

**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 Descarregamento do Posto .....	11
7.5.	Fluxograma da Verificação do Saldo .....	13
7.6.	Fluxograma do Fornecimento de Energia .....	14
7.7.	Fluxograma do Débito .....	15
7.8.	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.

Inicialmente, o posto encontra-se desligado e, após ligar a máquina, é mostrado no display a opção do utilizador alterar a base de dados.

## 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 até as suas opções serem 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, caso não esteja, 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.

## 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 verifica-se se o utilizador desligou a máquina.

## 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	1	2	3	4	0	F	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.

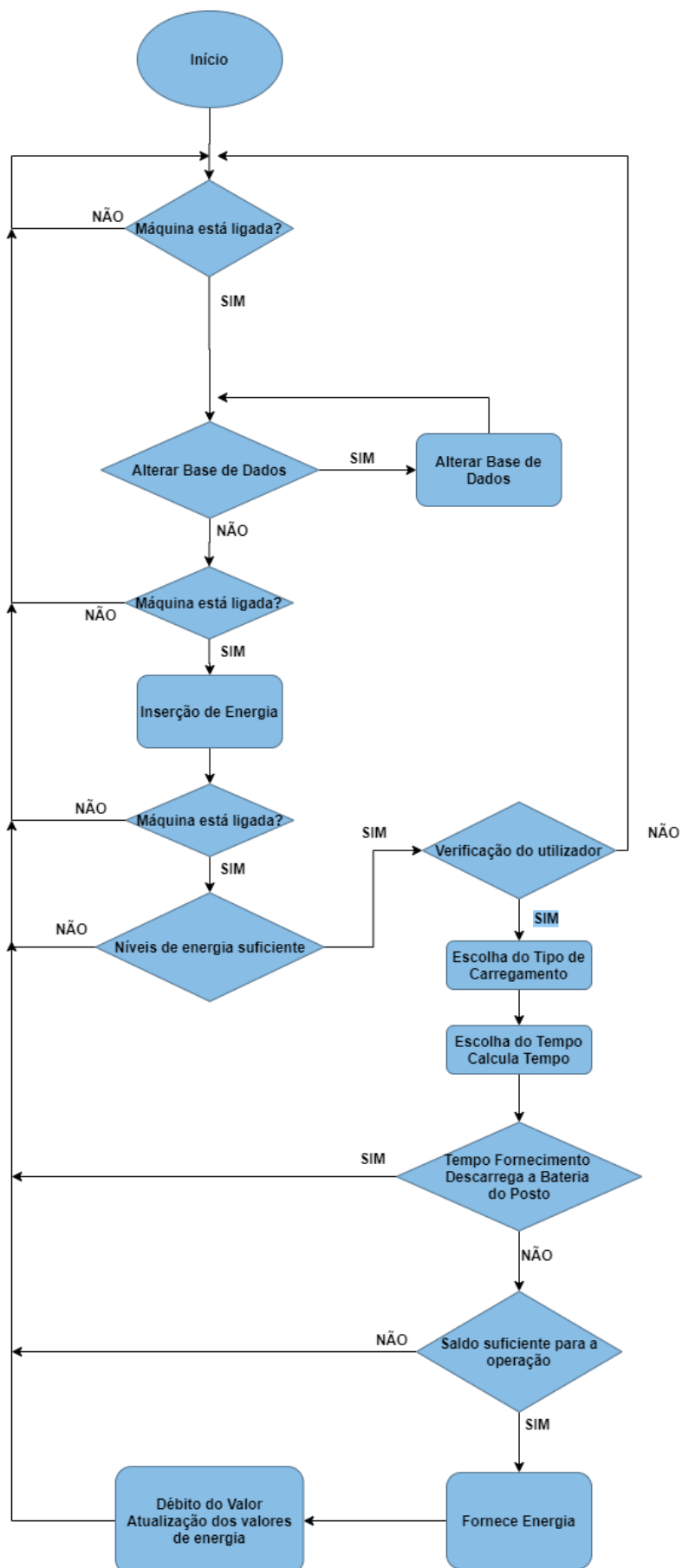
Em geral, este trabalho foi uma boa ferramenta de aprendizagem para uma melhor compreensão da arquitetura de computadores e o funcionamento destes.

