

FUNÇÕES

desmontador

int desmontador (int argc, char *argv[]);

Desmonta um arquivo em linguagem de máquina (.txt) em um arquivo em linguagem assembly 8085 (.txt).

Parâmetros

argc:	Contador do número de comandos da linha escrita no terminal.
argv:	Vetor de strings com os valores escritos na linha de comando do terminal.

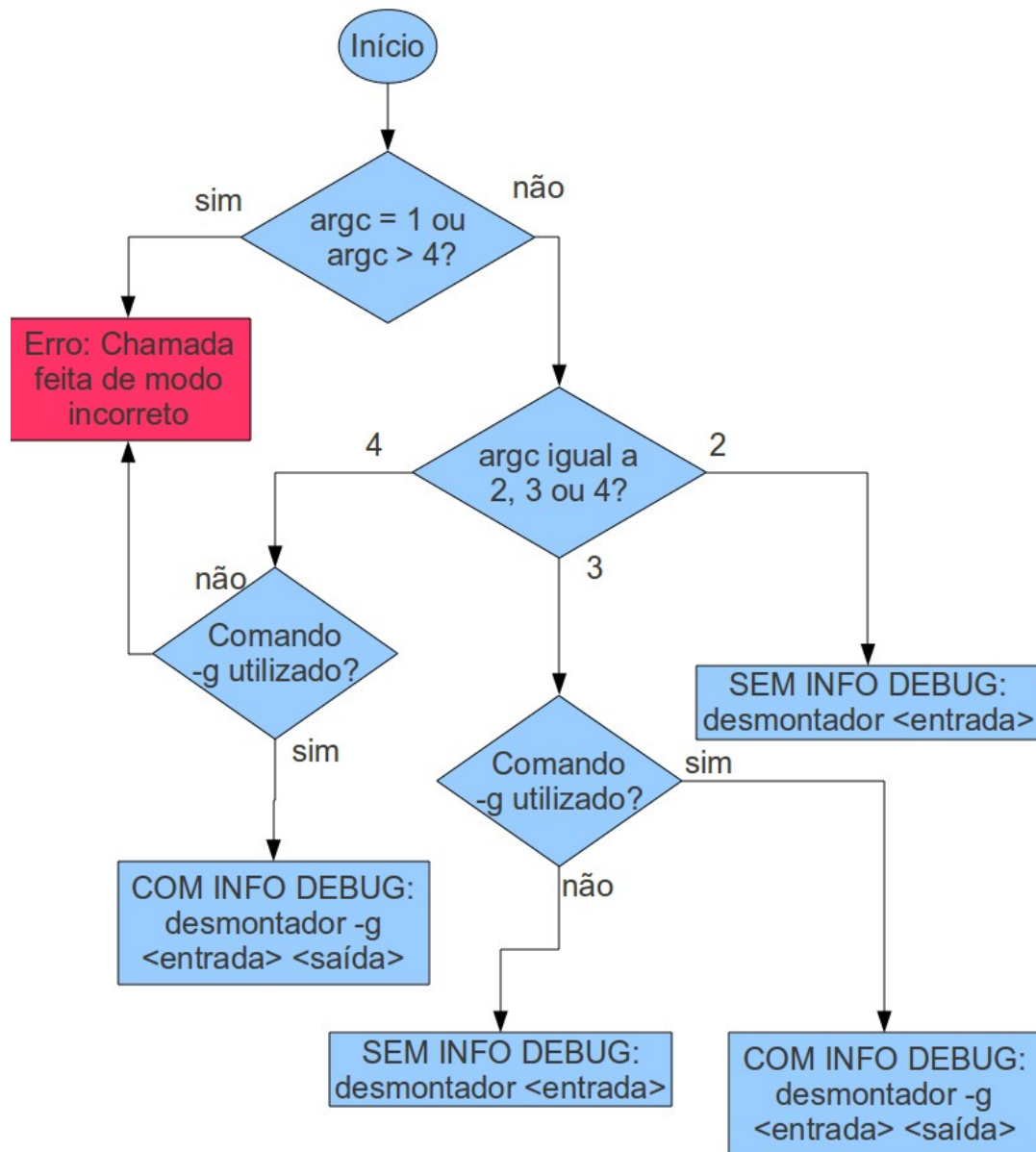
Valor de retorno

Retorna 0.

