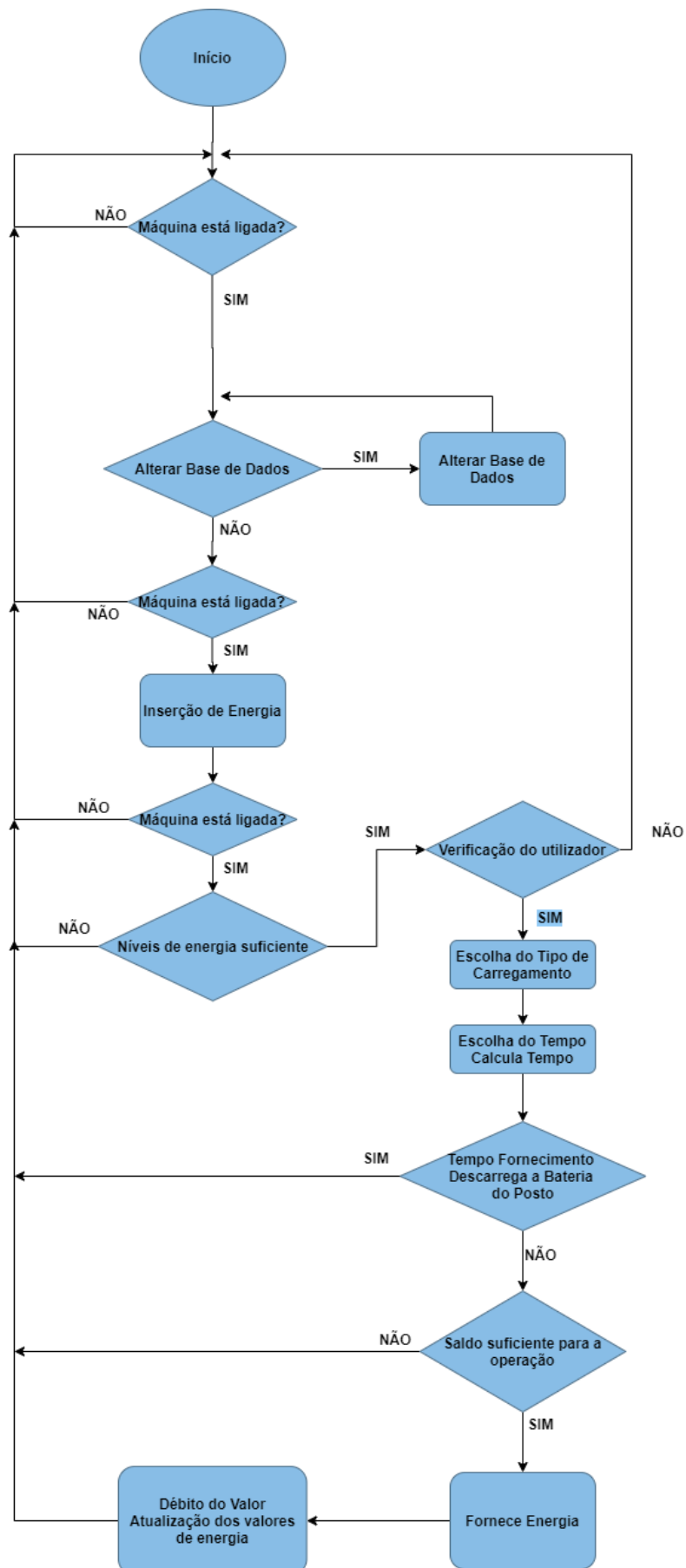
Em geral, este trabalho foi uma boa ferramenta de aprendizagem para uma melhor compreensão da arquitetura de computadores e o funcionamento destes.

6. Bibliografia

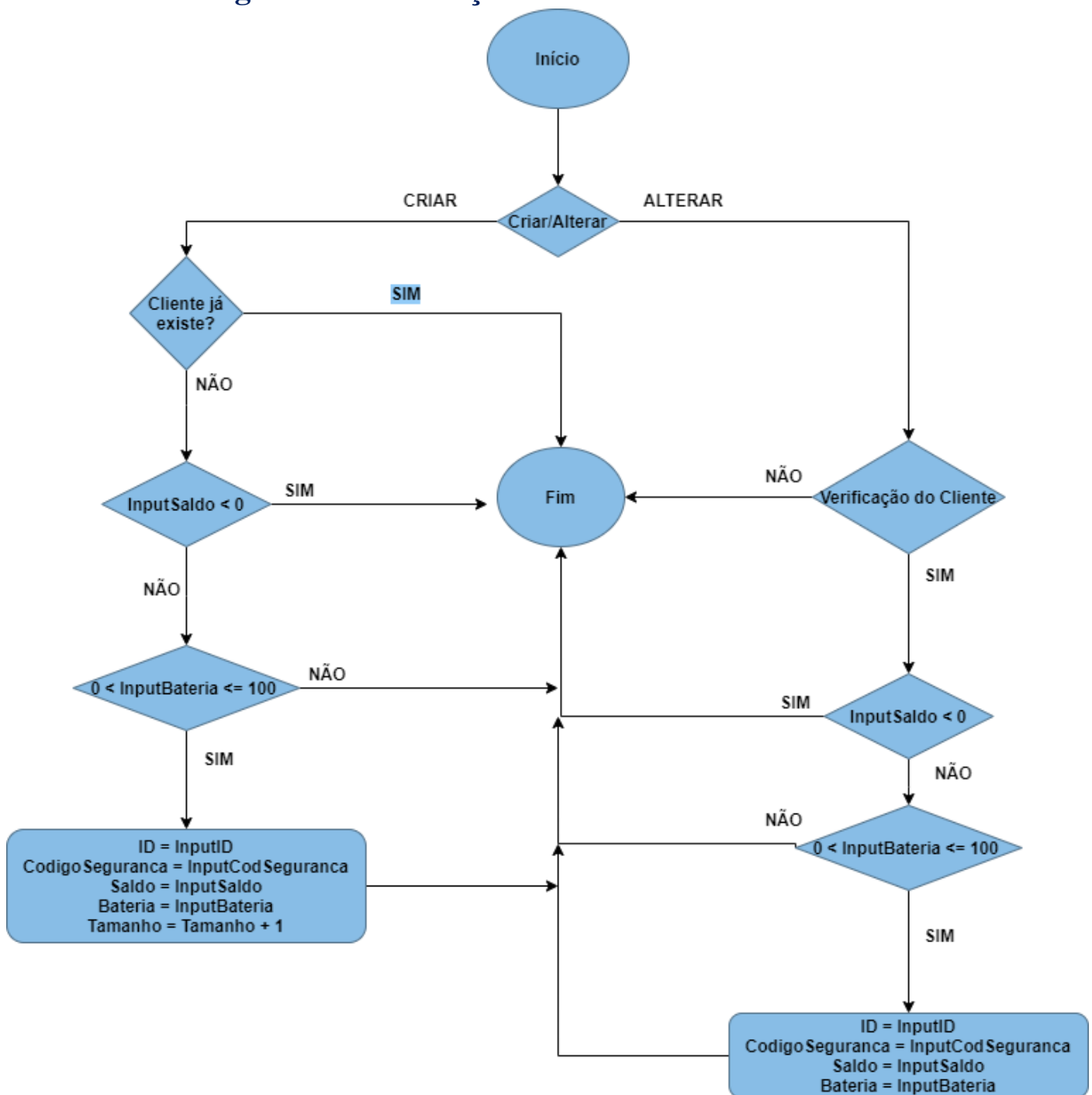
J. Delgado e C. Ribeiro, Arquitetura de Computadores, FCA - Editora de Informática, 2010.

7. Anexo A

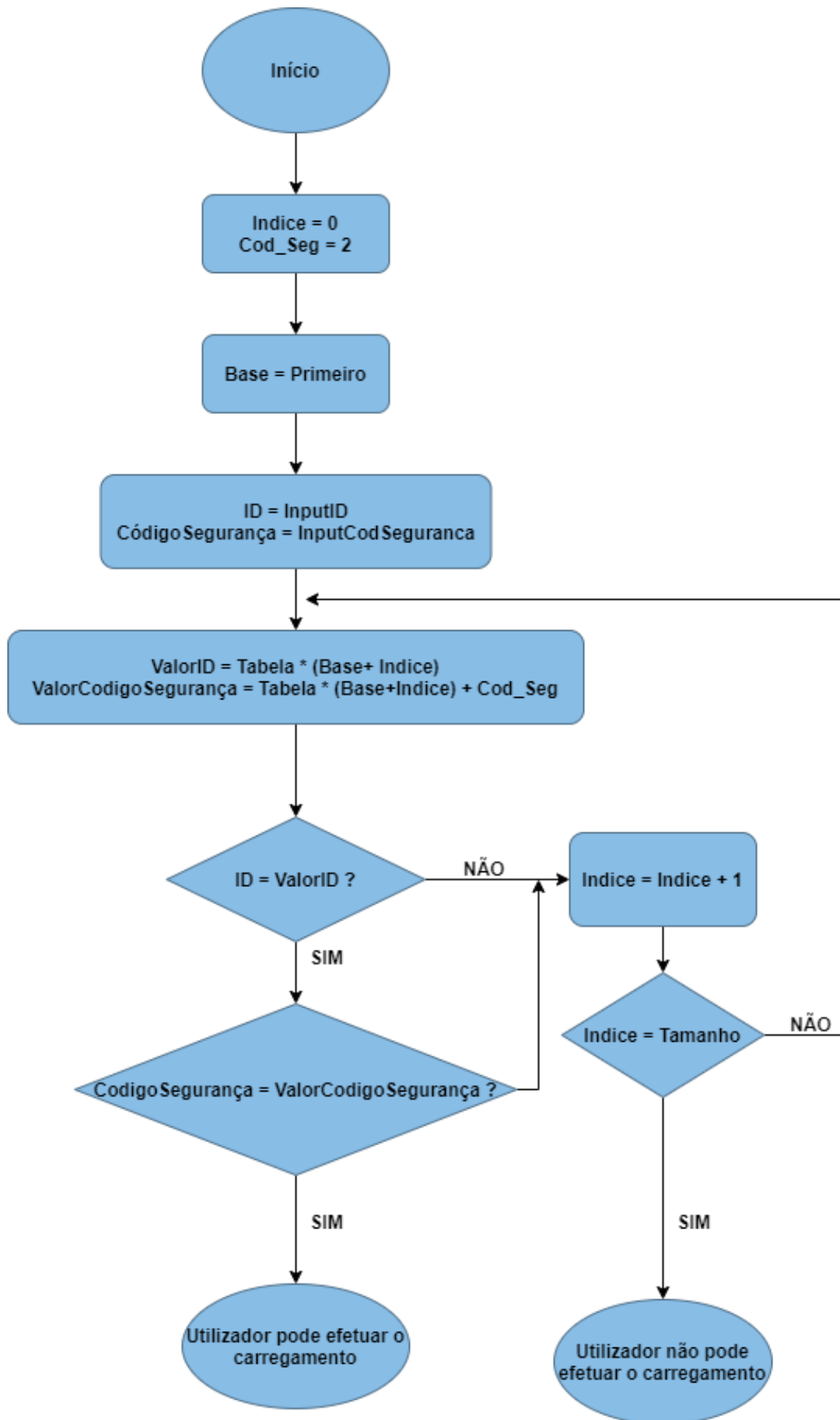
7.1. Fluxograma do Programa Geral



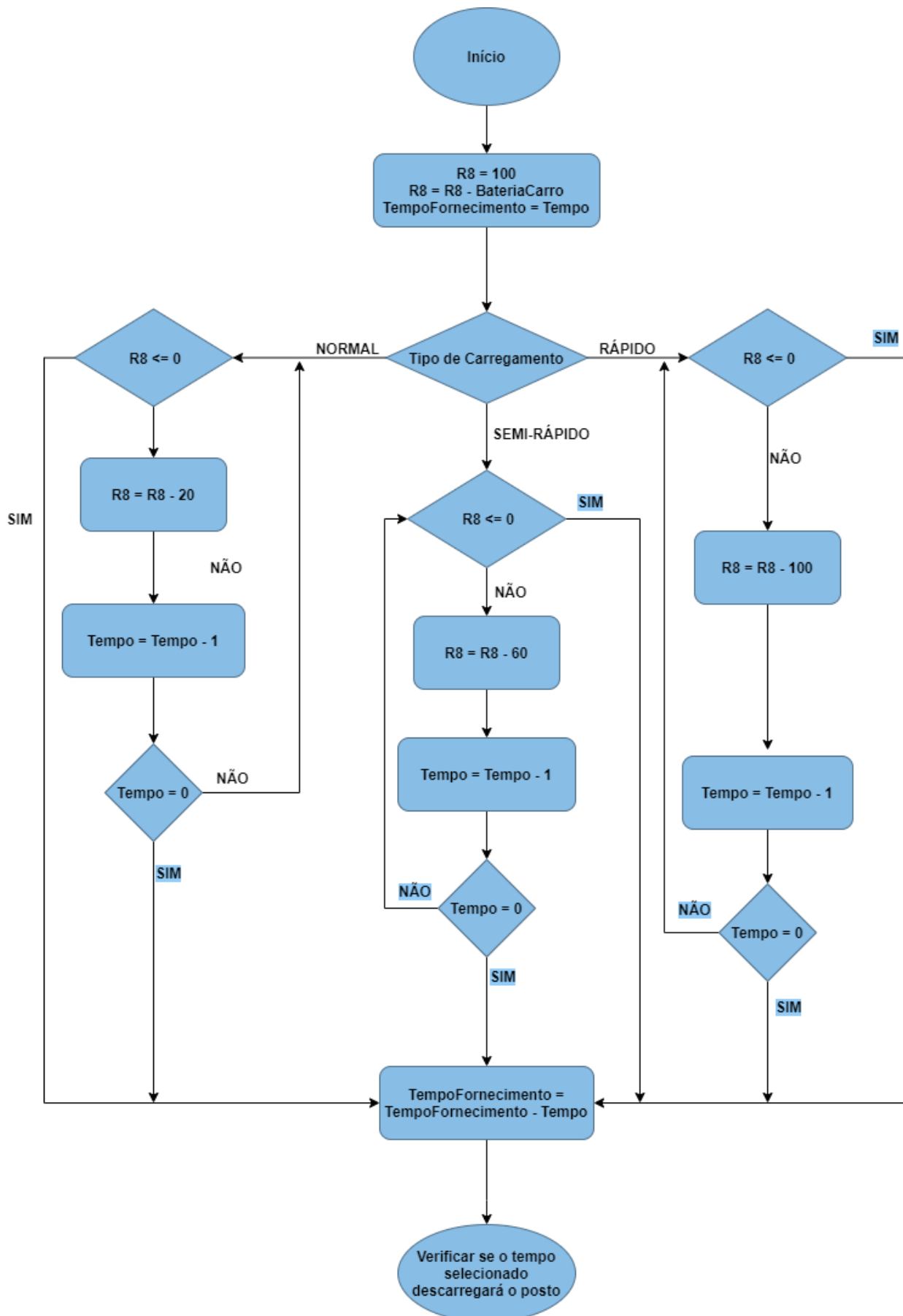
7.2. Fluxograma da Alteração da Base de Dados