## 6. Bibliografia

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

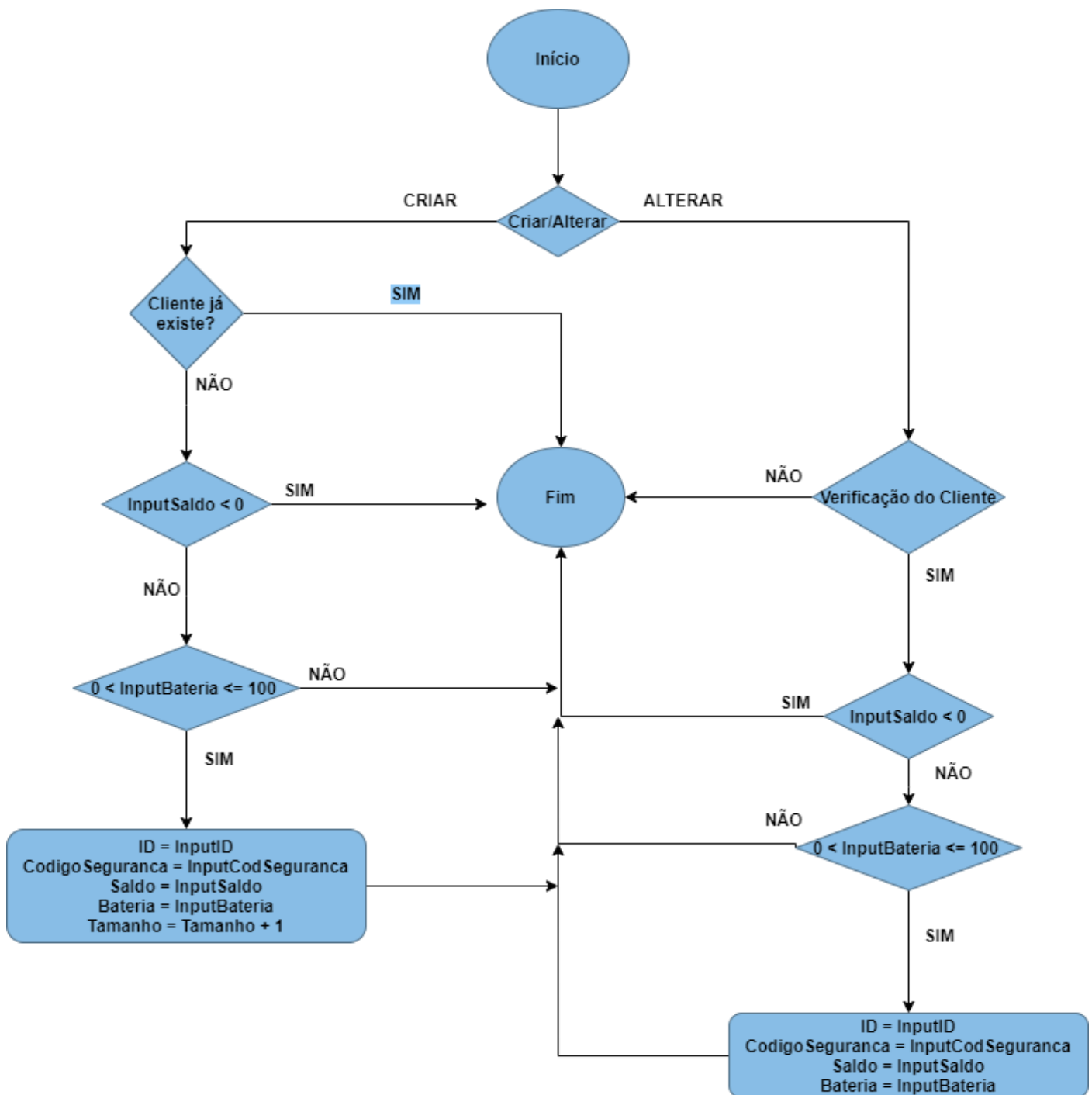
## 7. Anexo A

### 7.1. Fluxograma do Programa Geral

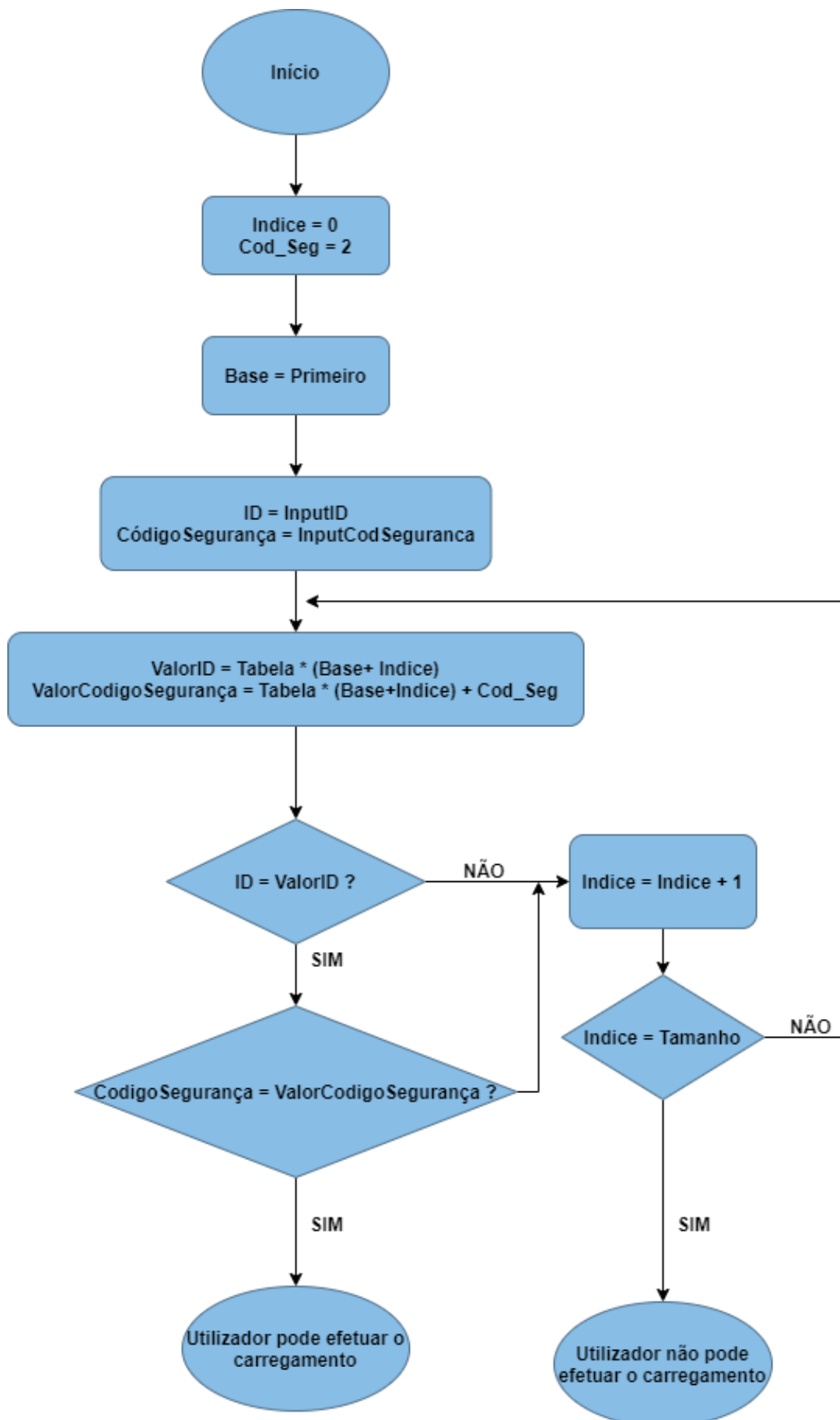




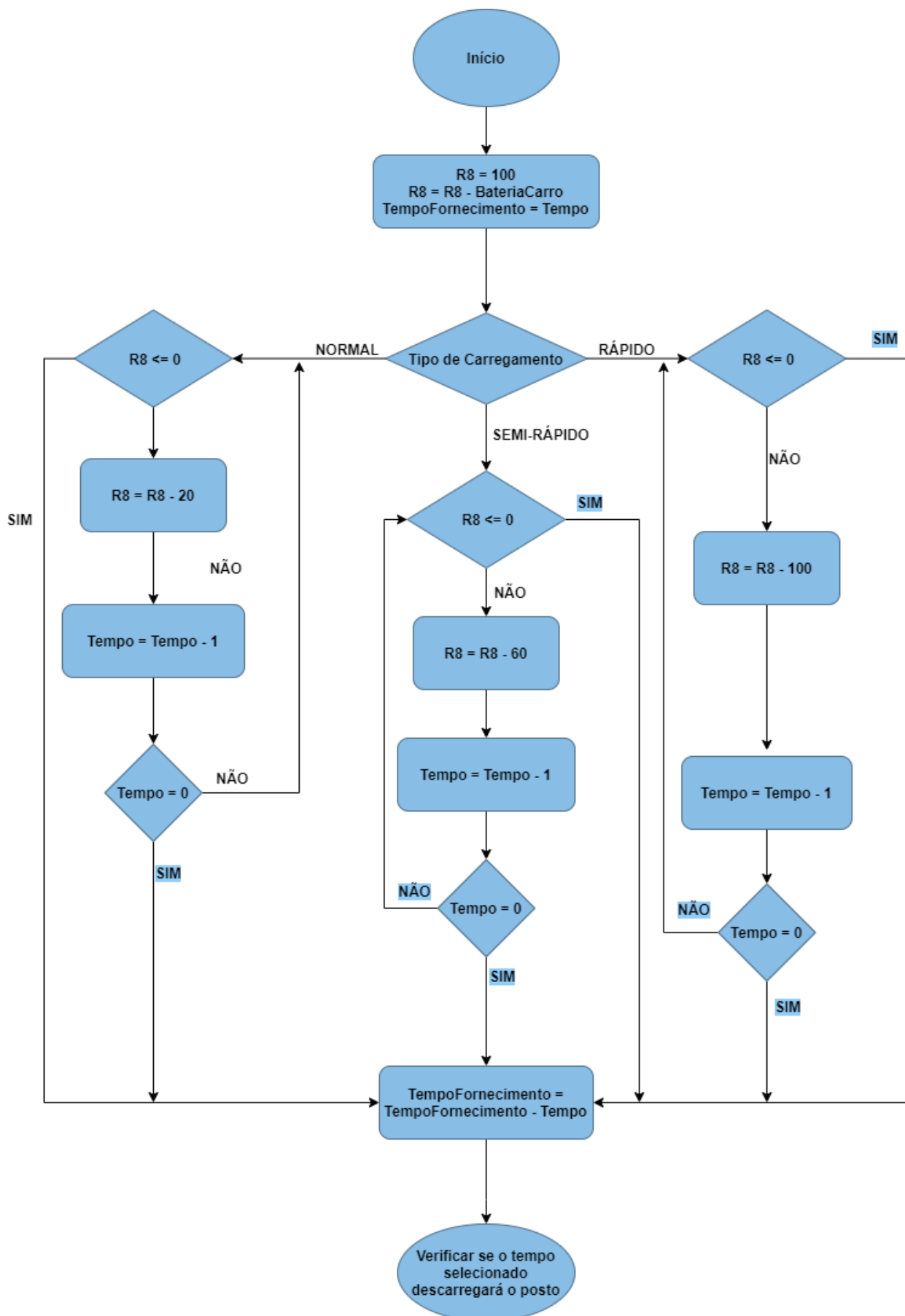
## 7.2. Fluxograma da Alteração da Base de Dados



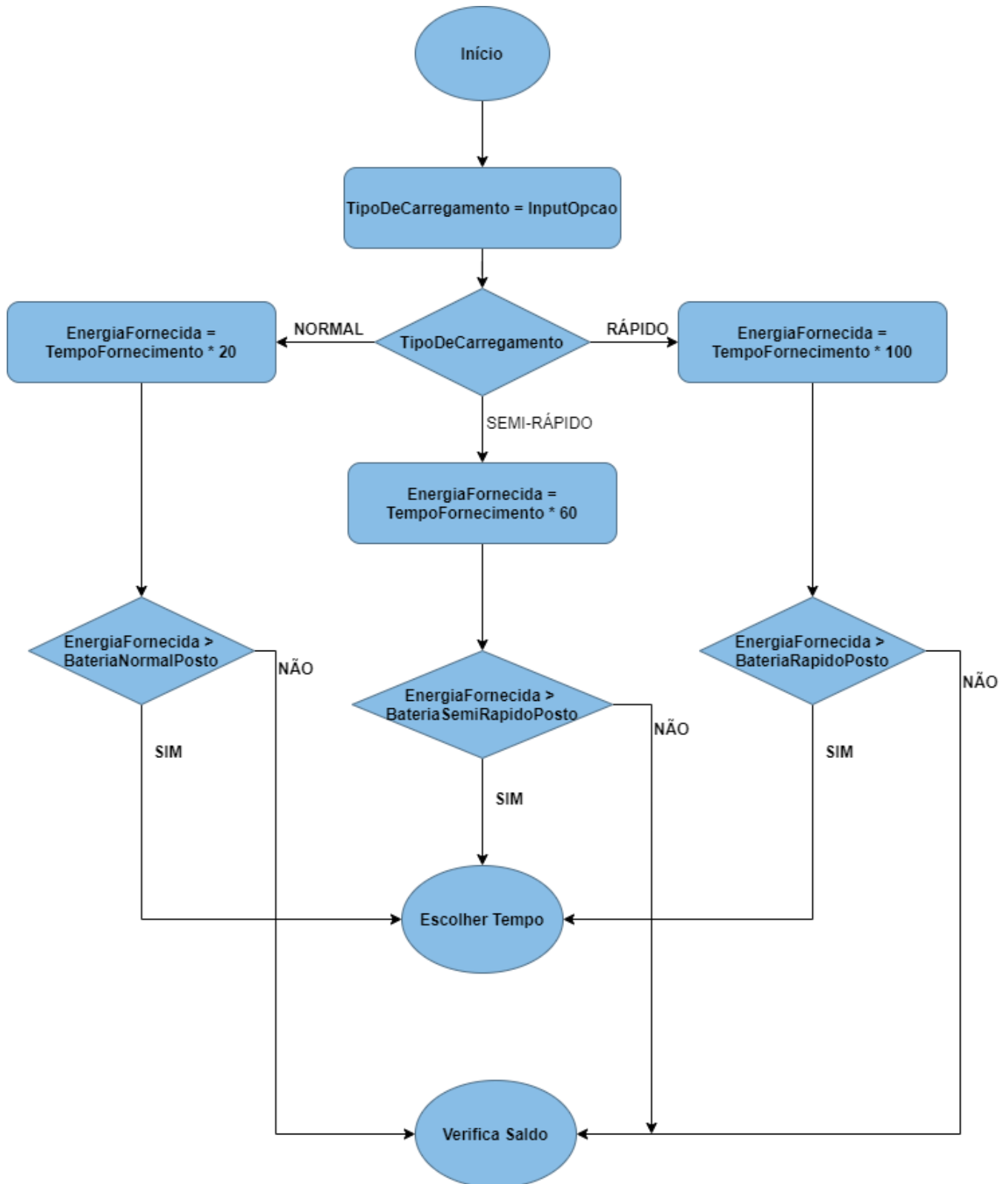
### 7.3. Fluxograma da Verificação do Utilizador



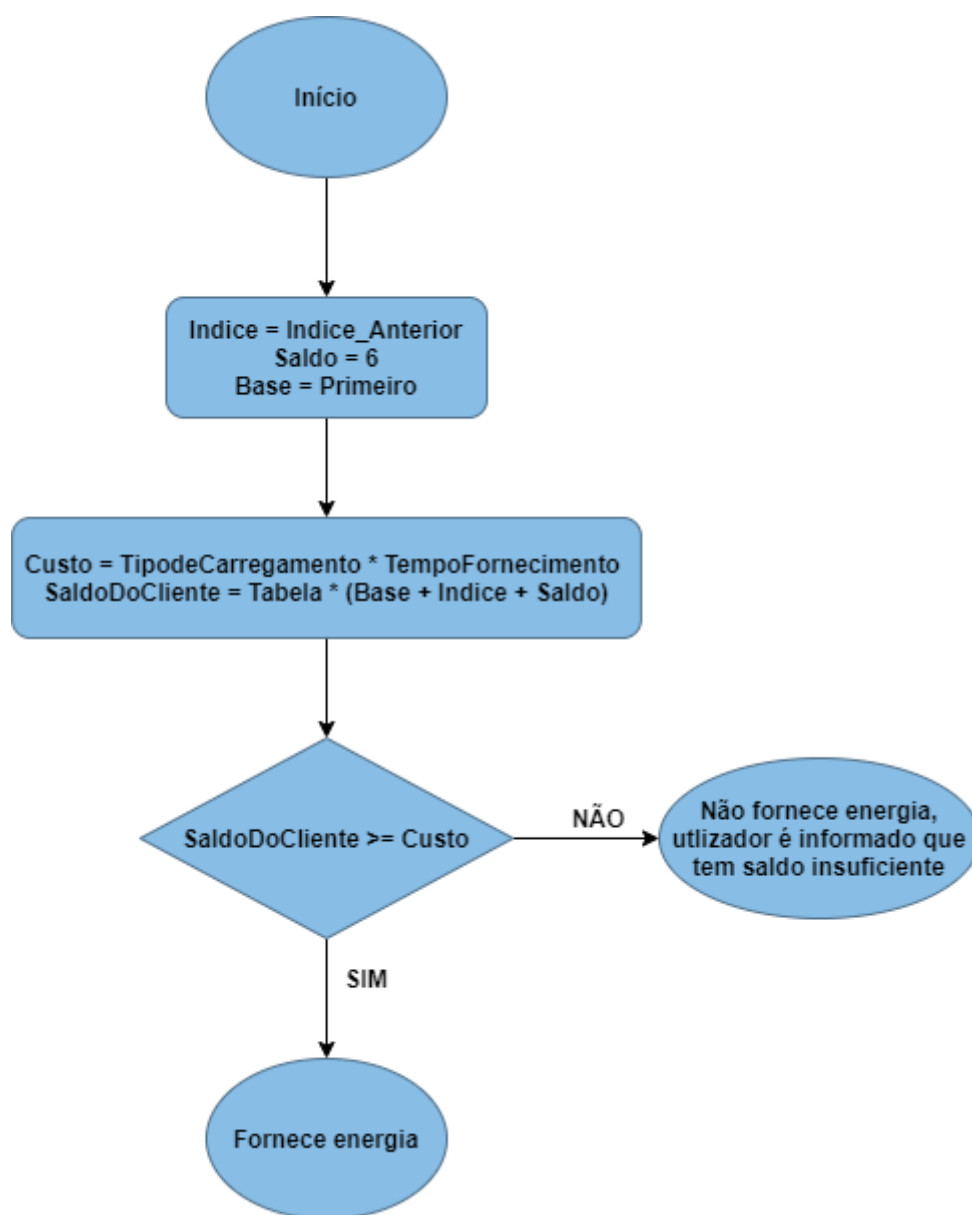
## 7.4. Fluxograma do Calcula Tempo



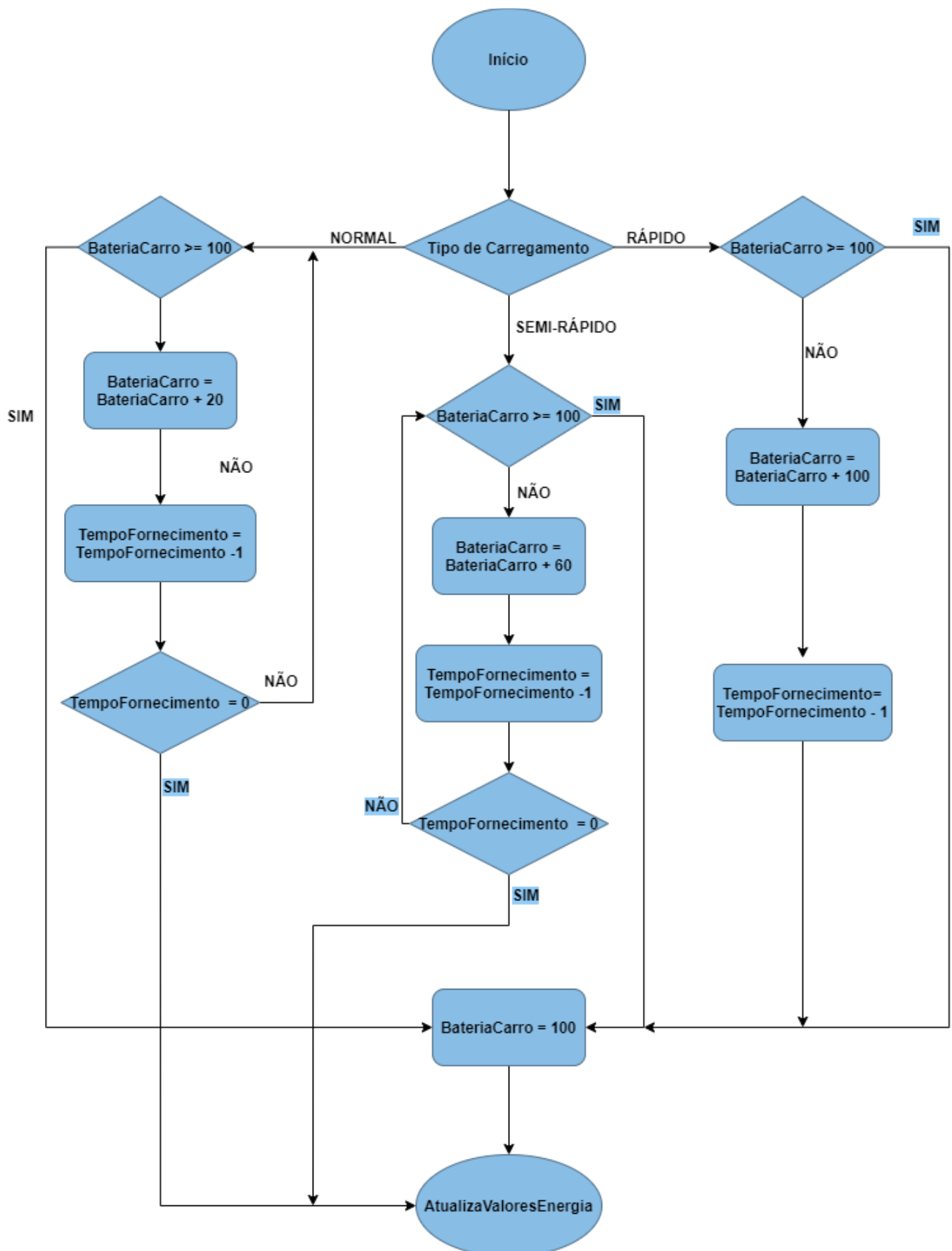
## 7.5. Fluxograma do Descarregamento do Posto



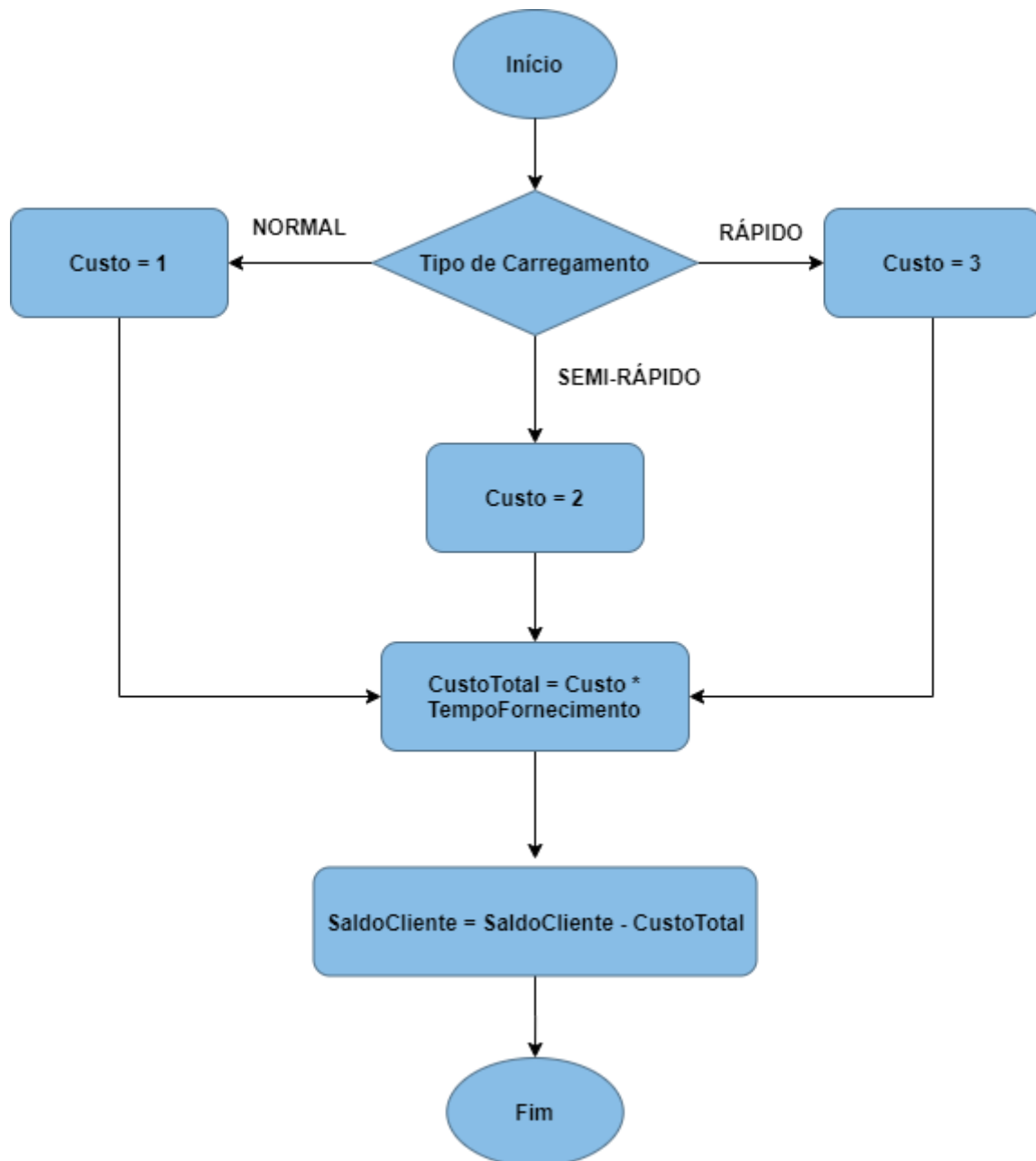
## 7.6. Fluxograma da Verificação do Saldo