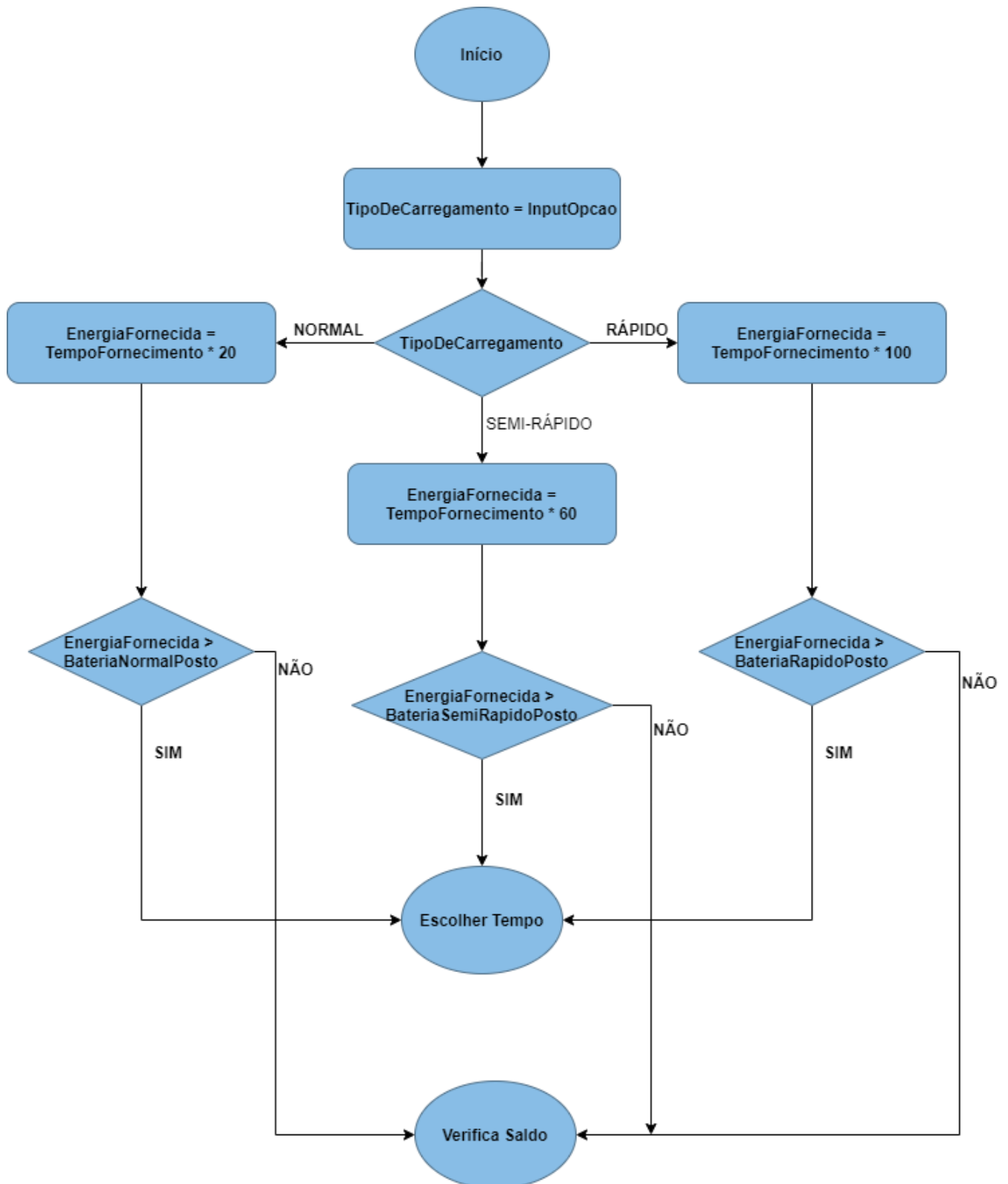
7.3. Fluxograma da Verificação do Utilizador



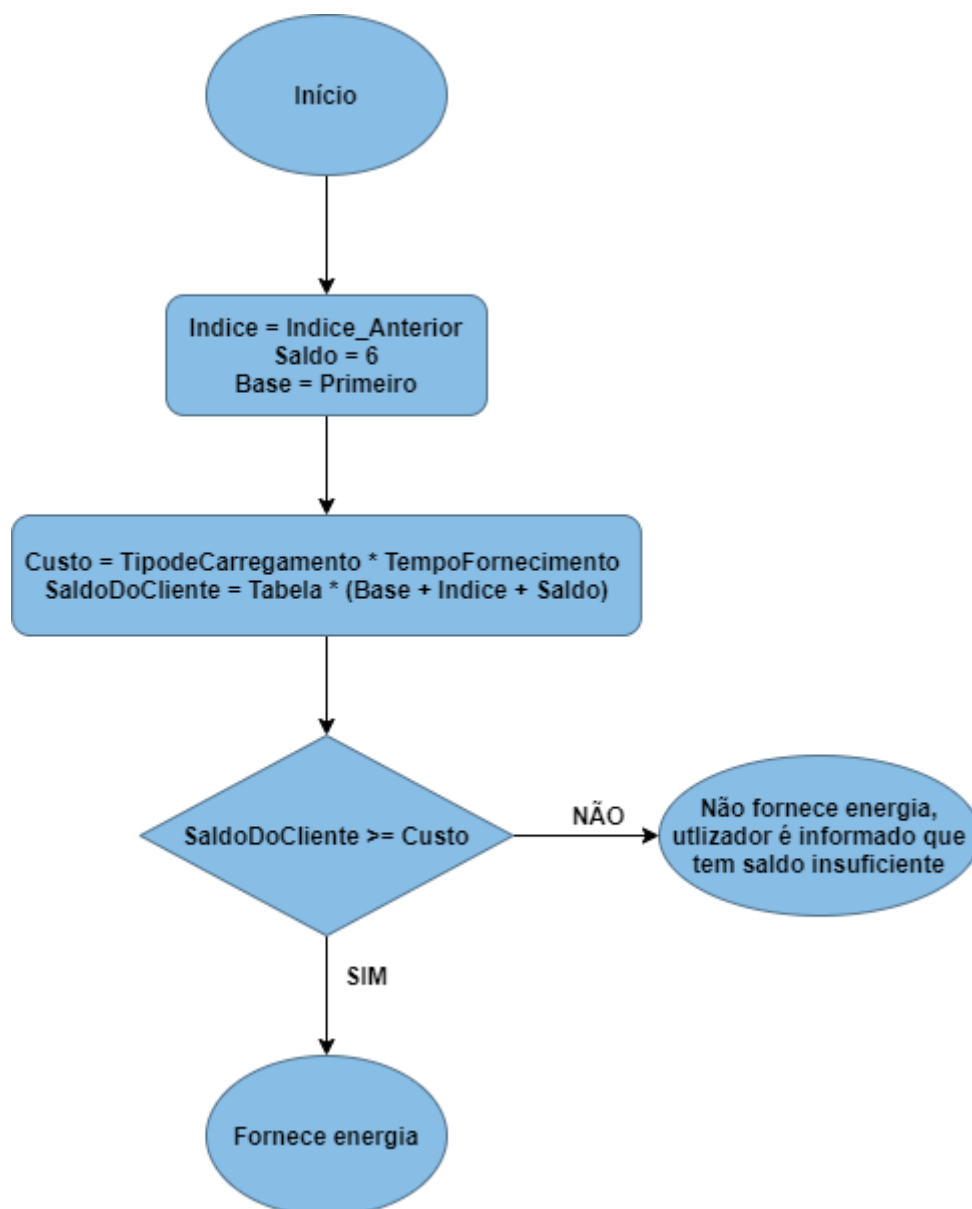
7.4. Fluxograma do Calcula Tempo



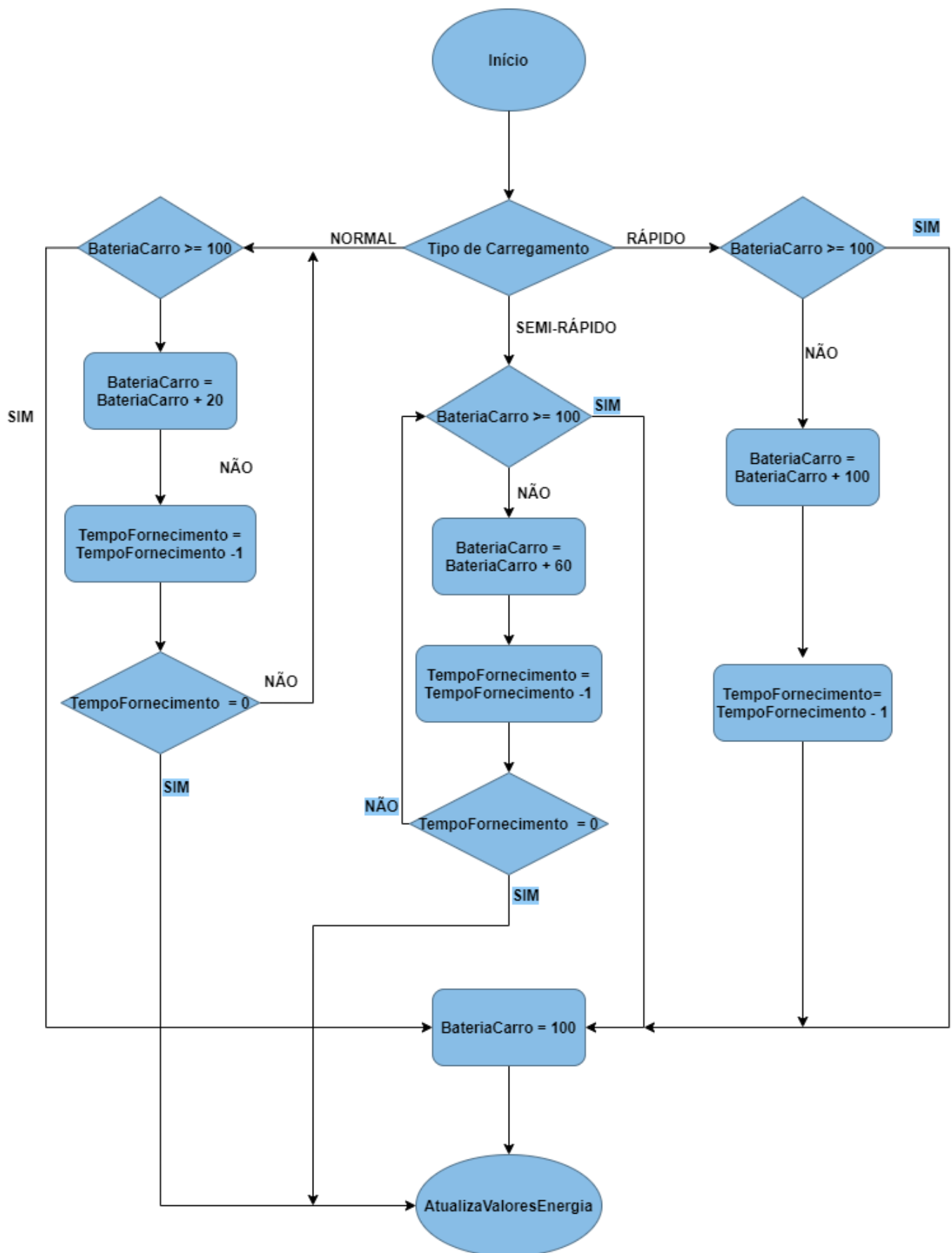
7.5. Fluxograma do Descarregamento do Posto