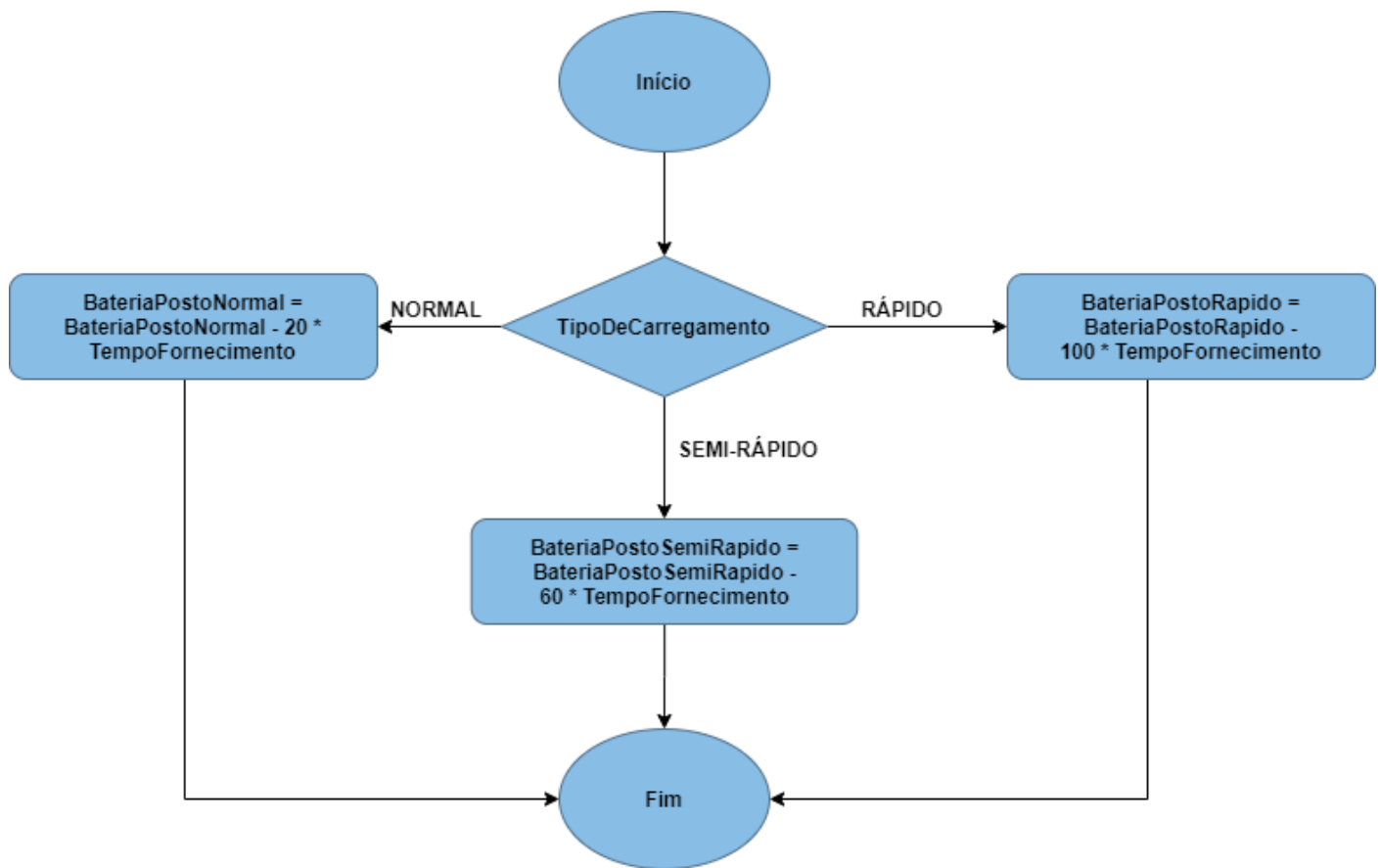
## 7.7. Fluxograma do Fornecimento de Energia



## 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 semirápido
7  CustoRapido           EQU 3       ;custo do carregamento rápido
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-rápido
10 EnderecoBateriaRapido EQU 10E4H   ;endereco onde é guardado o valor da bateria para o carregamento rápido
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)- Avancar "
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 InsereEnergia ;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 NiveisDeEnergia ;rotina para indicar se os postos estão operacionais
409 CALL Verificacao_Cliente ;rotina para efetuar a verificação do cliente
410 CMP R10,-1 ;compara-se o valor em R10 com -1
411 JEQ FimPrograma ;R10 será igual a -1 se o utilizador não foi verificado. Se tal acontecer, salta-se para o fim do programa
412 CALL Carregamento ;rotina para efetuar o carregamento do carro
413
414 FimPrograma:
415 RET
416
417 Desligado:
418 ;*****
419 PUSH R5 ; Guarda na pilha os registos alterados durante esta rotina
420 PUSH R6 ;*****
421 MOV R10,0 ;reset do valor de R10 para 0
422 MOV R6,Ligar ;mete no R6 o endereço do botão que liga a máquina
423 MOV B,R5,[R6] ;mete em R5 o byte endereçado por R6 (o botão de ligar / desligar)
424 CMP R5,0 ;compara R5 com 0
425 JNE FimDesligado ;se R5 é diferente de 0, o posto está ligado e efetua-se este jump; caso contrário, este jump não é efetuado
426 MOV R10,-1 ;mete-se em R10 o valor -1, para indicar que o posto está desligado
427
428 FimDesligado:
429 POP R6 ;*****
430 POP R5 ; Retira da pilha os registos guardados no início da rotina
431 RET ;*****
432
433 Fim:
434 JMP Fim
435
436 ;*****
437 ;
438 ; ROTINA AlteraBaseDeDados
439 ; Responsável por inserir novos clientes ou alterar dados de clientes
440 ; na base de dados
441 ;*****
442
443 AlteraBaseDeDados:
444 PUSH R0 ;*****
445 PUSH R1 ;
446 PUSH R2 ;
447 PUSH R3 ;
448 PUSH R4 ; Guarda na pilha os registos alterados durante esta rotina
449 PUSH R5 ;
450 PUSH R6 ;
451 PUSH R7 ;
452 PUSH R8 ;*****
453
454 AlterarOuNao:
455 ;Início da rotina
456 MOV R2, InputOpcao ;mete-se em R2 o endereço de onde ler a opção que o utilizador escolhe
457 MOV R9, Display_AlteraBaseDeDados ;Mete no registo 9 o endereço do display a mostrar ao utilizador
458 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
459 MOV B,R0,[R2] ;move o byte endereçado por R2, onde está a opção selecionada pelo utilizador, para R0
460 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
461 CMP R0,1 ;compara a opção selecionada pelo utilizador com 1
462 ;se for igual, o utilizador indicou que pretende alterar a base de dados
463 CMP R0,2 ;se R0 não é igual a 1, compara-se a 2
464 JNE OpcaoInvalidaDatabase ;se R0 não é igual a 2, o utilizador inseriu uma opção inválida
465 FIM_AlteraBaseDeDados ;se R0 é igual a 2, o utilizador indicou que não pretende alterar a base de dados e salta-se para o fim desta rotina
466
467 OpcaoInvalidaDatabase:
468 ;instruções relativas ao registo de uma opção inválida inserida pelo utilizador
469 MOV R9, Display_OpcaoInvalida ;Mete no registo 9 o endereço do display a mostrar ao utilizador
470 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
471 JMP AlterarOuNao ;volta ao início da rotina
472
473 CriarOuAlterar:
474 ;a rotina chega aqui se o utilizador indicou que quer fazer alterações à base de dados
475 MOV R9, Display_CriarOuAlterar ;Mete no registo 9 o endereço do display a mostrar ao utilizador
476 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
477 MOV B,R0,[R2] ;move o byte endereçado por R2, onde está a opção selecionada pelo utilizador, para R0
478 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
479 CMP R0,1 ;compara a opção selecionada pelo utilizador com 1
480 ;se R0 não é igual a 1, o utilizador ou pretende alterar os dados de um cliente existente ou inseriu uma opção inválida, por isso salta-se para a tag AlterarDados
481 JNE AlterarDados ;se R0 não é igual a 1, o utilizador ou pretende alterar os dados de um cliente existente ou inseriu uma opção inválida, por isso salta-se para a tag AlterarDados
482 MOV R5, InputID ;mete em R5 o endereço de onde se lê o ID do novo cliente
483 MOV R6, InputCodSeguranca ;mete em R6 o endereço de onde se lê o código de segurança do novo cliente
484 MOV R7, InputSaldo ;mete em R7 o endereço de onde se lê o saldo do novo cliente
485 MOV R4, InputBateria ;mete em R7 o endereço de onde se lê a quantidade de bateria do carro do cliente
486 MOV R9, Display_InserirDadosNovos ;Mete no registo 9 o endereço do display a mostrar ao utilizador
487 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
488 MOV R0,[R5] ;mete em R0 o valor do ID do novo cliente que o utilizador meteu
489 MOV R1,[R6] ;mete em R1 o valor do código de segurança do novo cliente que o utilizador meteu
490 MOV R2,[R7] ;mete em R2 o valor do saldo do novo cliente que o utilizador meteu
491 MOV R3,[R4] ;mete em R3 o valor da bateria do carro do novo cliente que o utilizador meteu
492 CALLF LimpaPerifericosEntrada ;limpa os endereços de onde se lê os inputs do utilizador
493 MOV R8, Base_Tabela_Dados ;mete-se em R8 o início da base de dados
494 MOV R6,0 ;mete-se em R6 o índice
495 MOV R4,Proximo ;mete-se em R4 o valor de Proximo para avançarmos para o próximo cliente a verificar
496 MOV R9,EnderecoTamanho ;mete em R9 o endereço onde encontra-se o tamanho da base de dados
497 MOV R7,[R9] ;R7 é o número de clientes na base de dados
498
499
500 Ciclo_VerificarClienteExiste:
501 ;ciclo em que se percorre a base de dados
502 CMP R6,R7 ;compara-se o índice com o tamanho da base de dados
503 JEQ CriarCliente ;se é igual, chegou-se ao fim da base de dados e efetua-se este salto
504 MOV R5,[R6] ;R5 tem o valor do ID da tabela de base de dados a verificar
505 CMP R0,R5 ;compara-se o ID inserido pelo utilizador com o ID do cliente a ser verificado atualmente
506 JNE ContinuarTabelaVerificacao ;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.
507 MOV R9, Display_ClienteExiste ;Mete no registo 9 o endereço do display a mostrar ao utilizador
508 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
509 JMP AlterarOuNao ;efetua-se o salto para o início da rotina
510
511 ContinuarTabelaVerificacao:
512 ;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
513 ADD R6,1 ;adiciona-se ao índice 1 para indicar que avançamos um cliente
514 ADD R4,R4 ;avança a base para o próximo cliente a verificar
515 JMP Ciclo_VerificarClienteExiste ;faz o salto para o ciclo para verificarmos este próximo cliente
516
517 CriarCliente:
518 MOV R8,Base_Tabela_Dados ;mete-se em R8 o início da base de dados
519 MOV R10,[R9] ;mete-se em R10 o tamanho da base de dados (endereçado por R9)
520 MOV R7,Proximo ;mete-se em R7 o tamanho do salto a ser efetuado para saltar dos dados de um cliente para os dados de outro
521 MUL R10,R7 ;esta multiplicação do tamanho pelo valor Proximo garante que o novo cliente é criado no fim da base de dados
522 ADD R8,R10 ;adiciona-se o valor da multiplicação anterior ao início da base de dados, para gravar-se os dados inseridos no fim da base de dados
523 CMP R2,0 ;compara o valor do saldo com zero
524 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
525 CMP R3,0 ;compara o valor da bateria do carro com zero
526 JLT BateriaInvalida ;se o valor da bateria do carro for negativo, efetua-se este salto e o novo cliente não é criado
527 MOV R7,100 ;mete em R7 o valor 100 para fins de comparação
528 CMP R3,R7 ;compara o valor da bateria do carro com R7, que é 100
529 JGT BateriaInvalida ;se o valor da bateria do carro for maior que 100, efetua-se este salto e o novo cliente não é criado
530 MOV R8,R0 ;mete em R8, que é o fim da base de dados, onde se vai inserir o novo cliente, o ID deste
531 MOV R3,CodSeguranca,R1 ;mete em R3 o código de segurança, o lugar onde está o código de segurança de um cliente, o código de segurança do novo cliente
532 MOV R8,Saldo,R2 ;mete em R8 o Saldo, o lugar onde está o saldo de um cliente, o saldo do novo cliente
533 MOV R8,BateriaCarro,R3 ;mete em R8 o BateriaCarro, o lugar onde está o valor da bateria do carro do cliente, o valor da bateria do carro do novo cliente
534 MOV R1,EnderecoTamanho ;mete em R1 o endereço onde está guardado o tamanho da base de dados para fins de atualização
535 MOV R0,R1 ;mete em R0 o tamanho da base de dados
536 ADD R0,R0 ;incrementa-se o tamanho por 1, pois foi criado um novo cliente
537 MOV R1,R0 ;atualiza-se o tamanho guardado em memória com o novo tamanho, em R0
538 MOV R9, Display_NovoClienteCriado ;Mete no registo 9 o endereço do display a mostrar ao utilizador
539 CALL RefreshDisplay ;Mostra o display metido anteriormente em R9 ao utilizador
540 JMP AlterarOuNao ;Volta ao início do programa

```

```

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 rapida, 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 rapido) 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
710 OpcaoInvalida:
711     MOV R9, Display_OpcaoInvalida        ;Mete no registo 9 o endereço do display a mostrar ao utilizador
712     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
713     CALLF LimpaPerifericosEntrada      ;limpa os endereços de onde se lê os inputs do utilizador
714     JMP InicioInserirEnergia           ;voltar ao início da rotina
715
716 AtualizaPostos:
717     MOV R9, Display_BateriaCarregada    ;Mete no registo 9 o endereço do display a mostrar ao utilizador
718     CALL RefreshDisplay                ;Mostra o display metido anteriormente em R9 ao utilizador
719     JMP InicioInserirEnergia           ;Volta ao início da rotina
720
721 FimInserirEnergia:
722     POP R8                              ;*****
723     POP R7                              ;
724     POP R6                              ; Retira da pilha os registos guardados no início da rotina
725     POP R5                              ;
726     POP R4                              ;
727     POP R3                              ;*****
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
741     MOV R4, Normal                     ;R7 contém o nível mínimo que a bateria normal deve ter para ser considerada funcional (carregamento de uma hora)
742     MOV R5, Semirapido                 ;R8 contém o nível mínimo que a bateria semirapida deve ter para ser considerada funcional (carregamento de uma hora)
743     MOV R6, Rapido                     ;R9 contém o nível mínimo que a bateria rapida deve ter para ser considerada funcional (carregamento de uma hora)
744     CMP R0, R4                         ;compara o valor da bateria do posto normal com o seu valor mínimo
745     JGE VerificaSemiRapido             ;se verificarmos que a bateria normal tem o nível mínimo, verificamos as restantes
746     ADD R3, 1                          ;caso contrário, adicionamos 1 ao contador e verifica-se a bateria semirapida
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_InputVerifyCliente ;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
896 Ciclo_CTN:
897 CMP R8, 0 ;compara o valor do registo 8 com 0
898 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
899 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (20)
900 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
901 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
902 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
903 JMP Ciclo_CTN ;salta para o "tag" Ciclo_CTN
904
905 CalculaTempoSemiRapido:
906 MOV R7, Semirapido ;coloca no registo 7 o valor de energia de um carregamento SemiRapido/hora
907 MOV R8, 100 ;coloca no registo 8 a constante 100
908 SUB R8,R6 ;subtrai a 100 o valor de percentagem da bateria do carro
909
910 Ciclo_CTSR:
911 CMP R8, 0 ;compara o valor do registo 8 com 0
912 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
913 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (60)
914 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
915 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
916 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
917 JMP Ciclo_CTSR ;salta para o "tag" Ciclo_CTSR
918
919 CalculaTempoRapido:
920 MOV R7, Rapido ;coloca no registo 7 o valor de energia de um carregamento Rapido/hora
921 MOV R8, 100 ;coloca no registo 8 a constante 100
922 SUB R8,R6 ;subtrai a 100 o valor de percentagem da bateria do carro
923
924 Ciclo_CTR:
925 CMP R8, 0 ;compara o valor do registo 8 com 0
926 JLE VerificaTempo ;se o valor do registo 8 for inferior ou igual a 0, salta para o tag "VerificaTempo"
927 SUB R8,R7 ;é subtraído a 100 o valor da energia do carregamento (100)
928 SUB R4,1 ;subtrai 1 ao registo 4 (ao tempo)
929 CMP R4,0 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
930 JEQ VerificaTempo ;se o valor do registo 4 for 0, salta para o tag VerificaTempo"
931 JMP Ciclo_CTR ;salta para o "tag" Ciclo_CTR

```