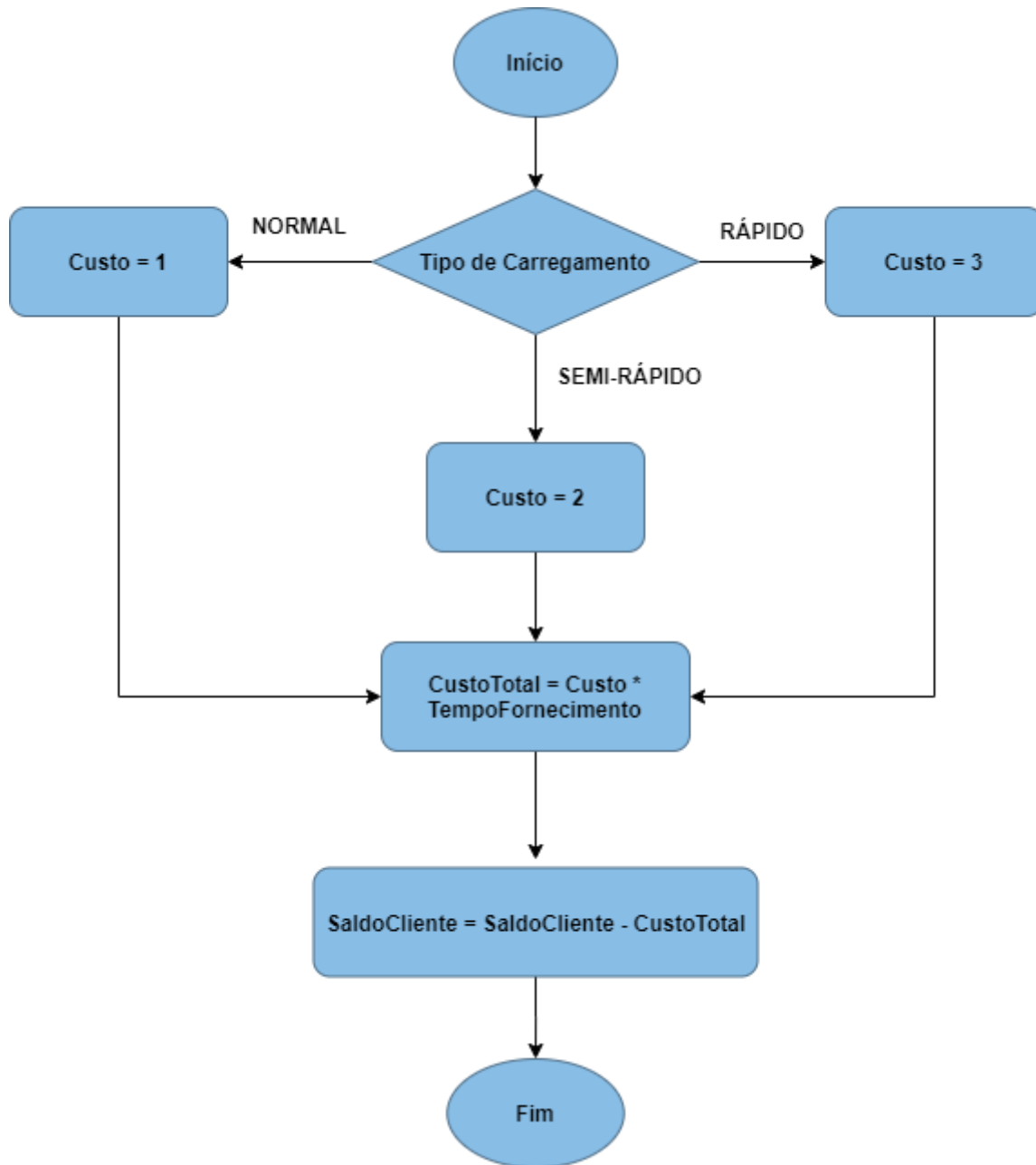
7.6. Fluxograma da Verificação do Saldo



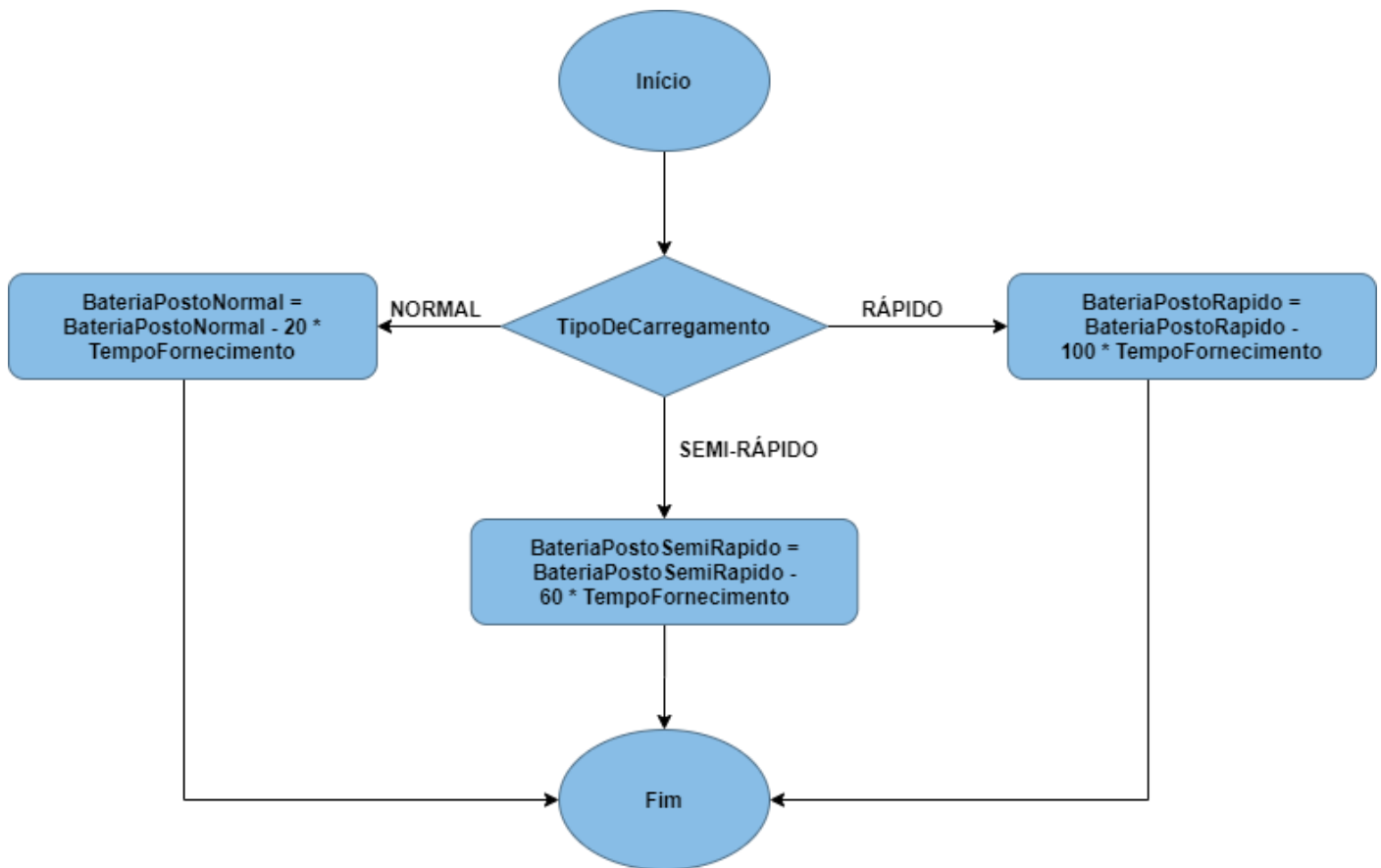
7.7. Fluxograma do Fornecimento de Energia



7.8. Fluxograma do Débito



7.9. Fluxograma da Atualização do Posto



8. Anexo B

```

1 ;constantes:
2 Normal EQU 20 ;percentagem da bateria carregada por uma hora em carregamento normal
3 Semirapido EQU 60 ;percentagem da bateria carregada por uma hora em carregamento semirapido
4 Rapido EQU 100 ;percentagem da bateria carregada por uma hora em carregamento rapido
5 CustoNormal EQU 1 ;custo do carregamento normal
6 CustoSemiRapido EQU 2 ;custo do carregamento semirapido
7 CustoRapido EQU 3 ;custo do carregamento rapido
8 EnderecoBateriaNormal EQU 10E0H ;endereco onde é guardado o valor da bateria para o carregamento normal
9 EnderecoBateriaSemiRapido EQU 10E2H ;endereco onde é guardado o valor da bateria para o carregamento semi-rapido
10 EnderecoBateriaRapido EQU 10E4H ;endereco onde é guardado o valor da bateria para o carregamento rapido
11
12 InicioDisplay EQU 0020H ;Inicio do display principal
13 FimDisplay EQU 008FH ;Fim do display principal
14
15 ;endereços de memória relativos aos inputs:
16 OK EQU 00A0H ;endereco do botão OK
17 Ligar EQU 00A1H ;endereco do botão para ligar o posto (ou desligar)
18 InputID EQU 00B0H ;endereco onde inserir o ID do cliente
19 InputCodSeguranca EQU 00B2H ;endereco onde inserir o código de segurança do cliente
20 InputSaldo EQU 00B4H ;endereco onde inserir o ID do cliente
21 InputBateria EQU 00B6H ;endereco onde inserir o código de segurança do cliente
22 InputOpcao EQU 00C0H ;endereco onde inserir o tipo de carregamento
23 InputTempo EQU 00C2H ;endereco onde inserir o tempo desejado
24 InputIncrementoBateria EQU 00C4H ;endereco onde inserir a bateria a adicionar à bateria selecionada
25
26 ;endereços relativos à base de dados
27 Base_Tabela_Dados EQU 1100H ;endereco do inicio da base de dados
28 CodSeguranca EQU 02H ;aumento relativo ao inicio dos dados do cliente para ler o código de segurança
29 Saldo EQU 04H ;aumento relativo ao inicio dos dados do cliente para ler o saldo
30 BateriaCarro EQU 06H ;aumento relativo ao inicio dos dados do cliente para ler quanta bateria o carro do cliente tem
31 Proximo EQU 08H ;salto a executar para ler os dados do próximo cliente
32 EnderecoTamanho EQU 10FEH ;endereco que contém o número de clientes na base de dados
33
34 StackPointer EQU 8000H ;endereco da pilha
35
36
37
38 ;*****
39 ; DISPLAYS
40 ;*****
41 PLACE 2000H
42 Display_InputVerifyCliente: ;Display do menu para inserir ID e código de segurança
43 String " VERIFICACAO "
44 String " "
45 String " INTRODUZA ID "
46 String " "
47 String " E CODIGO SEG. "
48 String " "
49 String " OK - continuar "
50
51 PLACE 2080H
52 Display_VerificacaoSucesso: ;Display para informar o utilizador que foi verificado
53 String " "
54 String " "
55 String " AUTORIZADO "
56 String " COM SUCESSO "
57 String " "
58 String " "
59 String " OK - continuar "
60
61 PLACE 2100H
62 Display_VerificacaoFalhada: ;Display para informar o utilizador que não foi verificado
63 String " ATENCAO "
64 String " "
65 String " "
66 String " "
67 String " NAO AUTORIZADO "
68 String " "
69 String " "
70 String " OK - continuar "
71
72 PLACE 2180H
73 Display_InserEnergia: ;Display do menu para carregar o posto
74 String " CARREGAR POSTO "
75 String "Escolha bateria:"
76 String "1)- Normal "
77 String "2)- Semi-Rapido "
78 String "3)- Rapido "
79 String "4)- Avançar "
80 String " OK - continuar "
81
82 PLACE 2200H
83 Display_InserEnergiaQuanta: ;Display a perguntar ao utilizador quanta bateria carregar
84 String " CARREGAR POSTO "
85 String " Quanto quer "
86 String " carregar? "
87 String " "
88 String " "
89 String " "
90 String " OK - continuar "
91
92 PLACE 2280H
93 Display_NiveisDeEnergia: ;Display para informar o utilizador do estado das baterias do posto
94 String " ESTADO DO POSTO"
95 String "Normal: "
96 String "SemiRapido: "
97 String " "
98 String "Rapido: "
99 String " "
100 String " OK - continuar "
101

```

```

102 PLACE 2300H
103 Display_EscolheCarregamento:
104     String " ESCOLHA O TIPO "
105     String "DE CARREGAMENTO:"
106     String " 1- Normal "
107     String " 2- Semi-Rapido"
108     String " 3- Rapido "
109     String " "
110     String " OK - continuar "
111
112 PLACE 2380H
113 Display_OpcaoInvalida:
114     String " ATENCAO "
115     String " "
116     String " OPCAO "
117     String " INVALIDA "
118     String " "
119     String " "
120     String " OK - continuar "
121
122 PLACE 2400H
123 Display_EscolherTempo:
124     String " CARREGAMENTO "
125     String " "
126     String "Por quanto tempo"
127     String "quer carregar o "
128     String " seu carro? "
129     String " "
130     String " OK - continuar "
131
132 PLACE 2480H
133 Display_TempoInvalido:
134     String " OPCAO "
135     String " INVALIDA "
136     String " "
137     String "Tempo tem de ser"
138     String " superior a 0 "
139     String " "
140     String " OK - continuar "
141
142 PLACE 2500H
143 Display_Debito:
144     String " CARREGAMENTO "
145     String " "
146     String " Saldo: "
147     String " euros"
148     String " Custo: "
149     String " euros"
150     String " OK - continuar "
151
152 PLACE 2580H
153 Display_SaldoInsuficiente:
154     String " ATENCAO "
155     String " "
156     String " SALDO "
157     String " INSUFICIENTE "
158     String " "
159     String " "
160     String " OK - continuar "
161
162 PLACE 2600H
163 Display_InfoCarregamento:
164     String " CARREGAMENTO "
165     String " "
166     String " Tipo: "
167     String " "
168     String " Duracao: "
169     String " horas"
170     String " OK - continuar "
171
172 PLACE 2680H
173 Display_CarregamentoConcluido:
174     String " "
175     String " "
176     String " CARREGAMENTO "
177     String " CONCLUIDO "
178     String " "
179     String " "
180     String " OK - continuar "
181
182 PLACE 2700H
183 Display_Overflow:
184     String " ATENCAO "
185     String " "
186     String " OCORREU "
187     String " OVERFLOW "
188     String " "
189     String " OK - continuar "
190
191 PLACE 2780H
192 Display_UltrapassaCargaMaxima:
193     String " ATENCAO "
194     String " O tempo "
195     String " selecionado e "
196     String " superior ao "
197     String "necessario para "
198     String "carregar bateria"
199     String " OK - continuar "
200
;Display do menu para escolher o tipo de carregamento

;Display para informar o utilizador que escolheu uma opção inválida

;Display para perguntar ao utilizador por quanto tempo o carro deve carregar

;Display para informar o utilizador que o tempo que ele inseriu é inválido (tem de ser superior a 0)

;Display para informar o utilizador do saldo que terá após o carregamento e quanto dinheiro o carregamento custou-lhe

;Display para informar o utilizador que não tem saldo suficiente para o carregamento

;Display para informar o utilizador o tipo de carregamento que escolheu e quanto tempo demorará

;Display para informar o utilizador que o carregamento terminou

;Display para informar o utilizador que ao incrementar a bateria pretendida, ocorreu overflow

;Display para informar o utilizador que o tempo que selecionou é superior ao necessário para carregar por completo o seu carro

```

```

201 PLACE 2800H
202 Display_TempoUltrapassa: ;Display para informar o utilizador que a bateria do carregamento que ele pretendia não é suficiente para o carregamento
203     String " ATENCAO "
204     String " NAO HA ENERGIA "
205     String " SUFICIENTE PARA"
206     String " O CARREGAMENTO "
207     String " PRETENDIDO "
208     String " "
209     String " OK - continuar "
210
211 PLACE 2890H
212 Display_AlterarBaseDeDados: ;Display do menu para alterar a base de dados
213     String " BASE DE DADOS "
214     String "Pretende fazer "
215     String "alteracoes na "
216     String "base de dados? "
217     String "1- Sim "
218     String "2- Nao "
219     String " OK - continuar "
220
221 PLACE 2900H
222 Display_CriarOuAlterar: ;Display do menu para selecionar entre criar ou alterar um cliente
223     String " BASE DE DADOS "
224     String "Pretende criar "
225     String "um cliente ou "
226     String "alterar um? "
227     String "1- Criar "
228     String "2- Alterar Dados"
229     String " OK - continuar "
230
231 PLACE 2980H
232 Display_QualCliente: ;Display para perguntar ao utilizador a qual cliente alterar os dados
233     String " BASE DE DADOS "
234     String "Insira o ID e "
235     String "codigo de seg. "
236     String "do cliente que "
237     String "pretende alterar"
238     String " "
239     String " OK - continuar "
240
241 PLACE 2A00H
242 Display_InserirDadosParaAlterar: ;Display para perguntar ao utilizador os novos dados do cliente que vai ser alterado
243     String " BASE DE DADOS "
244     String "Insira o ID, "
245     String "codigo de seg., "
246     String "saldo e bateria "
247     String "do carro novos "
248     String "para alterar "
249     String " OK - continuar "
250
251 PLACE 2A80H
252 Display_DadosAlterados: ;Display para informar o utilizador que os dados foram alterados com sucesso
253     String " BASE DE DADOS "
254     String " "
255     String " DADOS ALTERADOS"
256     String " COM SUCESSO! "
257     String " "
258     String " "
259     String " OK - continuar "
260
261 PLACE 2B00H
262 Display_ClienteNaoEstaNaDatabase: ;Display para informar o utilizador que o cliente que pretendia não está na base de dados
263     String " BASE DE DADOS "
264     String " "
265     String " Cliente nao "
266     String " esta na base "
267     String " de dados "
268     String " "
269     String " OK - continuar "
270
271 PLACE 2B80H
272 Display_InserirDadosNovos: ;Display para perguntar ao utilizador quais dados o novo cliente a ser criado terá
273     String " BASE DE DADOS "
274     String "Insira o ID, "
275     String "codigo de seg., "
276     String "saldo e bateria "
277     String "do carro do novo"
278     String " cliente "
279     String " OK - continuar "
280
281 PLACE 2C00H
282 Display_SaldoInvalido: ;Display para informar o utilizador que o saldo que inseriu é inválido
283     String " BASE DE DADOS "
284     String " "
285     String " SALDO INVALIDO "
286     String " (tem de ser "
287     String " positivo) "
288     String " "
289     String " OK - continuar "
290
291 PLACE 2C80H
292 Display_BateriaInvalida: ;Display para informar o utilizador que a bateria do carro que inseriu é inválida
293     String " BASE DE DADOS "
294     String "BATERIA DO CARRO"
295     String " INVALIDA "
296     String " (tem de ser "
297     String "positiva e menor"
298     String " que 100) "
299     String " OK - continuar "
300

```

```

301 PLACE 2D00H
302 Display_NovoClienteCriado: ;Display para informar o utilizador que o novo cliente foi criado
303 String " BASE DE DADOS "
304 String " "
305 String " NOVO CLIENTE "
306 String " CRIADO! "
307 String " "
308 String " "
309 String " OK - continuar "
310
311 PLACE 2D80H
312 Display_IncrementaBateriaInvalido: ;Display para informar o utilizador que o valor que inseriu para carregar a bateria é inválido
313 String " ATENCAO "
314 String " "
315 String " VALOR DEVE "
316 String " SER POSITIVO! "
317 String " "
318 String " "
319 String " OK - continuar "
320
321 PLACE 2E00H
322 Display_BateriaCarregada: ;Display para informar o utilizador que a bateria foi carregada
323 String " CARREGAR POSTO "
324 String " "
325 String " BATERIA "
326 String " CARREGADA! "
327 String " "
328 String " "
329 String " OK - continuar "
330
331 PLACE 2E80H
332 Display_ClienteExiste: ;Display para informar o utilizador que o ID que inseriu é inválido (pois já existe um cliente com dito ID)
333 String " BASE DE DADOS "
334 String " "
335 String " ID INVALIDO "
336 String "(Ja existe um "
337 String " cliente com "
338 String " esse ID) "
339 String " OK - continuar "
340
341 PLACE 2F00H
342 Display_CarregarPosto: ;Display para informar o utilizador é necessário carregar o posto
343 String " ATENCAO "
344 String "O POSTO NAO ESTA"
345 String "OPERACIONAL. E "
346 String "PRECISO CARREGAR"
347 String " AS BATERIAS "
348 String " "
349 String " OK - continuar "
350
351 PLACE 2F80H
352 Display_BateriaJACarregada: ;Display para informar o utilizador que o seu carro já se encontra carregado
353 String " ATENCAO "
354 String " "
355 String " O Seu carro ja "
356 String " se encontra "
357 String " carregado! "
358 String " "
359 String " OK - continuar "
360
361
362 ;*****
363 ;
364 ; MAIN
365 ;*****
366 PLACE 0000H
367 Inicio:
368 MOV SP, StackPointer ;mete em SP o endereço do início da pilha
369 MOV R0, Main ;coloca em R0 o endereço da tag Main
370 JMP R0 ;salta para a tag Main
371
372
373 ;*****
374 ;
375 ; PROGRAMA
376 ;*****
377 PLACE 6000H
378 Main:
379 CALLF LimpaDisplay ;limpa o display
380 CALL Desligado ;chama a rotina para verificar se é para desligar a máquina (e desligá-la, caso seja)
381 CMP R10, -1 ;compara-se o valor em R10 com -1
382 JEQ Main ;R10 será igual a -1 se o posto estiver desligado. Se tal acontecer, salta-se para o main (não se avança)
383 MOV R6, EnderecoBateriaNormal ;mete em R6 o endereço onde está guardado o valor da bateria do posto normal
384 MOV R7, EnderecoBateriaSemiRapido ;mete em R7 o endereço onde está guardado o valor da bateria do posto semirapido
385 MOV R8, EnderecoBateriaRapido ;mete em R8 o endereço onde está guardado o valor da bateria do posto rapido
386 MOV R0, [R6] ;guarda em R0 o valor da bateria do posto normal presente na memória
387 MOV R1, [R7] ;guarda em R1 o valor da bateria do posto semirapido presente na memória
388 MOV R2, [R8] ;guarda em R2 o valor da bateria do posto rapido presente na memória
389 CALL Programa ;chama-se o programa
390 CALLF LimpaPeriféricosEntrada ;limpa os periféricos de entrada
391 MOV R6, EnderecoBateriaNormal ;coloca no registo 6 o endereço onde está guardado o valor da bateria do posto normal
392 MOV R7, EnderecoBateriaSemiRapido ;coloca no registo 7 o endereço onde está guardado o valor da bateria do posto semi-rapido
393 MOV R8, EnderecoBateriaRapido ;coloca no registo 8 o endereço onde está guardado o valor da bateria do posto rapido
394 MOV [R6], R0 ;atualiza o valor da bateria do posto normal
395 MOV [R7], R1 ;atualiza o valor da bateria do posto semi-rapido
396 MOV [R8], R2 ;atualiza o valor da bateria do posto rapido
397 JMP Main

```

```

398 Programa:
399 CALLF Limpadisplay ;limpa o display
400 CALL AlteraBaseDeDados ;rotina para alterar a base de dados
401 CALL Desligado ;chama a rotina para verificar se é para desligar a máquina (e desligá-la, caso seja)
402 CMP R10,-1 ;compara-se o valor em R10 com -1
403 JEQ FimPrograma ;R10 será igual a -1 se o posto estiver desligado. Se tal acontecer, salta-se para o fim do programa
404 CALL InserirEnergia ;rotina para carregar os postos
405 CALL Desligado ;chama a rotina para verificar se é para desligar a máquina (e desligá-la, caso seja)
406 CMP R10,-1 ;compara-se o valor em R10 com -1
407 JEQ FimPrograma ;R10 será igual a -1 se o posto estiver desligado. Se tal acontecer, salta-se para o fim do programa
408 CALL NiveleDeEnergia ;rotina para indicar se os postos estão operacionais
409 CMP R10,-1 ;compara-se o valor em R10 com -1
410 JEQ FimPrograma ;R10 será igual a -1 se o posto estiver desligado. Se tal acontecer, salta-se para o fim do programa
411 CALL Verificacao_Cliente ;rotina para efetuar a verificação do cliente
412 CMP R10,-1 ;compara-se o valor em R10 com -1
413 JEQ FimPrograma ;R10 será igual a -1 se o utilizador não foi verificado. Se tal acontecer, salta-se para o fim do programa
414 CALL Carregamento ;rotina para efetuar o carregamento do carro
415
416 FimPrograma:
417 RET
418
419 Desligado:
420
421 PUSH R5 ; Guarda na pilha os registos alterados durante esta rotina
422 PUSH R6 ;
423 MOV R10,0 ;reseta o valor de R10 para 0
424 MOV R6,Ligar ;mete no R6 o endereço do botão que liga a máquina
425 MOV R5,[R6] ;mete em R5 o byte endereçado por R6 (o botão de ligar / desligar)
426 CMP R5,0 ;compara R5 com 0
427 JNE FimDesligado ;se R5 é diferente de 0, o posto está ligado e efetua-se este jump; caso contrário, este jump não é efetuado
428 MOV R10,-1 ;mete-se em R10 o valor -1, para indicar que o posto está desligado
429
430 FimDesligado:
431 POP R6 ;
432 POP R5 ;
433 RET ;
434
435 Fim:
436 JMP Fim
437
438 ;
439 ;
440 ;
441 ;
442 ;
443 ;
444 ;
445 ;
446 ;
447 ;
448 ;
449 ;
450 ;
451 ;
452 ;
453 ;
454 ;
455 ;
456 ;
457 ;
458 ;
459 ;
460 ;
461 ;
462 ;
463 ;
464 ;
465 ;
466 ;
467 ;
468 ;
469 ;
470 ;
471 ;
472 ;
473 ;
474 ;
475 ;
476 ;
477 ;
478 ;
479 ;
480 ;
481 ;
482 ;
483 ;
484 ;
485 ;
486 ;
487 ;
488 ;
489 ;
490 ;
491 ;
492 ;
493 ;
494 ;
495 ;
496 ;
497 ;
498 ;
499 ;
500 ;
501 ;
502 ;
503 ;
504 ;
505 ;
506 ;
507 ;
508 ;
509 ;
510 ;
511 ;
512 ;
513 ;
514 ;
515 ;
516 ;
517 ;
518 ;
519 ;
520 ;
521 ;
522 ;
523 ;
524 ;
525 ;
526 ;
527 ;
528 ;
529 ;
530 ;
531 ;
532 ;
533 ;
534 ;
535 ;
536 ;
537 ;
538 ;
539 ;
540 ;
541 ;
542 ;
543 ;
544 ;
545 ;
546 ;
547 ;
548 ;
549 ;
550 ;
551 ;
552 ;
553 ;
554 ;
555 ;
556 ;
557 ;
558 ;
559 ;
560 ;
561 ;
562 ;
563 ;
564 ;
565 ;
566 ;
567 ;
568 ;
569 ;
570 ;
571 ;
572 ;
573 ;
574 ;
575 ;
576 ;
577 ;
578 ;
579 ;
580 ;
581 ;
582 ;
583 ;
584 ;
585 ;
586 ;
587 ;
588 ;
589 ;
590 ;
591 ;
592 ;
593 ;
594 ;
595 ;
596 ;
597 ;
598 ;
599 ;
600 ;
601 ;
602 ;
603 ;
604 ;
605 ;
606 ;
607 ;
608 ;
609 ;
610 ;
611 ;
612 ;
613 ;
614 ;
615 ;
616 ;
617 ;
618 ;
619 ;
620 ;
621 ;
622 ;
623 ;
624 ;
625 ;
626 ;
627 ;
628 ;
629 ;
630 ;
631 ;
632 ;
633 ;
634 ;
635 ;
636 ;
637 ;
638 ;
639 ;
640 ;
641 ;
642 ;
643 ;
644 ;
645 ;
646 ;
647 ;
648 ;
649 ;
650 ;
651 ;
652 ;
653 ;
654 ;
655 ;
656 ;
657 ;
658 ;
659 ;
660 ;
661 ;
662 ;
663 ;
664 ;
665 ;
666 ;
667 ;
668 ;
669 ;
670 ;
671 ;
672 ;
673 ;
674 ;
675 ;
676 ;
677 ;
678 ;
679 ;
680 ;
681 ;
682 ;
683 ;
684 ;
685 ;
686 ;
687 ;
688 ;
689 ;
690 ;
691 ;
692 ;
693 ;
694 ;
695 ;
696 ;
697 ;
698 ;
699 ;
700 ;
701 ;
702 ;
703 ;
704 ;
705 ;
706 ;
707 ;
708 ;
709 ;
710 ;
711 ;
712 ;
713 ;
714 ;
715 ;
716 ;
717 ;
718 ;
719 ;
720 ;
721 ;
722 ;
723 ;
724 ;
725 ;
726 ;
727 ;
728 ;
729 ;
730 ;
731 ;
732 ;
733 ;
734 ;
735 ;
736 ;
737 ;
738 ;
739 ;
740 ;
741 ;
742 ;
743 ;
744 ;
745 ;
746 ;
747 ;
748 ;
749 ;
750 ;
751 ;
752 ;
753 ;
754 ;
755 ;
756 ;
757 ;
758 ;
759 ;
760 ;
761 ;
762 ;
763 ;
764 ;
765 ;
766 ;
767 ;
768 ;
769 ;
770 ;
771 ;
772 ;
773 ;
774 ;
775 ;
776 ;
777 ;
778 ;
779 ;
780 ;
781 ;
782 ;
783 ;
784 ;
785 ;
786 ;
787 ;
788 ;
789 ;
790 ;
791 ;
792 ;
793 ;
794 ;
795 ;
796 ;
797 ;
798 ;
799 ;
800 ;
801 ;
802 ;
803 ;
804 ;
805 ;
806 ;
807 ;
808 ;
809 ;
810 ;
811 ;
812 ;
813 ;
814 ;
815 ;
816 ;
817 ;
818 ;
819 ;
820 ;
821 ;
822 ;
823 ;
824 ;
825 ;
826 ;
827 ;
828 ;
829 ;
830 ;
831 ;
832 ;
833 ;
834 ;
835 ;
836 ;
837 ;
838 ;
839 ;
840 ;
841 ;
842 ;
843 ;
844 ;
845 ;
846 ;
847 ;
848 ;
849 ;
850 ;
851 ;
852 ;
853 ;
854 ;
855 ;
856 ;
857 ;
858 ;
859 ;
860 ;
861 ;
862 ;
863 ;
864 ;
865 ;
866 ;
867 ;
868 ;
869 ;
870 ;
871 ;
872 ;
873 ;
874 ;
875 ;
876 ;
877 ;
878 ;
879 ;
880 ;
881 ;
882 ;
883 ;
884 ;
885 ;
886 ;
887 ;
888 ;
889 ;
890 ;
891 ;
892 ;
893 ;
894 ;
895 ;
896 ;
897 ;
898 ;
899 ;
900 ;
901 ;
902 ;
903 ;
904 ;
905 ;
906 ;
907 ;
908 ;
909 ;
910 ;
911 ;
912 ;
913 ;
914 ;
915 ;
916 ;
917 ;
918 ;
919 ;
920 ;
921 ;
922 ;
923 ;
924 ;
925 ;
926 ;
927 ;
928 ;
929 ;
930 ;
931 ;
932 ;
933 ;
934 ;
935 ;
936 ;
937 ;
938 ;
939 ;
940 ;
941 ;
942 ;
943 ;
944 ;
945 ;
946 ;
947 ;
948 ;
949 ;
950 ;
951 ;
952 ;
953 ;
954 ;
955 ;
956 ;
957 ;
958 ;
959 ;
960 ;
961 ;
962 ;
963 ;
964 ;
965 ;
966 ;
967 ;
968 ;
969 ;
970 ;
971 ;
972 ;
973 ;
974 ;
975 ;
976 ;
977 ;
978 ;
979 ;
980 ;
981 ;
982 ;
983 ;
984 ;
985 ;
986 ;
987 ;
988 ;
989 ;
990 ;
991 ;
992 ;
993 ;
994 ;
995 ;
996 ;
997 ;
998 ;
999 ;