```

932 VerificaTempo:
933     MOV R7,R8
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,R9
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 ;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
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 ;adicionado ao registo 5 o valor do registo 10, ou seja, o índice do cliente
1008 ;é colocado no registo 6, o valor do saldo do cliente
1009 ;é comparado o valor do registo 4 com o valor do registo 6, ou seja, o custo do carregamento com o saldo do utilizador
1010 ;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
1011 ;mete no registo 9, onde está o endereço do que pretendemos mostrar (Display_SaldoInsuficiente)
1012 ;mostra ao utilizador o display metido anteriormente em R9
1013 ;salta para o tag "FimCarregamento", pois não se carregará o carro
1014
1015 ;VERIFICA O TIPO DE CARREGAMENTO A SER FORNECIDO
1016 ;verifica se a opção escolhida é a rápida (R3- opção escolhida, 3- opção rápida)
1017 ;se não for, salta-se para a tag "NaoERapido"
1018 ;se for, comparamos o tempo escolhido (R9) com o tempo de fornecimento (R7)
1019 ;se R9 é maior, saltar para a tag "Excedeu"
1020
1021 ;compara o valor do registo 8 (usado no calculo do tempo) com 0
1022 ;for inferior a 0, saltar para a tag "Excedeu"
1023 ;caso contrário, salta para o tag "NaoExcedeu"
1024
1025 ;SE O TEMPO NÃO CHEGOU A 0 NO FIM DO CARREGAMENTO DA BATERIA
1026 ;mete no registo 9, onde está o endereço do display que pretendemos mostrar (Display_UltrapassaCargaMaxima)
1027 ;mostra ao utilizador o display metido anteriormente em R9
1028
1029 ;SE O TEMPO NÃO CHEGOU A 0 NO FIM DO CARREGAMENTO DA BATERIA (não foi necessário o tempo todo inserido pelo utilizador)
1030 ;coloca no registo 9 o valor do registo 7 (o tempo de fornecimento)
1031 ;coloca no registo 4 o valor do registo 7 (o tempo de fornecimento)
1032 ;é colocado no registo 5 o valor o endereço do início da base de dados
1033 ;é adicionado ao registo 5 o valor do registo 10, ou seja, o índice do cliente
1034 ;é colocado no registo 6, o valor da bateria do veículo do cliente
1035 ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoNormal (1, equivalente à opção)
1036 ;se o valor do registo 3 for igual ao do registo 6, salta para o tag "ForneceEnergiaNormal" - ou seja, o carregamento escolhido é o normal
1037 ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoSemiRapido (2, equivalente à opção)
1038 ;se o valor do registo 3 for igual ao do registo 7, salta para o tag "ForneceEnergiaSemiRapido" - ou seja, o carregamento escolhido é o semi-rápido
1039 ;compara o valor do registo 3 (tipo de carregamento escolhido) com o valor do custoRapido (3, equivalente à opção)
1040 ;se o valor do registo 3 for igual ao do registo 8, salta para o tag "ForneceEnergiaRapido" - ou seja, o carregamento escolhido é o rápido
1041
1042 ;mete no registo 9, onde está o endereço do que pretendemos mostrar (Display_SaldoInsuficiente)
1043 ;mostra ao utilizador o display metido anteriormente em R9
1044
1045 ;FORNECE ENERGIA DO TIPO NORMAL
1046 ;coloca no registo 7 o valor de energia de um carregamento Normal/hora
1047 ;coloca no registo 8 a constante 100
1048
1049 ;é adicionado à bateria do veículo, o valor da energia do carregamento (20)
1050 ;atualiza o valor da bateria do veículo do cliente
1051 ;subtrai 1 ao registo 4 (ao tempo)
1052 ;compara o valor do registo 6 com o valor do registo 8, ou seja, a bateria do veículo com a constante 100
1053 ;se o valor do registo 6 for superior ou igual a 100, salta para o tag "BateriaCarregada"
1054 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
1055 ;se o valor do registo 4 for 0, salta para o tag AtualizaValoresEnergia"
1056 ;salta para o "tag" Ciclo_FEN
1057
1058 ;FORNECE ENERGIA DO TIPO SEMIRAPIDO
1059 ;coloca no registo 7 o valor de energia de um carregamento Semi-Rápido/hora
1060 ;coloca no registo 8 a constante 100
1061
1062 ;é adicionado à bateria do veículo, o valor da energia do carregamento (20)
1063 ;atualiza o valor da bateria do veículo do cliente
1064 ;subtrai 1 ao registo 4 (ao tempo)
1065 ;compara o valor do registo 6 com o valor do registo 8, ou seja, a bateria do veículo com a constante 100
1066 ;se o valor do registo 6 for superior ou igual a 100, salta para o tag "BateriaCarregada"
1067 ;compara o valor do registo 4 com a constante 0, ou seja, se o tempo chegou a 0
1068 ;se o valor do registo 4 for 0, salta para o tag AtualizaValoresEnergia"
1069 ;salta para o "tag" Ciclo_FESR
1070
1071 ;FORNECE ENERGIA DO TIPO RAPIDO
1072 ;subtrai 1 ao registo 4 (ao tempo)
1073 ;Como o carregamento é rápido e este carrega a bateria toda numa hora, saltamos logo para a tag "BateriaCarregada"
1074
1075 ;QUANDO A BATERIA DO VEICULO ULTRAPASSA OS 100%
1076 ;coloca no registo 6 a constante 100
1077 ;atualiza o valor da bateria do veículo do cliente (coloca a 100)
1078
1079 ;VERIFICA SE O TEMPO CHEGOU A 0 NO FIM DO CARREGAMENTO DA BATERIA
1080 ;coloca no registo 7 o valor do registo 9 (o tempo escolhido pelo utilizador)
1081 ;move-se o registo 7 (tempo que demorou para carregar o carro) para o registo 8 para escrever o valor no display
1082 ;mete no registo 9, onde está o endereço do display que pretendemos mostrar
1083 ;mostra ao utilizador o display metido anteriormente em R9
1084 ;mete no registo 9 o número de bytes a adicionar ao início do display para escrever o valor no lugar certo
1085 ;chama a função que escreve o valor pretendido (que se encontra no registo 8)
1086 ;mete no registo 6 o início do display
1087 ;mete no registo 8 o valor a acrescentar ao início do display
1088 ;adiciona ao registo 6 o registo 8, para escrevermos o tipo de carregamento no lugar certo
1089 ;compara o tipo de carregamento escolhido com CustoNormal (1, igual à opção de escolher normal)
1090 ;se não for igual, o utilizador não escolheu o carregamento normal e verifica-se o próximo tipo de carregamento
1091 ;se foi igual, escrevemos o tipo de carregamento (Normal) no display
1092 ;efetua-se este salto para não verificar os outros tipos de carregamento

```

```

1064 VerSemi:
1065     CMP R3,CustoSemiRápido           ;compara o tipo de carregamento escolhido com CustoSemiRápido (2, igual à opção)
1066     JNE VerRápido                   ;se não for igual salta-se para escrever rápido
1067     CALLF EscreveSemiRápido         ;se for igual, escreve-se o tipo de carregamento (Semi-Rápido) no display
1068     JMP CarregamentoEscrito        ;efetua-se este salto para não verificar o último tipo de carregamento
1069
1070 VerRápido:
1071     CALLF EscreveRápido             ;não é efetuadas comparações pois só chega aqui se o utilizador não escolheu os outros dois tipos de carregamento
1072
1073 CarregamentoEscrito:
1074     CALLF VerificaOK                ;verifica-se que o utilizador pretende continuar
1075     MOV R4, R7                      ;é colocado no registo 4 o valor do tempo que foi necessario para o carregamento da bateria
1076     MUL R4,R3                      ;é multiplicado o valor do registo 7 com o valor do registo 3, ou seja, o valor do tempo necessario com o valor do custo do tipo de carregamento escolhido
1077     CALL Debito                    ;é feito o pagamento do carregamento
1078     CMP R3, CustoNormal             ;compara o valor do registo 3 com o valor do custoNormal (1, equivalente à opção)
1079     JEQ AtualizaPostoNormal         ;se o valor do registo 3 for igual ao do registo 6, salta para o tag "AtualizaPostoNormal" -> o carregamento foi Normal
1080     CMP R3, CustoSemiRápido         ;compara o valor do registo 3 com o valor do custoSemiRápido (2, equivalente à opção)
1081     JEQ AtualizaPostoSemiRápido     ;se o valor do registo 3 for igual ao do registo 7, salta para o tag "AtualizaPostoSemiRápido" -> o carregamento foi Semi-Rápido
1082     CMP R3, CustoRápido             ;compara o valor do registo 3 com o valor do custoRápido (3, equivalente à opção)
1083     JEQ AtualizaPostoRápido         ;se o valor do registo 3 for igual ao do registo 7, salta para o tag "AtualizaPostoRápido" -> o carregamento foi Rápido
1084
1085 Debito:
1086     MOV R6, [R5+Saldo]              ;REALIZA O PAGAMENTO DO CARREGAMENTO
1087     SUB R6, R7                      ;coloca no registo 6 o valor do saldo do cliente
1088     MOV [R5+Saldo],R6              ;subtrai ao registo 6 o valor do registo 7 (o custo do carregamento)
1089     CALL RefreshDisplay             ;atualiza o saldo do utilizador
1090     MOV R9,R7                      ;mete no registo 9, onde está o endereço do display que pretendemos mostrar
1091     CALL RefreshDisplay             ;mostra ao utilizador o display metido anteriormente em R9
1092     MOV R8,R6                      ;mete-se em R8 o saldo do utilizador (R6), para mostrar no display
1093     MOV R9,57                      ;mete-se em R9 o número de bytes a saltar desde o início do display para escrever R8 no lugar certo
1094     CALLF EscreveValores           ;escreve o valor de R8 no display
1095     MOV R8,R7                      ;mete-se em R8 o custo do carregamento (R7) para escrever no display
1096     MOV R9,59                      ;mete-se em R9 o número de bytes a saltar desde o início do display para escrever R8 no lugar certo
1097     CALLF EscreveValores           ;escreve o valor de R8 no display
1098     CALLF VerificaOK                ;verifica se o utilizador quer continuar
1099     RET
1100
1101 AtualizaPostoNormal:
1102     MOV R6, Normal                  ;ATUALIZA O VALOR DA BATERIA DO POSTO NORMAL
1103     MUL R6, R4                      ;coloca no registo 6 o valor da energia de um carregamento Normal/hora (20)
1104     SUB R0, R6                      ;é multiplicado o registo 6 (a energia do carregamento) com valor do registo 4 (o valor do tempo que foi necessario para o carregamento da bateria)
1105     JMP CarregamentoConcluido     ;subtrai ao registo 0 (bateria Normal do posto) o valor do registo 6 (o valor da energia fornecida no carregamento)
1106                                     ;salta para o tag "CarregamentoConcluido"
1107
1108 AtualizaPostoSemiRápido:
1109     MOV R6, Semirápido              ;ATUALIZA O VALOR DA BATERIA DO POSTO SEMI-RAPIDO
1110     MUL R6, R4                      ;coloca no registo 6 o valor da energia de um carregamento Semi-Rápido/hora (60)
1111     SUB R1, R6                      ;é multiplicado o registo 6 (a energia do carregamento) com valor do registo 4 (o valor do tempo que foi necessario para o carregamento da bateria)
1112     JMP CarregamentoConcluido     ;subtrai ao registo 1 (bateria Semi-Rápida do posto) o valor do registo 6 (o valor da energia fornecida no carregamento)
1113                                     ;salta para o tag "CarregamentoConcluido"
1114
1115 AtualizaPostoRápido:
1116     MOV R6, Rápido                 ;ATUALIZA O VALOR DA BATERIA DO POSTO RAPIDO
1117     MUL R6, R4                      ;coloca no registo 6 o valor da energia de um carregamento Rápido/hora (100)
1118     SUB R2, R6                      ;é multiplicado o registo 6 (a energia do carregamento) com valor do registo 4 (o valor do tempo que foi necessario para o carregamento da bateria)
1119     JMP CarregamentoConcluido     ;subtrai ao registo 2 (bateria Rápida do posto) o valor do registo 6 (o valor da energia fornecida no carregamento)
1120
1121 CarregamentoConcluido:
1122     MOV R9, Display_CarregamentoConcluido ;ATUALIZA VALORES DAS BATERIAS DO POSTO (CARREGAMENTO CONCLUIDO)
1123     CALL RefreshDisplay             ;mete no registo 9, onde está o endereço do display que pretendemos mostrar
1124                                     ;mostra ao utilizador o display metido anteriormente em R9
1125
1126 FimCarregamento:
1127     POP R8                          ;
1128     POP R7                          ;
1129     POP R6                          ;
1130     POP R5                          ; Retira da pilha os registos guardados no início da rotina
1131     POP R4                          ;
1132     POP R3                          ;
1133     RET                             ;
1134
1135 ;*****
1136 ;                               ROTINA VerificaOK
1137 ;                               Responsável por registar que o utilizador "carregou" no OK
1138 ;*****
1139 VerificaOK:
1140     PUSH R0                         ;*****
1141     PUSH R1                         ; Guarda na pilha os registos alterados durante esta rotina
1142
1143 CicloVerOK:
1144     MOV R0, OK                     ;mete em R0 o endereço de onde ver se o utilizador "carregou" OK
1145     MOV R1, [R0]                   ;mete em R1 o valor lido do endereço R0
1146     CMP R1,1                       ;compara R1 com 1
1147     JNE CicloVerOK                ;só quando o utilizador mudar o valor para 1 é que se avança o programa
1148     MOV R1,0                       ;mete-se 0 em R1 para fazer reset do botão OK
1149     MOV [R0],R1                   ;reset do botão OK
1150     POP R1                         ;*****
1151     POP R0                         ; Retira da pilha os registos guardados no início da rotina
1152     RET                             ;*****
1153
1154 ;*****
1155 ;                               ROTINA RefreshDisplay
1156 ;                               Responsável por atualizar o display que o utilizador vê consoante o
1157 ;                               display a mostrar, indicado pelo registo R9
1158 ;*****
1159 RefreshDisplay:
1160     PUSH R0                         ;*****
1161     PUSH R1                         ;
1162     PUSH R2                         ; Guarda na pilha os registos alterados durante esta rotina
1163     PUSH R3                         ;
1164     PUSH R4                         ;
1165     MOV R4,R9                       ;guarda-se em R4 o endereço do display a mostrar (R9)
1166     MOV R0,InicioDisplay           ;mete em R0 o início do display
1167     MOV R1,FimDisplay              ;mete em R1 o fim do display
1168
1169 Ciclo_RefreshDisplay:
1170     MOV R2,[R0]                   ;mete-se em R2 o conteúdo do display que queremos apresentar, indicado por R9
1171     MOV [R0],R2                   ;mete-se no display mostrado ao utilizador o conteúdo de R2
1172     ADD R0,2                       ;avança-se para a próxima palavra o display mostrado ao utilizador
1173     ADD R9,2                       ;avança-se para a próxima palavra o display que pretendemos mostrar ao utilizador
1174     CMP R0,R1                      ;compara-se R0, posição atual no display, com o fim do display (R1)
1175     JLE Ciclo_RefreshDisplay      ;se ainda não chegámos ao fim (é menor ou igual) volta-se ao início do ciclo
1176     MOV R3, Display_NiveisDeEnergia ;mete-se em R3 o endereço do display dos níveis de energia
1177     MOV R3,R4                     ;por isso, se R9 é igual a R3, efetua-se este salto para não se chamar o VerificaOK pois falta escrever no display os estados dos postos
1178     MOV R3, Display_Debito         ;mesmo raciocínio aqui, R3 agora é o endereço do display do débito
1179     CMP R3,R4                     ;compara-se o valor de R4 (endereço do display que foi mostrado) com R3 (endereço do display de débito)
1180     JEQ FimRefreshDisplay          ;se é igual, este salto significa que não se chama a rotina VerificaOK pois ainda há informação a ser escrita no display
1181     MOV R3, Display_InfoCarregamento ;mesmo raciocínio aqui, R3 agora é o endereço do display da informação do carregamento
1182     CMP R3,R4                     ;compara-se o valor de R4 (endereço do display que foi mostrado) com R3 (endereço do display da informação do carregamento)
1183     JEQ FimRefreshDisplay          ;se é igual, efetua-se este salto para não chamar a rotina VerificaOK pois ainda há informação a ser mostrada no display
1184     CALLF VerificaOK               ;registra se o utilizador quer proceder
1185
1186 FimRefreshDisplay:
1187     POP R4                         ;*****
1188     POP R3                         ;
1189     POP R2                         ;
1190     POP R1                         ; Retira da pilha os registos guardados no início da rotina
1191     RET                             ;*****

```

```

1192 ;*****
1193 ;          ROTINA LimpaPerifericosEntrada
1194 ;          Esta rotina, como o nome indica, "limpa" os endereços de onde se lê
1195 ;          os dados que o utilizador insira
1196 ;*****
1197 LimpaPerifericosEntrada:
1198     PUSH R0
1199     PUSH R1
1200     PUSH R2
1201     PUSH R3
1202     PUSH R4
1203     PUSH R5
1204     PUSH R6
1205     PUSH R7
1206     MOV R0, InputCodSeguranca
1207     MOV R1, InputID
1208     MOV R2, InputIncrementoBateria
1209     MOV R3, InputTempo
1210     MOV R4, InputOpcao
1211     MOV R5, InputSaldo
1212     MOV R6, InputBateria
1213     MOV R7, 0
1214     MOV [R0],R7
1215     MOV [R1],R7
1216     MOV [R2],R7
1217     MOV [R3],R7
1218     MOV [R4],R7
1219     MOV [R5],R7
1220     MOV [R6],R7
1221     POP R7
1222     POP R6
1223     POP R5
1224     POP R4
1225     POP R3
1226     POP R2
1227     POP R1
1228     POP R0
1229     RETF
1230
1231 ;*****
1232 ;          ROTINA LimpaDisplay
1233 ;          Como o nome indica, "limpa" o display mostrado ao utilizador,
1234 ;          preenchendo-o com espaços
1235 ;*****
1236 LimpaDisplay:
1237     PUSH R0
1238     PUSH R1
1239     PUSH R2
1240     MOV R0,InicioDisplay
1241     MOV R1,FimDisplay
1242     MOV R2,32
1243
1244     Ciclo_LimpaDisplay:
1245     MOVB [R0],R2
1246     ADD R0,1
1247     CMP R0,R1
1248     JLE Ciclo_LimpaDisplay
1249     POP R2
1250     POP R1
1251     POP R0
1252     RETF
1253
1254 ;*****
1255 ;          ROTINA Display_NiveisDeEnergia_InserirInformacao
1256 ;          Responsável por verificar os valores das baterias dos postos e escrever
1257 ;          no display mostrado ao utilizador o estado dos postos conforme os valores
1258 ;          das baterias (Funcional indica que a bateria está acima do valor mínimo de um carregamento de uma hora)
1259 ;*****
1260 Display_NiveisDeEnergia_InserirInformacao:
1261     PUSH R3
1262     PUSH R4
1263     PUSH R5
1264     PUSH R6
1265     MOV R5, InicioDisplay
1266     MOV R4,Normal
1267     MOV R6,23
1268     ADD R6,R5
1269     CMP R0,R4
1270     JLT NaoFuncionalNormal
1271     CALLF EscreveFuncional
1272     JMP InfoSemiRapido
1273
1274     NaoFuncionalNormal:
1275     CALLF EscreveNao_Func
1276
1277     InfoSemiRapido:
1278     MOV R4,Semirapido
1279     CMP R1,R4
1280     JLT NaoFuncionalSemiRapido
1281     MOV R6,48
1282     ADD R6,R5
1283     CALLF EscreveFuncional
1284     JMP InfoRapido
1285
1286     NaoFuncionalSemiRapido:
1287     MOV R6,43
1288     ADD R6,R5
1289     MOV R3,2
1290     CALLF EscreveNao
1291     CALLF EscreveFuncional
1292
1293     InfoRapido:
1294     MOV R4,Rapido
1295     CMP R2,R4
1296     JLT NaoFuncionalRapido
1297     MOV R6,80
1298     ADD R6,R5
1299     CALLF EscreveFuncional
1300     JMP FimInformacoesPosto
1301

```