```

```

536 AlterarDados: ;A rotina chega aqui se o utilizador ou pretende alterar os dados de um cliente existente ou inseriu uma opção inválida
537 CMP R3,2 ;Compara a opção escolhida pelo utilizador com 2
538 JNE OpcaoInvalidaDatabase ;se é diferente de 2, o utilizador inseriu uma opção inválida efetua-se este salto: se é igual a 2, procede-se para as instruções abaixo
539 MOV R5,InputID ;mete-se em R5 o endereço de onde se lerá o ID do cliente ao qual a alterar os dados
540 MOV R6, InputCodSeguranca ;mete-se em R6 o endereço de onde se lerá o código de segurança do cliente ao qual alterar os dados
541 MOV R8, Base_Tabela_Dados ;mete-se em R8 o início da base de dados
542 MOV R3,0 ;mete-se em R3 o índice
543 MOV R2,Proximo ;mete-se em R2 o valor de Proximo para avançarmos para o próximo cliente a verificar
544 MOV R9, Display_QualCliente ;Mete no registo 9 o endereço do display a mostrar ao utilizador
545 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
546 MOV R0,[R5] ;mete em R0 o ID que o utilizador inseriu
547 MOV R1,[R6] ;mete em R1 o código de segurança que o utilizador inseriu
548 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
549 MOV R9,EnderecoTamanho ;mete em R9 o endereço onde encontra-se o tamanho da base de dados
550 MOV R7,[R9] ;R7 é o número de clientes na base de dados
551
552 Ciclo_AlterarDados: ;ciclo em que se percorre a base de dados
553 CMP R3,R7 ;compara-se o índice com o tamanho da base de dados
554 JEQ ClientesNaoPresente ;se é igual, chegou-se ao fim da base de dados e efetua-se este salto
555 MOV R5,[R5] ;R5 tem o valor do ID da tabela de base de dados a verificar
556 MOV R6,[R6+CodSeguranca] ;R6 tem o valor de código de segurança da tabela de base de dados a verificar
557 CMP R0,R5 ;compara-se o ID inserido pelo utilizador com o ID do cliente a ser verificado atualmente
558 JNE ContinuarTabela ;se os valores forem diferentes, efetua-se este salto para avançar para o próximo cliente, se não estivermos já no último cliente da base de dados
559 CMP R1,R6 ;compara-se o código de segurança inserido pelo utilizador com o código de segurança do cliente a ser verificado atualmente
560 JNE ContinuarTabela ;se os valores forem diferentes, efetua-se este salto para avançar para o próximo cliente, se não estivermos já no último cliente da base de dados
561 MOV R10, Proximo ;se tanto o ID como o código de segurança são iguais, guarda-se em R10 o incremento a efetuar para ler os dados de um novo cliente
562 MUL R10,R3 ;multiplicamos o valor de Proximo pelo índice para obtermos em R10 o incremento a adicionar ao início da base de dados para alterarmos os dados do cliente verificado
563 JMP AlterarDadosCliente ;efetua-se o salto para a parte que tratará de alterar os dados do cliente que o utilizador selecionou
564
565 ContinuarTabela: ;esta parte avança para verificar os dados do próximo cliente, porque previamente os dados que o utilizador inseriu e os que estão na base de dados não coincidiam
566 ADD R3,1 ;adiciona-se ao índice 1 para indicar que avançamos um cliente
567 ADD R5,2 ;avança a base de dados para o próximo cliente a verificar
568 JMP Ciclo_AlterarDados ;faz o salto para o ciclo para verificarmos este próximo cliente
569
570 ClientesNaoPresente: ;esta parte da rotina indica ao utilizador que os dados (ID e código de segurança) que inseriu não coincidem com os dados de nenhum cliente na base de dados
571 MOV R9, Display_ClientesNaoEstaNaDatabase ;Mete no registo 9 o endereço do display a mostrar ao utilizador
572 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
573 JMP AlterarOuNao ;salta-se para o início da função
574
575 AlterarDadosCliente: ;chega-se aqui se encontrou-se na base de dados o cliente ao qual o utilizador pretende alterar os dados
576 MOV R5, InputID ;mete em R5 o endereço de onde se lê o novo ID do cliente ao qual alteraremos os dados
577 MOV R6, InputCodSeguranca ;mete em R6 o endereço de onde se lê o novo código de segurança do cliente ao qual alteraremos os dados
578 MOV R7, InputSaldo ;mete em R7 o endereço de onde se lê o novo saldo do cliente ao qual alteraremos os dados
579 MOV R4, InputBateria ;mete em R7 o endereço de onde se lê quanta bateria o carro do cliente terá agora que iremos alterar os seus dados
580 MOV R9, Display_InserirDadosParaAlterar ;Mete no registo 9 o endereço do display a mostrar ao utilizador
581 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
582 MOV R0,0 ;mete em R0 o novo valor do ID do cliente, que o utilizador meteu
583 MOV R1,[R6] ;mete em R1 o novo valor do código de segurança do cliente, que o utilizador meteu
584 MOV R2,[R7] ;mete em R2 o novo valor do saldo do cliente, que o utilizador meteu
585 MOV R3,[R4] ;mete em R3 o novo valor da bateria do carro do cliente, que o utilizador meteu
586 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
587 CMP R2,0 ;compara o valor do saldo com zero
588 JLT SaldoInvalido ;se o saldo que o utilizador inseriu para o novo cliente for negativo, efetua-se este salto e o novo cliente não é criado
589 CMP R3,100 ;compara o valor da bateria do carro com zero
590 JLT BateriaInvalida ;se o valor da bateria do carro for negativo, efetua-se este salto e o novo cliente não é criado
591 MOV R7,100 ;mete em R7 o valor 100 para fins de comparação
592 CMP R3,R7 ;compara o valor da bateria do carro com R7, que é 100
593 JLT BateriaInvalida ;se o valor da bateria do carro for maior que 100, efetua-se este salto e o novo cliente não é criado
594 MOV R8,Base_Tabela_Dados ;mete em R8 o início da base de dados
595 ADD R8,R10 ;acrescenta ao início da base de dados o R10 que foi definido anteriormente como sendo o incremento a adicionar ao início da base de dados para chegarmos aos dados do cliente pretendido pelo utilizador
596 MOV [R8],R0 ;atualizar o ID do cliente com o ID que o utilizador inseriu
597 MOV [R8+CodSeguranca],R1 ;atualizar o código de segurança do cliente com o código de segurança que o utilizador inseriu
598 MOV [R8+Saldo],R2 ;atualizar o saldo do cliente com o saldo do cliente que o utilizador inseriu
599 MOV [R8+BateriaCarro],R3 ;atualizar o valor da bateria do carro do cliente com o valor que o utilizador inseriu
600 MOV R9,Display_DadosAlterados ;Mete no registo 9 o endereço do display a mostrar ao utilizador
601 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
602 JMP AlterarOuNao ;Voltar ao início da rotina
603
604 SaldoInvalido: ;a rotina chega aqui se o valor que o utilizador inseriu como possível saldo é inválido (valor negativo)
605 MOV R9, Display_SaldoInvalido ;Mete no registo 9 o endereço do display a mostrar ao utilizador
606 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
607 JMP AlterarOuNao ;volta ao início da rotina
608
609 BateriaInvalida: ;a rotina chega aqui se o valor que o utilizador inseriu como possível valor da bateria do carro é inválido (valor negativo ou superior a 100)
610 MOV R9, Display_BateriaInvalida ;Mete no registo 9 o endereço do display a mostrar ao utilizador
611 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
612 JMP AlterarOuNao ;volta ao início da rotina
613
614 FimAlterarBaseDeDados: ;Fim da rotina, chega aqui se o utilizador já não pretende fazer alterações à base de dados
615 POP R8 ;*****
616 POP R7 ;
617 POP R6 ;
618 POP R5 ;
619 POP R4 ; Retira da pilha os registos guardados no início da rotina
620 POP R3 ;
621 POP R2 ;
622 POP R1 ;
623 POP R0 ; *****
624 RET ;
625
626 ;*****
627 ; ROTINA InsereEnergia
628 ; Responsável por recarregar as baterias dos postos, se o utilizador assim desejar
629 ;*****
630
631 InsereEnergia:
632 PUSH R3 ;*****
633 PUSH R4 ;
634 PUSH R5 ; Guarda na pilha os registos alterados durante esta rotina
635 PUSH R6 ;
636 PUSH R7 ;
637 PUSH R8 ;*****
638
639 InicioInsereEnergia:
640 MOV R4,InputOpcao ;R4 contém o endereço de onde se lê o input de qual bateria carregar
641 MOV R9, Display_InsereEnergia ;Mete no registo 9 o endereço do display a mostrar ao utilizador
642 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
643 MOVB R4,[R6] ;R4 contém a seleção de qual bateria carregar, por parte do utilizador
644 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
645 MOV R6,R0 ;guarda em R6 o valor original da bateria do posto normal, caso o utilizador carregue a bateria com um valor inválido
646 MOV R7,R1 ;guarda em R7 o valor original da bateria do posto semirapido, caso o utilizador carregue a bateria com um valor inválido
647 MOV R8,R2 ;guarda em R8 o valor original da bateria do posto rapido, caso o utilizador carregue a bateria com um valor inválido
648 CMP R4,4 ;compara a opção escolhida pelo utilizador com o valor 4, que é a opção de avançar
649 JNE IncrementaNormal ;se não são iguais, o utilizador ou pretende carregar uma das baterias ou inseriu uma opção inválida, efetua-se o salto para verificar posteriormente
650 JMP FimInsereEnergia ;se são iguais, salta-se para o fim da função
651
652 IncrementaNormal:
653 CMP R4, CustoNormal ;compara a opção escolhida pelo utilizador com o valor do custoNormal (1)
654 JNE IncrementaSemiRapido ;se verificar-se que a bateria escolhida não é a normal, procede-se para a verificação das outras baterias
655 MOV R9, Display_InsereEnergiaQuanta ;Mete no registo 9 o endereço do display a mostrar ao utilizador
656 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
657 MOV R5, InputIncrementoBateria ;R5 contém o endereço de onde se lê o input de quanto carregar a bateria
658 MOV R3,[R5] ;R3 contém o valor a adicionar à bateria selecionada, se possível
659 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
660 CMP R3,0 ;compara o valor que o utilizador inseriu com 0
661 JLE IncrementaBateriaInvalido ;se o valor que o utilizador inseriu é negativo (menor que 0), efetua-se este salto pois o utilizador não pode tirar bateria aqui, só adicionar
662 ADD R0,R3 ;adicionamos a R0 (bateria normal) o valor que o utilizador inseriu
663 CMP R0,0 ;comparamos o valor da bateria após a adição com 0, para verificar se ocorreu overflow
664 JLT OverflowBateria ;se ocorrer overflow efetuar este salto para informar o utilizador
665 JMP AtualizaPostos ;se não ocorrer overflow, atualizamos os valores dos postos em memória
666
667 IncrementaSemiRapido:
668 CMP R4, CustoSemiRapido ;compara a opção escolhida pelo utilizador com o valor do custoSemiRapido (2)
669 JNE IncrementaRapido ;se verificar-se que a bateria escolhida não é a semirapida, procede-se para a verificação das outras baterias
670 MOV R9, Display_InsereEnergiaQuanta ;Mete no registo 9 o endereço do display a mostrar ao utilizador
671 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
672 MOV R5, InputIncrementoBateria ;R5 contém o endereço de onde se lê o input de quanto carregar a bateria
673 MOV R3,[R5] ;R3 contém o valor a adicionar à bateria selecionada, se possível
674 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
675 CMP R3,0 ;compara o valor que o utilizador inseriu com 0
676 JLE IncrementaBateriaInvalido ;se o valor que o utilizador inseriu é negativo (menor que 0), efetua-se este salto pois o utilizador não pode tirar bateria aqui, só adicionar
677 ADD R1,R3 ;adicionamos a R1 (bateria semirapido) o valor que o utilizador inseriu
678 CMP R1,0 ;comparamos o valor da bateria após a adição com 0, para verificar se ocorreu overflow
679 JLT OverflowBateria ;se ocorrer overflow efetuar este salto para informar o utilizador
680 JMP AtualizaPostos ;se não ocorrer overflow, atualizamos os valores dos postos em memória
681