```

1302 NaoFuncionalRapido:
1303     MOV R6,73 ;mete-se em R6 o avanço relativo ao início do display para escrever a informação no lugar certo
1304     ADD R6,R5 ;acrescenta-se a R6 o início do display
1305     MOV R3,4 ;mete-se em R3 o número de espaços a dar depois de escrever "Nao" para avançar para a próxima linha
1306     CALLF EscreveNao ;como não está funcional, escreve-se "Nao Funcional"
1307     CALLF EscreveFuncional ;como não está funcional, escreve-se "Nao Funcional"
1308
1309 FimInformacoesPosto:
1310     CALLF VerificaOK ;verifica que o utilizador quer proceder
1311     POP R6 ;
1312     POP R5 ;
1313     POP R4 ; Retira da pilha os registos guardados no início da rotina
1314     POP R3 ;
1315     RET ;
1316
1317 ;*****
1318 ;             ROTINA EscreveNao
1319 ;             Como o nome indica, esta rotina é responsável por escrever
1320 ;             a palavra "Nao" através do código ASCII e avançar para a
1321 ;             próxima linha, através de espaços
1322 ;             R4 - carater a mostrar no display,
1323 ;             R6 - endereço onde escrever o carater (R4)
1324 ;*****
1325 EscreveNao:
1326     MOV R4,78 ;código ASCII de "N"
1327     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1328     ADD R6,1 ;avança para o próximo byte
1329     MOV R4,97 ;código ASCII de "a"
1330     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1331     ADD R6,1 ;avança para o próximo byte
1332     MOV R4,111 ;código ASCII de "o"
1333     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1334     ADD R6,1 ;avança para o próximo byte
1335
1336 CicloEspacos:
1337     MOV R4,32 ;ciclo para escrever o número de espaços necessários para avançar para a próxima linha
1338     MOV [R6],R4 ;código ASCII do espaço
1339     ADD R6,1 ;escreve-se no display o carater dado por R4
1340     SUB R3,1 ;avança para o próximo byte
1341     CMP R3,0 ;subtrai 1 a R3 (o contador de espaços)
1342     JNE CicloEspacos ;verifica-se se R3 já chegou a 0
1343     RETF ;se ainda não chegou, volta-se ao início do ciclo para escrever mais um espaço
1344 ;se chegou, a rotina terminou
1345
1346 ;*****
1347 ;             ROTINA EscreveFuncional
1348 ;             Semelhante à anterior, só que escreve "Funcional" em vez de "Nao"
1349 ;             e não faz parágrafo, ao contrário da anterior
1350 ;             R4 - carater a mostrar no display,
1351 ;             R6 - endereço onde escrever o carater (R4)
1352 ;*****
1353 EscreveFuncional:
1354     MOV R4,70 ;código ASCII de "F"
1355     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1356     ADD R6,1 ;avança para o próximo byte
1357     MOV R4,117 ;código ASCII de "u"
1358     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1359     ADD R6,1 ;avança para o próximo byte
1360     MOV R4,110 ;código ASCII de "n"
1361     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1362     ADD R6,1 ;avança para o próximo byte
1363     MOV R4,99 ;código ASCII de "c"
1364     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1365     ADD R6,1 ;avança para o próximo byte
1366     MOV R4,105 ;código ASCII de "i"
1367     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1368     ADD R6,1 ;avança para o próximo byte
1369     MOV R4,111 ;código ASCII de "o"
1370     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1371     ADD R6,1 ;avança para o próximo byte
1372     MOV R4,110 ;código ASCII de "n"
1373     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1374     ADD R6,1 ;avança para o próximo byte
1375     MOV R4,97 ;código ASCII de "a"
1376     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1377     ADD R6,1 ;avança para o próximo byte
1378     MOV R4,108 ;código ASCII de "l"
1379     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1380     RETF
1381
1382 ;*****
1383 ;             ROTINA EscreveNao_Func
1384 ;             Semelhante às anteriores, só que esta escreve "Nao Func.".
1385 ;             Só é utilizada para demonstrar que o posto normal não está operacional
1386 ;             R4 - carater a mostrar no display,
1387 ;             R6 - endereço onde escrever o carater (R4)
1388 ;*****
1389 EscreveNao_Func:
1390     MOV R4,78 ;código ASCII de "N"
1391     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1392     ADD R6,1 ;avança para o próximo byte
1393     MOV R4,97 ;código ASCII de "a"
1394     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1395     ADD R6,1 ;avança para o próximo byte
1396     MOV R4,111 ;código ASCII de "o"
1397     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1398     ADD R6,1 ;avança para o próximo byte
1399     MOV R4,32 ;código ASCII do espaço
1400     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1401     ADD R6,1 ;avança para o próximo byte
1402     MOV R4,70 ;código ASCII de "F"
1403     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1404     ADD R6,1 ;avança para o próximo byte
1405     MOV R4,117 ;código ASCII de "u"
1406     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1407     ADD R6,1 ;avança para o próximo byte
1408     MOV R4,110 ;código ASCII de "n"
1409     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1410     ADD R6,1 ;avança para o próximo byte
1411     MOV R4,99 ;código ASCII de "c"
1412     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1413     ADD R6,1 ;avança para o próximo byte
1414     MOV R4,46 ;código ASCII do ponto "."
1415     MOV [R6],R4 ;escreve-se no display o carater dado por R4
1416     RETF

```



```

1418 ;*****
1419 ;
1420 ;          ROTINA EscreveNormal
1421 ;          Semelhante às anteriores, só que esta escreve "Normal".
1422 ;          Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso Normal
1423 ;          R4 - carater a mostrar no display,
1424 ;          R6 - endereço onde escrever o carater (R4)
1425 ;*****
1426 EscreveNormal:
1427     MOV R4,78 ;código ASCII de "N"
1428     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1429     ADD R6,1 ;avança para o próximo byte
1430     MOV R4,111 ;código ASCII de "o"
1431     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1432     ADD R6,1 ;avança para o próximo byte
1433     MOV R4,114 ;código ASCII de "r"
1434     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1435     ADD R6,1 ;avança para o próximo byte
1436     MOV R4,109 ;código ASCII de "m"
1437     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1438     ADD R6,1 ;avança para o próximo byte
1439     MOV R4,97 ;código ASCII de "a"
1440     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1441     ADD R6,1 ;avança para o próximo byte
1442     MOV R4,108 ;código ASCII de "l"
1443     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1444     RETF

1446 ;*****
1447 ;
1448 ;          ROTINA EscreveSemiRapido
1449 ;          Semelhante às anteriores, só que esta escreve "Semi-Rapido".
1450 ;          Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso SemiRapido
1451 ;          R4 - carater a mostrar no display,
1452 ;          R6 - endereço onde escrever o carater (R4)
1453 ;*****
1454 EscreveSemiRapido:
1455     MOV R4,83 ;código ASCII de "S"
1456     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1457     ADD R6,1 ;avança para o próximo byte
1458     MOV R4,101 ;código ASCII de "e"
1459     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1460     ADD R6,1 ;avança para o próximo byte
1461     MOV R4,109 ;código ASCII de "m"
1462     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1463     ADD R6,1 ;avança para o próximo byte
1464     MOV R4,105 ;código ASCII de "i"
1465     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1466     ADD R6,1 ;avança para o próximo byte
1467     MOV R4,45 ;código ASCII de "-", o hifen
1468     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1469     ADD R6,1 ;avança para o próximo byte
1470     MOV R4,82 ;código ASCII de "R"
1471     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1472     ADD R6,1 ;avança para o próximo byte
1473     MOV R4,97 ;código ASCII de "a"
1474     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1475     ADD R6,1 ;avança para o próximo byte
1476     MOV R4,112 ;código ASCII de "p"
1477     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1478     ADD R6,1 ;avança para o próximo byte
1479     MOV R4,105 ;código ASCII de "i"
1480     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1481     ADD R6,1 ;avança para o próximo byte
1482     MOV R4,100 ;código ASCII de "d"
1483     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1484     ADD R6,1 ;avança para o próximo byte
1485     MOV R4,111 ;código ASCII de "o"
1486     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1487     RETF

1489 ;*****
1490 ;
1491 ;          ROTINA EscreveRapido
1492 ;          Semelhante às anteriores, só que esta escreve "Rapido".
1493 ;          Só é utilizada para demonstrar o tipo de carregamento escolhido pelo utilizador, neste caso Rapido
1494 ;          R4 - carater a mostrar no display,
1495 ;          R6 - endereço onde escrever o carater (R4)
1496 ;*****
1497 EscreveRapido:
1498     MOV R4,82 ;código ASCII de "R"
1499     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1500     ADD R6,1 ;avança para o próximo byte
1501     MOV R4,97 ;código ASCII de "a"
1502     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1503     ADD R6,1 ;avança para o próximo byte
1504     MOV R4,112 ;código ASCII de "p"
1505     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1506     ADD R6,1 ;avança para o próximo byte
1507     MOV R4,105 ;código ASCII de "i"
1508     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1509     ADD R6,1 ;avança para o próximo byte
1510     MOV R4,100 ;código ASCII de "d"
1511     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1512     ADD R6,1 ;avança para o próximo byte
1513     MOV R4,111 ;código ASCII de "o"
1514     MOVB [R6],R4 ;escreve-se no display o carater dado por R4
1515     RETF

```

```

1516
1517 ;*****
1518 ;
1519 ;           ROTINA EscreveValores
1520 ; Rotina responsável por escrever valores numéricos no display.
1521 ; R8 - valor numerico a mostrar no display
1522 ; R9 - número de bytes a avançar para escrever o valor no lugar certo
1523 ;*****
1524 EscreveValores:
1525     PUSH R0
1526     PUSH R1
1527     PUSH R2
1528     MOV R0,InicioDisplay
1529     ADD R0,R9
1530     MOV R1,10
1531     MOV R2,48
1532
1533 CicloEscreveValores:
1534     MOV R3,R8
1535     MOD R3,R1
1536     ADD R3,R2
1537     MOVB [R0],R3
1538     SUB R0,1
1539     DIV R8,R1
1540     CMP R8,0
1541     JEQ FimEscreveValores
1542     JMP CicloEscreveValores
1543
1544 FimEscreveValores:
1545     POP R3
1546     POP R2
1547     POP R1
1548     POP R0
1549     RETF

```