```



```

682 IncrementaRapido:
683     CMP R4, CustoRapido                ;compara a opção escolhida pelo utilizador com o valor do custoRapido (3)
684     JNE OpcaoInvalida                  ;se verificar-se que a bateria escolhida não é a rápida, então o utilizador inseriu uma opção inválida
685     MOV R9, Display_InserirEnergiaQuanta
686     CALL RefreshDisplay                ;Mete no registo 9 o endereço do display a mostrar ao utilizador
687     MOV R5, InputIncrementoBateria     ;R5 contém o endereço de onde se lê o input de quanto carregar a bateria
688     MOV R3, [R5]                       ;R3 contém o valor a adicionar à bateria selecionada, se possível
689     CALLF LimpaPerifericosEntrada      ;limpa os endereços de onde se lê os inputs do utilizador
690     CMP R3, 0                          ;compara o valor que o utilizador inseriu com 0
691     JLT IncrementaBateriaInvalido      ;se o valor que o utilizador inseriu é negativo (menor que 0), efetua-se este salto pois o utilizador não pode tirar bateria aqui, só adicionar
692     ADD R2, R3                         ;adicionamos a R2 (bateria rápido) o valor que o utilizador inseriu
693     CMP R2, 0                          ;comparamos o valor da bateria após a adição com 0, para verificar se ocorreu overflow
694     JLT OverflowBateria                ;se ocorrer overflow efetuar este salto para informar o utilizador
695     JMP AtualizaPostos                 ;se não ocorrer overflow, atualizamos os valores dos postos em memória
696
697 OverflowBateria:
698     MOV R0, R6                         ;se ocorreu overflow, far-se reset dos valores das baterias com os valores guardados no início da rotina
699     MOV R1, R7                         ;reset do valor da bateria do posto normal
700     MOV R2, R8                         ;reset do valor da bateria do posto semirapido
701     MOV R9, Display_Overflow           ;Mete no registo 9, onde está o endereço do display a mostrar, o display que pretendemos mostrar
702     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
703     JMP InicioInserirEnergia          ;volta ao início da rotina
704
705 IncrementaBateriaInvalido:
706     MOV R9, Display_IncrementaBateriaInvalido
707     CALL RefreshDisplay                ;Mete no registo 9, onde está o endereço do display a mostrar, o display que pretendemos mostrar
708     JMP InicioInserirEnergia          ;Mostra o display metido anteriormente em R9 ao utilizador
709     ;volta ao início da rotina
710
711 OpcaoInvalida:
712     MOV R9, Display_OpcaoInvalida      ;Mete no registo 9 o endereço do display a mostrar ao utilizador
713     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
714     CALLF LimpaPerifericosEntrada      ;limpa os endereços de onde se lê os inputs do utilizador
715     JMP InicioInserirEnergia          ;voltar ao início da rotina
716
717 AtualizaPostos:
718     MOV R9, Display_BateriaCarregada   ;Mete no registo 9 o endereço do display a mostrar ao utilizador
719     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
720     JMP InicioInserirEnergia          ;Volta ao início da rotina
721
722 FimInserirEnergia:
723     POP R8                             ;*****
724     POP R7                             ;
725     POP R6                             ; Retira da pilha os registos guardados no início da rotina
726     POP R5                             ;
727     POP R4                             ;
728     RET                               ;*****
729
730 ;*****
731 ;      ROTINA NiveisDeEnergia
732 ;      Responsável por mostrar ao utilizador quais dos postos estão funcionais ou não funcionais
733 ;*****
734 NiveisDeEnergia:
735     PUSH R3                           ;*****
736     PUSH R4                           ;
737     PUSH R5                           ; Guarda na pilha os registos alterados durante esta rotina
738     PUSH R6                           ;
739     PUSH R7                           ;*****
740     MOV R10, 0                        ;R7 contém o nível mínimo que a bateria normal deve ter para ser considerada funcional (carregamento de uma hora)
741     MOV R4, Normal                    ;R8 contém o nível mínimo que a bateria semirapida deve ter para ser considerada funcional (carregamento de uma hora)
742     MOV R5, Semirapido                ;R9 contém o nível mínimo que a bateria rapida deve ter para ser considerada funcional (carregamento de uma hora)
743     MOV R6, Rapido                    ;compara o valor da bateria do posto normal com o seu valor mínimo
744     CMP R0, R4                        ;se verificarmos que a bateria normal tem o nível mínimo, verificamos as restantes
745     JGE VerificaSemiRapido            ;caso contrário, adicionamos 1 ao contador e verifica-se a bateria semirapida
746     ADD R3, 1                         ;
747
748 VerificaSemiRapido:
749     CMP R1, R5                        ;compara o valor da bateria do posto semirapido com o seu valor mínimo
750     JGE VerificaRapido                ;se verificarmos que a bateria semirapida tem o nível mínimo, verificamos a bateria rapida
751     ADD R3, 1                         ;caso contrário, adicionamos 1 ao contador e verifica-se a bateria rapida
752
753 VerificaRapido:
754     CMP R2, R6                        ;compara o valor da bateria do posto rapido com o seu valor mínimo
755     JGE FimNiveisDeEnergia            ;se verificarmos que a bateria rapida tem o nível mínimo, o posto esta operacional e salta-se para o fim da rotina
756     ADD R3, 1                         ;caso contrário, adicionamos 1 ao contador
757     CMP R3, 3                         ;compara-se o valor do contador com 3 (o número de baterias)
758     JNE FimNiveisDeEnergia            ;se o contador não é igual a 3, o posto está operacional e salta-se para o fim da função
759     MOV R10, -1                       ;caso contrário, mete-se o R10 a -1 para o programa não avançar para a verificação do utilizador
760     MOV R9, Display_CarregarPosto     ;Mete no registo 9 o endereço do display a mostrar ao utilizador
761     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
762
763 FimNiveisDeEnergia:
764     MOV R9, Display_NiveisDeEnergia    ;Mete no registo 9 o endereço do display a mostrar ao utilizador
765     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
766     CALL Display_NiveisDeEnergia_InserirInformacao ;chama a rotina que insere a informação sobre os estados dos postos no display
767     POP R7                             ;*****
768     POP R6                             ;
769     POP R5                             ; Retira da pilha os registos guardados no início da rotina
770     POP R4                             ;
771     POP R3                             ;*****
772     RET
773
774 ;*****
775 ;      ROTINA Verificacao_Cliente
776 ;      Responsável por receber o ID e código de segurança do utilizador e compará-los
777 ;      com os valores presentes na base de dados para ver se pode efetuar um carregamento
778 ;*****
779 Verificacao_Cliente:
780     PUSH R0                           ;*****
781     PUSH R1                           ;
782     PUSH R2                           ;
783     PUSH R3                           ;
784     PUSH R4                           ; Guarda na pilha os registos alterados durante esta rotina
785     PUSH R5                           ;
786     PUSH R6                           ;
787     PUSH R7                           ;
788     PUSH R8                           ;*****
789     MOV R3, InputID                   ;R3 é o endereço de onde se lê o ID do utilizador
790     MOV R4, InputCodSeguranca         ;R4 é o endereço de onde se lê o código de segurança do utilizador
791     MOV R9, Display_InputVerifyClient ;Mete no registo 9 o endereço do display a mostrar ao utilizador
792     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
793     MOV R5, [R3]                      ;R5 é o ID que o utilizador inseriu
794     MOV R6, [R4]                      ;R6 é o código de segurança que o utilizador inseriu
795     CALLF LimpaPerifericosEntrada      ;limpa os endereços de onde se lê os inputs do utilizador
796     MOV R0, Base_Tabela_Dados         ;move para R0 a base da tabela de dados, será a base dos dados do cliente que estamos a verificar e contém o ID deste
797     MOV R1, 0                         ;R1 será o índice
798     MOV R2, Proximo                   ;mete-se em R2 o valor a acrescentar ao início dos dados do cliente não verificado para avançar para o próximo
799     MOV R7, EnderecoTamanho           ;mete-se no registo 7 o endereço de onde ler o tamanho da base de dados
800     MOV R8, [R7]                      ;le-se o endereço em R7 para obter em R8 o tamanho da base de dados (nº de clientes)
801
802 Ciclo_Verify_Cliente:
803     MOV R10, -1                       ;mete-se em R10 o valor -1 para indicar, se chega-se ao fim da base de dados sem o utilizador ser verificado, que o utilizador não pode proceder para carregar o carro
804     CMP R1, R8                         ;compara-se o índice com o tamanho da base de dados
805     JEQ NaoVerificado                 ;se o índice é igual ao tamanho, chegou-se ao fim da base de dados sem o utilizador ser verificado
806     MOV R3, [R0]                      ;R3 tem o valor do ID da tabela de base de dados a verificar
807     MOV R4, [R0+CodSeguranca]         ;R4 tem o valor de código de segurança da tabela de base de dados a verificar
808     CMP R5, R3                         ;compara-se o ID inserido pelo utilizador com o ID da base de dados a verificar
809     JNE VerificacaoFalhada            ;se são diferentes avançar para o próximo, se possível
810     CMP R6, R4                         ;compara-se o código de segurança inserido pelo utilizador com o código de segurança da tabela de base de dados a verificar
811     JNE VerificacaoFalhada            ;se são diferentes avançar para o próximo, se possível
812     MOV R10, R2                       ;caso contrário, mete-se em R10 o valor para passar aos dados de um próximo cliente (que está em R2)
813     MUL R10, R1                       ;multiplica o valor de Proximo pelo índice, para se obter o início dos dados do cliente verificado relativamente ao início da base de dados
814     MOV R9, Display_VerificacaoSucesso ;Mete no registo 9 o endereço do display a mostrar ao utilizador
815     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
816     JMP FimVerificacao_Cliente        ;saltar para o fim da rotina pois o utilizador foi verificado com sucesso
817

```

```

818 VerificacaoFalhada: ;se os dados que o utilizador inseriu não coincidirem com os dados do cliente que estava a ser verificado efetua-se isto
819 ADD R1,1 ;adiciona-se um ao índice
820 ADD R0,R2 ;avança a base para o proximo cliente a verificar
821 JMP Ciclo_Verify_Cliente ;volta-se a verificar os dados do utilizador com o novo cliente a verificar
822
823 NaoVerificado: ;chega-se aqui se o utilizador não é verificado
824 MOV R8, Display_VerificacaoFalhada ;Mete no registo 9 o endereço do display a mostrar ao utilizador
825 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
826
827 FimVerificacao_Cliente:
828 POP R8 ;
829 POP R7 ;
830 POP R6 ;
831 POP R5 ;
832 POP R4 ;
833 POP R3 ; Retira da pilha os registos guardados no início da rotina
834 POP R2 ;
835 POP R1 ;
836 POP R0 ;
837 RET ;
838
839 ;
840 ; ROTINA Carregamento
841 ; Responsável por efetuar o carregamento do carro do cliente previamente autorizado
842 ; efetuando as verificações necessárias e atualizando os postos
843 ;
844 Carregamento:
845 PUSH R3 ;
846 PUSH R4 ;
847 PUSH R5 ;
848 PUSH R6 ; Guarda na pilha os registos alterados durante esta rotina
849 PUSH R7 ;
850 PUSH R8 ;
851 EscolhaCarregamento: ;VERIFICAR O TIPO DE CARREGAMENTO ESCOLHIDO PELO UTILIZADOR
852 MOV R9, Display_EscolheCarregamento ;mete no registo 9, onde está o endereço do display que pretendemos mostrar (Display_EscolheCarregamento)
853 MOV R5, InputOpcao ;coloca no registo 5 o endereço de onde ler o tipo de carregamento
854 CALL RefreshDisplay ;mostra ao utilizador o display metido anteriormente em R9
855 MOV R3, [R5] ;coloca no registo 3 o tipo de carregamento escolhido pelo utilizador
856 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
857 CMP R3, CustoNormal ;compara o registo 3 com o custoNormal (1, equivalente à opção)
858 JEQ EscolhaTempo ;se o valor do registo 3 for igual ao custoNormal, salta para o tag "EscolhaTempo" - ou seja, escolheu o carregamento normal
859 CMP R3, CustoSemiRapido ;compara o registo 3 com o custoSemiRapido (2, equivalente à opção)
860 JEQ EscolhaTempo ;se o valor do registo 3 for igual ao custoSemiRapido, salta para o tag "EscolhaTempo" - ou seja, escolheu o carregamento semi-rapido
861 CMP R3, CustoRapido ;compara o registo 3 com o custoRapido (3, equivalente à opção)
862 JEQ EscolhaTempo ;se o valor do registo 3 for igual ao custoRapido, salta para o tag "EscolhaTempo" - ou seja, escolheu o carregamento rapido
863 MOV R9, Display_OpcaoInvalida ;mete no registo 9, onde está o endereço do display que pretendemos mostrar (Display_OpcaoInvalida)
864 CALL RefreshDisplay ;mostra ao utilizador o display metido anteriormente em R9
865 JMP EscolhaCarregamento ;volta para o Display_inicial (volta a escolher o tipo carregamento)
866
867 EscolhaTempo: ;VERIFICAR O TEMPO ESCOLHIDO PELO UTILIZADOR
868 MOV R9, Display_EscolherTempo ;mete no registo 9, onde está o endereço do display que pretendemos mostrar
869 CALL RefreshDisplay ;mostra ao utilizador o display metido anteriormente em R9
870 MOV R5, InputTempo ;coloca no registo 5 o endereço de onde ler quanto tempo carregar
871 MOV R4, [R5] ;coloca no registo 4 o tempo escolhido pelo utilizador
872 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
873 CMP R4, 0 ;compara o valor do registo 4 com a constante 0
874 JGT CalculaTempo ;se o valor do registo 4 for superior a 0, ou seja, o tempo for superior a 0, salta para o tag "CalculaTempo"
875 MOV R9, Display_TempoInvalido ;mete no registo 9, onde está o endereço do display que pretendemos mostrar (Display_TempoInvalido)
876 CALL RefreshDisplay ;mostra o display metido anteriormente em R9 ao utilizador
877 JMP EscolhaTempo ;salta para o tag "EscolhaTempo" --> tempo inválido
878
879 CalculaTempo: ;CALCULA O TEMPO QUE DEMORARÁ A CARREGAR O CARRO
880 MOV R9,R4 ;coloca no registo 9 o valor do registo 4 (o tempo escolhido pelo utilizador)
881 MOV R5, Base_Tabela_Dados ;é colocado no registo 5 o valor o endereço do início da base de dados
882 ADD R5,R10 ;é adicionado ao registo 5 o valor do registo 10, ou seja, o índice do cliente
883 MOV R6, [R5+BateriaCarro] ;é colocado no registo 6, o valor da bateria do veículo do cliente
884 CMP R3, CustoNormal ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoNormal (1, equivalente à opção)
885 JEQ CalculaTempoNormal ;se forem iguais, salta para o tag "CalculaTempoNormal" - ou seja, o carregamento escolhido é o normal
886 CMP R3, CustoSemiRapido ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoSemiRapido (2, equivalente à opção)
887 JEQ CalculaTempoSemiRapido ;se forem iguais, salta para o tag "CalculaTempoSemiRapido" - ou seja, o carregamento escolhido é o semi-rapido
888 CMP R3, CustoRapido ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoRapido (3, equivalente à opção)
889 JEQ CalculaTempoRapido ;se forem iguais, salta para o tag "CalculaTempoRapido" - ou seja, o carregamento escolhido é o rapido
890
891 CalculaTempoNormal:
892 MOV R7, Normal ;coloca no registo 7 o valor de energia de um carregamento Normal/hora
893 MOV R8, 100 ;coloca no registo 8 a constante 100
894 SUB R8,R6 ;subtrai a 100 o valor de percentagem da bateria do carro
895 Ciclo_CTN:
896 CMP R8, 0 ;compara o valor do registo 8 com 0
897 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
898 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (20)
899 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
900 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
901 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
902 JMP Ciclo_CTN ;salta para o "tag" Ciclo_CTN
903
904 CalculaTempoSemiRapido:
905 MOV R7, Semirapido ;coloca no registo 7 o valor de energia de um carregamento SemiRapido/hora
906 MOV R8, 100 ;coloca no registo 8 a constante 100
907 SUB R8,R6 ;subtrai a 100 o valor de percentagem da bateria do carro
908 Ciclo_CTSR:
909 CMP R8, 0 ;compara o valor do registo 8 com 0
910 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
911 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (60)
912 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
913 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
914 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
915 JMP Ciclo_CTSR ;salta para o "tag" Ciclo_CTSR
916
917 CalculaTempoRapido:
918 MOV R7, Rapido ;coloca no registo 7 o valor de energia de um carregamento Rapido/hora
919 MOV R8, 100 ;coloca no registo 8 a constante 100
920 SUB R8,R6 ;subtrai a 100 o valor de percentagem da bateria do carro
921 Ciclo_CTR:
922 CMP R8, 0 ;compara o valor do registo 8 com 0
923 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
924 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (100)
925 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
926 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
927 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
928 JMP Ciclo_CTR ;salta para o "tag" Ciclo_CTR
929
930
931

```



```

932 VerificaTempo:
933     MOV R7,R9
934     SUB R7,R4
935     CMP R7,0
936     JEQ BateriaJACarregada
937     MOV R4,R7
938     CMP R3,CustoNormal
939     JNE VerificaEscolhaTempoSuperiorSemiRapido
940     MOV R5,R4
941     MOV R6,Normal
942     MUL R5,R6
943     CMP R5,R0
944     JGT SemBateriaParaCarregamento
945     JMP VerificaSaldo
946
947 VerificaEscolhaTempoSuperiorSemiRapido:
948     CMP R3,CustoSemiRapido
949     JNE VerificaEscolhaTempoSuperiorRapido
950     MOV R5,R4
951     MOV R6,SemiRapido
952     MUL R5,R6
953     CMP R5,R1
954     JGT SemBateriaParaCarregamento
955     JMP VerificaSaldo
956
957 VerificaEscolhaTempoSuperiorRapido:
958     MOV R5,R4
959     MOV R6,Rapido
960     MUL R5,R6
961     CMP R5,R2
962     JGT SemBateriaParaCarregamento
963     JMP VerificaSaldo
964
965 SemBateriaParaCarregamento:
966     MOV R9,Display_TempoUltrapassa
967     CALL RefreshDisplay
968     JMP FimCarregamento
969
970 VerificaSaldo:
971     MOV R4,R3
972     MOV R5,Base_Tabela_Dados
973     ADD R5,R0
974     MOV R6,[R5+Saldo]
975     CMP R4,R6
976     JLE ForneceEnergia
977     MOV R9,Display_SaldoInsuficiente
978     CALL RefreshDisplay
979     JMP FimCarregamento
980
981 ;armazena em R7 o valor em R9 (valor originalmente introduzido pelo utilizador)
982 ;subtrai a R7, valor de tempo originalmente introduzido, R4, para obter o tempo que realmente demorará
983 ;compara R7 com 0
984 ;se R7=0, então não é preciso carregar a bateria
985 ;armazena em R4 o valor em R7 (para futuras verificações)
986 ;compara o registo 3 com o valor do custoNormal (equivalente à opção)
987 ;se o valor do registo 3 for diferente do valor do registo 6, salta para o tag "VerificaEscolhaTempoSuperiorSemiRapido" - ou seja, é verificado se o tipo de carregamento não é normal
988 ;coloca no registo 5 o valor do registo 4 (o tempo que demorará o carregamento)
989 ;coloca no registo 6 o valor de energia de um carregamento Normal/hora
990 ;é multiplicado o valor do registo 5 com o valor do registo 6 --> registo 5 com a energia total do carregamento (com o valor do tempo inserido pelo utilizador)
991 ;compara o valor do registo 5 com o valor do registo 0 (valor da bateria Normal do posto)
992 ;se o valor do registo 5 é superior ao valor do registo 0, salta para o tag "SemBateriaParaCarregamento" - as opções escolhidas pelo utilizador irão descarregar a bateria do posto
993 ;salta para o tag "VerificaSaldo"
994
995 ;compara o valor do registo 3 com o valor do custoSemiRapido (equivalente à opção)
996 ;se o valor do registo 3 não for igual ao valor do registo 7, salta para o tag "VerificaEscolhaTempoSuperiorRapido" - ou seja, é verificado se o tipo de carregamento não é semi-rápido
997 ;coloca no registo 5 o valor do registo 4 (o tempo que demorará o carregamento)
998 ;coloca no registo 6 o valor da energia de um carregamento SemiRapido/hora
999 ;é multiplicado o valor do registo 5 com o valor do registo 6 --> registo 5 com a energia total do carregamento
1000 ;compara o valor do registo 5 com o valor do registo 1 (valor da bateria Semi-Rápida do posto)
1001 ;se o valor do registo 5 é superior ao valor do registo 1, salta para o tag "SemBateriaParaCarregamento" - as opções escolhidas pelo utilizador irão descarregar a bateria do posto
1002 ;salta para o tag "VerificaSaldo"
1003
1004 ;coloca no registo 5 o valor do registo 4 (o tempo que demorará o carregamento)
1005 ;coloca no registo 6 o valor da energia de um carregamento Rapido/hora
1006 ;é multiplicado o valor do registo 5 com o valor do registo 6 --> registo 5 com a energia total do carregamento
1007 ;compara o valor do registo 5 com o valor do registo 2 (valor da bateria Rápida do posto)
1008 ;se o valor do registo 5 é superior ao valor do registo 2, salta para o tag "SemBateriaParaCarregamento" - as opções escolhidas pelo utilizador irão descarregar a bateria do posto
1009 ;salta para o tag "VerificaSaldo"
1010
1011 ;mete no registo 9, onde está o endereço do display que pretendemos mostrar
1012 ;mostra ao utilizador o display metido anteriormente em R9
1013 ;salta para o tag "FimCarregamento"
1014
1015 ;VERIFICA SE O UTILIZADOR TEM SALDO SUFICIENTE PARA EFETUAR O CARREGAMENTO
1016 ;é multiplicado o valor do registo 4 com o valor do registo 3, ou seja, o tempo pelo custo/hora do carregamento --> registo 4 com o valor do custo do carregamento
1017 ;é colocado no registo 5 o valor o endereço do início da base de dados
1018 ;é adicionado ao registo 5 o valor do registo 10, ou seja, o índice do cliente
1019 ;é colocado no registo 6, o valor do saldo do cliente
1020 ;é comparado o valor do registo 4 com o valor do registo 6, ou seja, o custo do carregamento com o saldo do utilizador
1021 ;se o valor do registo 4 (o custo) for inferior ou igual ao do registo 6 (saldo), salta para o tag "ForneceEnergia", ou seja, o utilizador tem saldo suficiente
1022 ;mete no registo 9, onde está o endereço do que pretendemos mostrar (Display_SaldoInsuficiente)
1023 ;mostra ao utilizador o display metido anteriormente em R9
1024 ;salta para o tag "FimCarregamento", pois não se carregará o carro
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3
```

```

1070 CarregamentoEscrito:
1071     CALLF VerificaOK
1072     MOV R4, R7
1073     MUL R7, R3
1074     CALL Debito
1075     CMP R3, CustoNormal
1076     JEQ AtualizaPostoNormal
1077     CMP R3, CustoSemiRapido
1078     JEQ AtualizaPostoSemiRapido
1079     CMP R3, CustoRapido
1080     JEQ AtualizaPostoRapido
1081
1082
1083 Debito:
1084     MOV R6, [R5+Saldo]
1085     SUB R6, R7
1086     MOV [R5+Saldo], R6
1087     MOV R9, R7
1088     CALL RefreshDisplay
1089     MOV R8, R6
1090     MOV R9, 57
1091     CALLF EscreveValores
1092     MOV R8, R7
1093     MOV R9, 89
1094     CALLF EscreveValores
1095     CALLF VerificaOK
1096     RET
1097
1098 AtualizaPostoNormal:
1099     MOV R6, Normal
1100     MUL R6, R4
1101     SUB R0, R6
1102     JMP CarregamentoConcluido
1103
1104 AtualizaPostoSemiRapido:
1105     MOV R6, SemiRapido
1106     MUL R6, R4
1107     SUB R0, R6
1108     JMP CarregamentoConcluido
1109
1110 AtualizaPostoRapido:
1111     MOV R6, Rapido
1112     MUL R6, R4
1113     SUB R2, R6
1114
1115 CarregamentoConcluido:
1116     MOV R9, DisplayCarregamentoConcluido
1117     CALL RefreshDisplay
1118
1119
1120 FimCarregamento:
1121     POP R8
1122     POP R7
1123     POP R6
1124     POP R5
1125     POP R4
1126     POP R3
1127     RET
1128
1129 ;*****
1130 ;                               ROTINA VerificaOK
1131 ;                               Responsável por registar que o utilizador "carregou" no OK
1132 ;*****
1133 VerificaOK:
1134     PUSH R0
1135     PUSH R1
1136
1137 CicloVerOK:
1138     MOV R0, OK
1139     MOV R1, [R0]
1140     CMP R1, 1
1141     JNE CicloVerOK
1142     MOV R1, 0
1143     MOV R0, R1
1144     POP R1
1145     POP R0
1146     RETF
1147
1148 ;*****
1149 ;                               ROTINA RefreshDisplay
1150 ;                               Responsável por atualizar o display que o utilizador vê consoante o
1151 ;                               display a mostrar, indicado pelo registo R9
1152 ;*****
1153 RefreshDisplay:
1154     PUSH R0
1155     PUSH R1
1156     PUSH R2
1157     PUSH R3
1158     PUSH R4
1159     MOV R4, R9
1160     MOV R0, InicioDisplay
1161     MOV R1, FimDisplay
1162
1163
1164 Ciclo_RefreshDisplay:
1165     MOV R2, [R9]
1166     MOV [R0], R2
1167     ADD R0, 2
1168     ADD R9, 2
1169     CMP R0, R1
1170     JLE Ciclo_RefreshDisplay
1171     MOV R3, Display_NiveisDeEnergia
1172     CMP R3, R4
1173     JEQ FimRefreshDisplay
1174     MOV R3, Display_Debito
1175     CMP R3, R4
1176     JEQ FimRefreshDisplay
1177     MOV R3, Display_InfoCarregamento
1178     CMP R3, R4
1179     JEQ FimRefreshDisplay
1180     CALLF VerificaOK
1181
1182 FimRefreshDisplay:
1183     POP R4
1184     POP R3
1185     POP R2
1186     POP R1
1187     POP R0
1188     RET

```

```

1189 ;*****
1190 ;          ROTINA LimpaPerifericosEntrada
1191 ;          Esta rotina, como o nome indica, "limpa" os endereços de onde se lê
1192 ;          os dados que o utilizador insira
1193 ;*****
1194 LimpaPerifericosEntrada:
1195     PUSH R0 ;*****
1196     PUSH R1 ;
1197     PUSH R2 ;
1198     PUSH R3 ;
1199     PUSH R4 ; Guarda na pilha os registos alterados durante esta rotina
1200     PUSH R5 ;
1201     PUSH R6 ;
1202     PUSH R7 ;*****
1203     MOV R0, InputCodSeguranca
1204     MOV R1, InputID
1205     MOV R2, InputIncrementoBateria
1206     MOV R3, InputTempo ; Endereços dos periféricos a "limpar" (por a 0)
1207     MOV R4, InputOpcao ;
1208     MOV R5, InputSaldo ;
1209     MOV R6, InputBateria ;*****
1210     MOV R7, 0 ;mete-se no registo 7 o 0 para limpar os periféricos
1211     MOV [R0],R7 ;*****
1212     MOV [R1],R7 ;
1213     MOV [R2],R7 ;
1214     MOV [R3],R7 ; Limpeza dos periféricos de entrada
1215     MOV [R4],R7 ;
1216     MOV [R5],R7 ;
1217     MOV [R6],R7 ;*****
1218     POP R7 ;
1219     POP R6 ;
1220     POP R5 ;
1221     POP R4 ; Retira da pilha os registos guardados no início da rotina
1222     POP R3 ;
1223     POP R2 ;
1224     POP R1 ;
1225     POP R0 ;*****
1226     RETF
1227
1228 ;*****
1229 ;          ROTINA LimpaDisplay
1230 ;          Como o nome indica, "limpa" o display mostrado ao utilizador,
1231 ;          preenchendo-o com espaços
1232 ;*****
1233 LimpaDisplay:
1234     PUSH R0 ;*****
1235     PUSH R1 ; Guarda na pilha os registos alterados durante esta rotina
1236     PUSH R2 ;*****
1237     MOV R0,InicioDisplay ;mete-se em R0 o início do display
1238     MOV R1,FimDisplay ;mete-se em R1 o fim do display
1239     MOV R2,32 ;mete-se em R2 um caractêr vazio (valor em código ASCII de 32)
1240
1241 Ciclo_LimpaDisplay:
1242     MOVB [R0],R2 ;mete-mos no byte endereçado por R0 o caractêr vazio
1243     ADD R0,1 ;avança-se para o próximo byte
1244     CMP R0,R1 ;verifica-se se chegou-se ao fim do display
1245     JLE Ciclo_LimpaDisplay ;se não, volta-se ao início do ciclo
1246     POP R2 ;*****
1247     POP R1 ; Retira da pilha os registos guardados no início da rotina
1248     POP R0 ;*****
1249     RETF
1250
1251 ;*****
1252 ;          ROTINA Display_NiveisDeEnergia_InserirInformacao
1253 ;          Responsável por verificar os valores das baterias dos postos e escrever
1254 ;          no display mostrado ao utilizador o estado dos postos conforme os valores
1255 ;          das baterias (Funcional indica que a bateria está acima do valor mínimo de um carregamento de uma hora)
1256 ;*****
1257 Display_NiveisDeEnergia_InserirInformacao:
1258     PUSH R3 ;*****
1259     PUSH R4 ;
1260     PUSH R5 ; Guarda na pilha os registos alterados durante esta rotina
1261     PUSH R6 ;*****
1262     MOV R5, InicioDisplay ;mete em R5 o início do display
1263     MOV R4,Normal ;mete-se em R4 o valor mínimo da bateria que o posto normal tem de ter
1264     MOV R6,23 ;mete-se em R6 o aumento ao início do display, para escrever a informação no lugar certo
1265     ADD R6,R5 ;adiciona-se a R6 o início do display (R5) para escrever o estado do posto no lugar certo
1266     CMP R0,R4 ;compara-se a bateria do posto normal (R0) com o valor mínimo (R4)
1267     JLT NaoFuncionalNormal ;se é menor, efetua-se este salto para escrever essa informação
1268     CALLF EscreveFuncional ;se é maior ou igual, o posto está funcional e escreve-se essa informação
1269     JMP InfoSemiRapido ;salta-se para escrever o estado do posto semi-rápido
1270
1271 NaoFuncionalNormal:
1272     CALLF EscreveNao_Func ;como o posto não está funcional (se chegou aqui), escreve-se essa informação
1273
1274 InfoSemiRapido:
1275     MOV R4,Semirapido ;mete-se em R4 o valor mínimo da bateria que o posto semi-rápido tem de ter
1276     CMP R1,R4 ;compara-se a bateria do posto semi-rápido (R1) com o valor mínimo (R4)
1277     JLT NaoFuncionalSemiRapido ;se é menor, efetua-se este salto para escrever essa informação
1278     MOV R6,48 ;se é maior ou igual, mete-se em R6 o avanço relativo ao início do display para escrever a informação no lugar certo
1279     ADD R6,R5 ;acrescenta-se a R6 o início do display
1280     CALLF EscreveFuncional ;se o salto anterior não foi efetuado, então o posto está funcional e escreve-se essa informação no display
1281     JMP InfoRapido ;avança-se para escrever a informação relativa ao posto Rápido
1282
1283 NaoFuncionalSemiRapido:
1284     MOV R6,43 ;como é menor, mete-se em R6 o avanço relativo ao início do display para escrever a informação no lugar certo
1285     ADD R6,R5 ;acrescenta-se a R6 o início do display
1286     MOV R3,2 ;mete-se em R3 quantos espaços dar para avançar para a próxima linha, depois de escrever o "Nao"
1287     CALLF EscreveNao ;como não está funcional, escreve-se "Nao Funcional"
1288     CALLF EscreveFuncional ;como não está funcional, escreve-se "Nao Funcional"
1289

```

```

1290 InfoRapido:
1291     MOV R4,Rapido                                ;mete-se em R4 o valor mínimo da bateria que o posto rápido tem de ter
1292     CMP R2,R4                                    ;compara-se a bateria do posto rápido (R2) com o valor mínimo (R4)
1293     JLT NaoFuncionalRapido                      ;se é menor, efetua-se este salto para escrever a informação que não está funcional
1294     MOV R6,80                                    ;se é maior ou igual, mete-se em R6 o avanço relativo ao início do display para escrever a informação no lugar certo
1295     ADD R6,R5                                    ;acrescenta-se a R6 o início do display
1296     CALLF EscreveFuncional                      ;escreve-se que o posto não está funcional
1297     JMP FimInformacoesPosto                    ;salta-se par o fim da função
1298
1299 NaoFuncionalRapido:
1300     MOV R6,73                                    ;mete-se em R6 o avanço relativo ao início do display para escrever a informação no lugar certo
1301     ADD R6,R5                                    ;acrescenta-se a R6 o início do display
1302     MOV R3,4                                    ;mete-se em R3 o número de espaços a dar depois de escrever "Nao" para avançar para a próxima linha
1303     CALLF EscreveNao                            ;como não está funcional, escreve-se "Nao Funcional"
1304     CALLF EscreveFuncional                      ;como não está funcional, escreve-se "Nao Funcional"
1305
1306 FimInformacoesPosto:
1307     CALLF VerificaOK                            ;verifica que o utilizador quer proceder
1308     POP R6                                     ;*****
1309     POP R5                                     ;
1310     POP R4                                     ; Retira da pilha os registos guardados no início da rotina
1311     POP R3                                     ;*****
1312     RET
1313
1314 ;*****
1315 ;           ROTINA EscreveNao
1316 ;           Como o nome indica, esta rotina é responsável por escrever
1317 ;           a palavra "Nao" através do código ASCII e avançar para a
1318 ;           próxima linha, através de espaços
1319 ;           R4 - carater a mostrar no display,
1320 ;           R6 - endereço onde escrever o carater (R4)
1321 ;*****
1322 EscreveNao:
1323     MOV R4,78                                    ;código ASCII de "N"
1324     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1325     ADD R6,1                                    ;avança para o próximo byte
1326     MOV R4,97                                    ;código ASCII de "a"
1327     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1328     ADD R6,1                                    ;avança para o próximo byte
1329     MOV R4,111                                   ;código ASCII de "o"
1330     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1331     ADD R6,1                                    ;avança para o próximo byte
1332
1333 CicloEspacos:
1334     MOV R4,32                                    ;ciclo para escrever o número de espaços necessários para avançar para a próxima linha
1335     MOVB [R6],R4                                ;código ASCII do espaço
1336     ADD R6,1                                    ;escreve-se no display o carater dado por R4
1337     SUB R3,1                                    ;avança para o próximo byte
1338     CMP R3,0                                    ;subtrai 1 a R3 (o contador de espaços)
1339     JNE CicloEspacos                          ;verifica-se se R3 já chegou a 0
1340     RETF                                       ;se ainda não chegou, volta-se ao início do ciclo para escrever mais um espaço
1341     ;se chegou, a rotina terminou
1342
1343 ;*****
1344 ;           ROTINA EscreveFuncional
1345 ;           Semelhante à anterior, só que escreve "Funcional" em vez de "Nao"
1346 ;           e não faz parágrafo, ao contrário da anterior
1347 ;           R4 - carater a mostrar no display,
1348 ;           R6 - endereço onde escrever o carater (R4)
1349 ;*****
1350 EscreveFuncional:
1351     MOV R4,70                                    ;código ASCII de "F"
1352     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1353     ADD R6,1                                    ;avança para o próximo byte
1354     MOV R4,117                                   ;código ASCII de "u"
1355     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1356     ADD R6,1                                    ;avança para o próximo byte
1357     MOV R4,110                                   ;código ASCII de "n"
1358     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1359     ADD R6,1                                    ;avança para o próximo byte
1360     MOV R4,99                                    ;código ASCII de "c"
1361     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1362     ADD R6,1                                    ;avança para o próximo byte
1363     MOV R4,105                                   ;código ASCII de "i"
1364     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1365     ADD R6,1                                    ;avança para o próximo byte
1366     MOV R4,111                                   ;código ASCII de "o"
1367     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1368     ADD R6,1                                    ;avança para o próximo byte
1369     MOV R4,110                                   ;código ASCII de "n"
1370     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1371     ADD R6,1                                    ;avança para o próximo byte
1372     MOV R4,97                                    ;código ASCII de "a"
1373     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1374     ADD R6,1                                    ;avança para o próximo byte
1375     MOV R4,108                                   ;código ASCII de "l"
1376     MOVB [R6],R4                                ;escreve-se no display o carater dado por R4
1377     RETF

```

```

1378 ;*****
1379 ;
1380 ;           ROTINA EscreveNao_Func
1381 ; Semelhante às anteriores, só que esta escreve "Nao Func.".
1382 ; Só é utilizada para demonstrar que o posto normal não está operacional
1383 ; R4 - carater a mostrar no display,
1384 ; R6 - endereço onde escrever o carater (R4)
1385 ;*****
1386 EscreveNao_Func:
1387     MOV R4,78                ;código ASCII de "N"
1388     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1389     ADD R6,1                ;avança para o próximo byte
1390     MOV R4,97               ;código ASCII de "a"
1391     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1392     ADD R6,1                ;avança para o próximo byte
1393     MOV R4,111              ;código ASCII de "o"
1394     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1395     MOV R4,32               ;código ASCII do espaço
1396     ADD R6,1                ;avança para o próximo byte
1397     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1398     MOV R4,70               ;código ASCII de "f"
1399     ADD R6,1                ;avança para o próximo byte
1400     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1401     ADD R6,1                ;avança para o próximo byte
1402     MOV R4,117              ;código ASCII de "u"
1403     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1404     ADD R6,1                ;avança para o próximo byte
1405     MOV R4,110              ;código ASCII de "n"
1406     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1407     ADD R6,1                ;avança para o próximo byte
1408     MOV R4,99               ;código ASCII de "c"
1409     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1410     ADD R6,1                ;avança para o próximo byte
1411     MOV R4,46               ;código ASCII do ponto "."
1412     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1413     RETF
1414
1415 ;*****
1416 ;
1417 ;           ROTINA EscreveNormal
1418 ; Semelhante às anteriores, só que esta escreve "Normal".
1419 ; Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso Normal
1420 ; R4 - carater a mostrar no display,
1421 ; R6 - endereço onde escrever o carater (R4)
1422 ;*****
1423 EscreveNormal:
1424     MOV R4,78                ;código ASCII de "N"
1425     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1426     ADD R6,1                ;avança para o próximo byte
1427     MOV R4,111              ;código ASCII de "o"
1428     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1429     ADD R6,1                ;avança para o próximo byte
1430     MOV R4,114              ;código ASCII de "x"
1431     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1432     ADD R6,1                ;avança para o próximo byte
1433     MOV R4,109              ;código ASCII de "m"
1434     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1435     ADD R6,1                ;avança para o próximo byte
1436     MOV R4,97               ;código ASCII de "a"
1437     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1438     ADD R6,1                ;avança para o próximo byte
1439     MOV R4,108              ;código ASCII de "l"
1440     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1441     RETF
1442
1443 ;*****
1444 ;
1445 ;           ROTINA EscreveSemiRapido
1446 ; Semelhante às anteriores, só que esta escreve "Semi-Rapido".
1447 ; Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso SemiRapido
1448 ; R4 - carater a mostrar no display,
1449 ; R6 - endereço onde escrever o carater (R4)
1450 ;*****
1451 EscreveSemiRapido:
1452     MOV R4,83                ;código ASCII de "S"
1453     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1454     ADD R6,1                ;avança para o próximo byte
1455     MOV R4,101              ;código ASCII de "e"
1456     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1457     ADD R6,1                ;avança para o próximo byte
1458     MOV R4,109              ;código ASCII de "m"
1459     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1460     ADD R6,1                ;avança para o próximo byte
1461     MOV R4,105              ;código ASCII de "i"
1462     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1463     ADD R6,1                ;avança para o próximo byte
1464     MOV R4,45               ;código ASCII de "-", o hifen
1465     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1466     ADD R6,1                ;avança para o próximo byte
1467     MOV R4,82               ;código ASCII de "R"
1468     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1469     ADD R6,1                ;avança para o próximo byte
1470     MOV R4,97               ;código ASCII de "a"
1471     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1472     ADD R6,1                ;avança para o próximo byte
1473     MOV R4,112              ;código ASCII de "p"
1474     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1475     ADD R6,1                ;avança para o próximo byte
1476     MOV R4,105              ;código ASCII de "i"
1477     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1478     ADD R6,1                ;avança para o próximo byte
1479     MOV R4,100              ;código ASCII de "d"
1480     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1481     ADD R6,1                ;avança para o próximo byte
1482     MOV R4,111              ;código ASCII de "o"
1483     MOVB [R6],R4            ;escreve-se no display o carater dado por R4
1484     RETF
1485

```



```

1486 ;*****
1487 ;
1488 ;           ROTINA EscreveRapido
1489 ; Semelhante às anteriores, só que esta escreve "Rapido".
1490 ; Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso Rapido
1491 ; R4 - carater a mostrar no display,
1492 ; R6 - endereço onde escrever o carater (R4)
1492 ;*****
1493 EscreveRapido:
1494     MOV R4,82 ;código ASCII de "R"
1495     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1496     ADD R6,1 ;avança para o próximo byte
1497     MOV R4,97 ;código ASCII de "a"
1498     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1499     ADD R6,1 ;avança para o próximo byte
1500     MOV R4,112 ;código ASCII de "p"
1501     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1502     ADD R6,1 ;avança para o próximo byte
1503     MOV R4,105 ;código ASCII de "i"
1504     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1505     ADD R6,1 ;avança para o próximo byte
1506     MOV R4,100 ;código ASCII de "d"
1507     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1508     ADD R6,1 ;avança para o próximo byte
1509     MOV R4,111 ;código ASCII de "o"
1510     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1511     RETF
1512
1514 ;*****
1515 ;           ROTINA EscreveValores
1516 ; Rotina responsável por escrever valores numéricos no display.
1517 ; R8 - valor numerico a mostrar no display
1518 ; R9 - número de bytes a avançar para escrever o valor no lugar certo
1519 ;*****
1520 EscreveValores:
1521     PUSH R0 ;*****
1522     PUSH R1 ;
1523     PUSH R2 ; Guarda na pilha os registos alterados durante esta rotina
1524     PUSH R3 ;*****
1525     MOV R0,InicioDisplay ; mete em R0 o início do display
1526     ADD R0, R9 ; adiciona ao início do display o valor em R9 (número de bytes a avançar para escrever o valor no lugar certo)
1527     MOV R1,10 ;mete em R1 o valor 10, para fins de cálculo de resto e divisão
1528     MOV R2,48 ;mete em R2 o incremento ao valor numerico para obter o carater ASCII do valor numerico
1529
1530 CicloEscreveValores:
1531     MOV R3,R8 ;guarda em R3 uma cópia do valor (R8)
1532     MOD R3,R1 ;calcula-se o resto de R3 por R1 (valor por 10), para obter o valor a mostrar no display
1533     ADD R3,R2 ;adicionar ao valor obtido R2 (para obter o código ASCII do valor)
1534     MOVB [R0],R3 ;escrever o valor numérico no display (byte endereçado por R0)
1535     SUB R0,1 ;como estamos a escrever o valor da direita para a esquerda, subtrair 1 a R0
1536     DIV R8,R1 ;divide-se o valor por 10, para avançar para mostrar o próximo valor (possivelmente)
1537     CMP R8,0 ;compara o valor numérico com 0
1538     JEQ FimEscreveValores ;se chegou a 0, já não há mais caracteres a mostrar, salta-se para o fim da função
1539     JMP CicloEscreveValores ;se não chegou a 0, volta-se ao início do ciclo
1540
1541 FimEscreveValores:
1542     POP R3 ;*****
1543     POP R2 ; Retira da pilha os registos guardados no início da rotina
1544     POP R1 ;
1545     POP R0 ;*****
1546     RETF

```