Bibliotecas requeridas

- stdio.h
- stdlib.h
- string.h
- desmontador.h

Fluxograma



desmontar

```
void desmontar(char *linguagemmaquina, char *linguagemassembly);
```

Chama todas as funções necessárias para desmontar o arquivo pelo Program Header.

Parâmetros

linguagemmaquina: Nome do arquivo aonde se encontra a linguagem de máquina a ser desmontada.

linguagemassembly: Nome do arquivo alvo aonde se desmontará o arquivo..

Valor de retorno

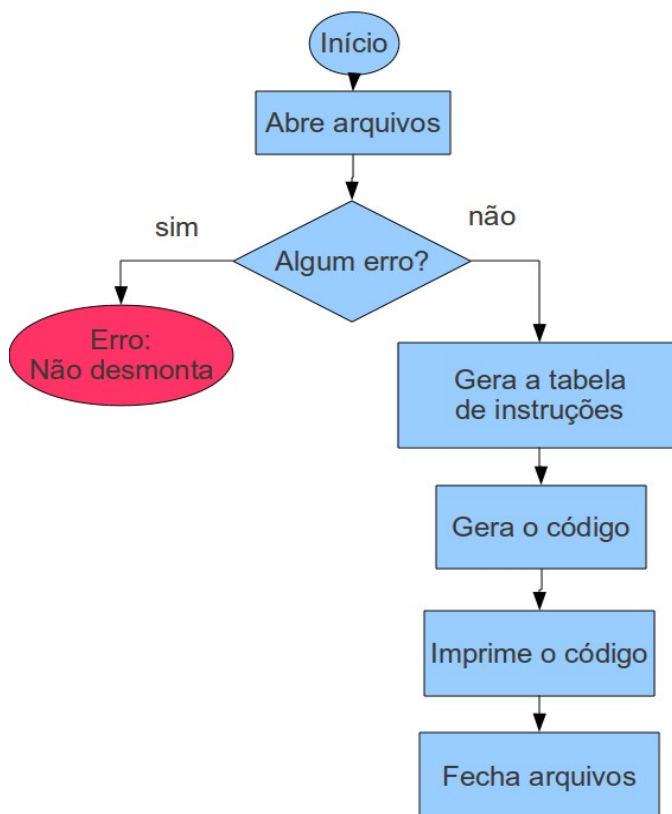
Void.

Bibliotecas requeridas

stdio.h

desmontador.h

Fluxograma



geratabela

`void geratabela(tabelaS **p_inicio, FILE *tabela);`

Gera a estrutura de dados com os dados da tabela a partir do arquivo “tabela.txt”.

Parâmetros

`p_inicio`: início da posição onde ficará o primeiro elemento da tabela.
`tabela`: nome do arquivo fonte onde se encontra todas as informações para a criação da tabela.

Valor de retorno

Void.

Bibliotecas requeridas

`stdio.h`
`stdlib.h`
`desmontador.h`

Estrutura de dados

```
typedef struct elementotabela{  
    int opcode[2];  
    int tipo;  
    char nome[8];  
    struct elementotabela *next;  
} tabelaS;
```

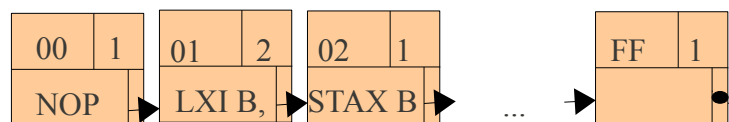
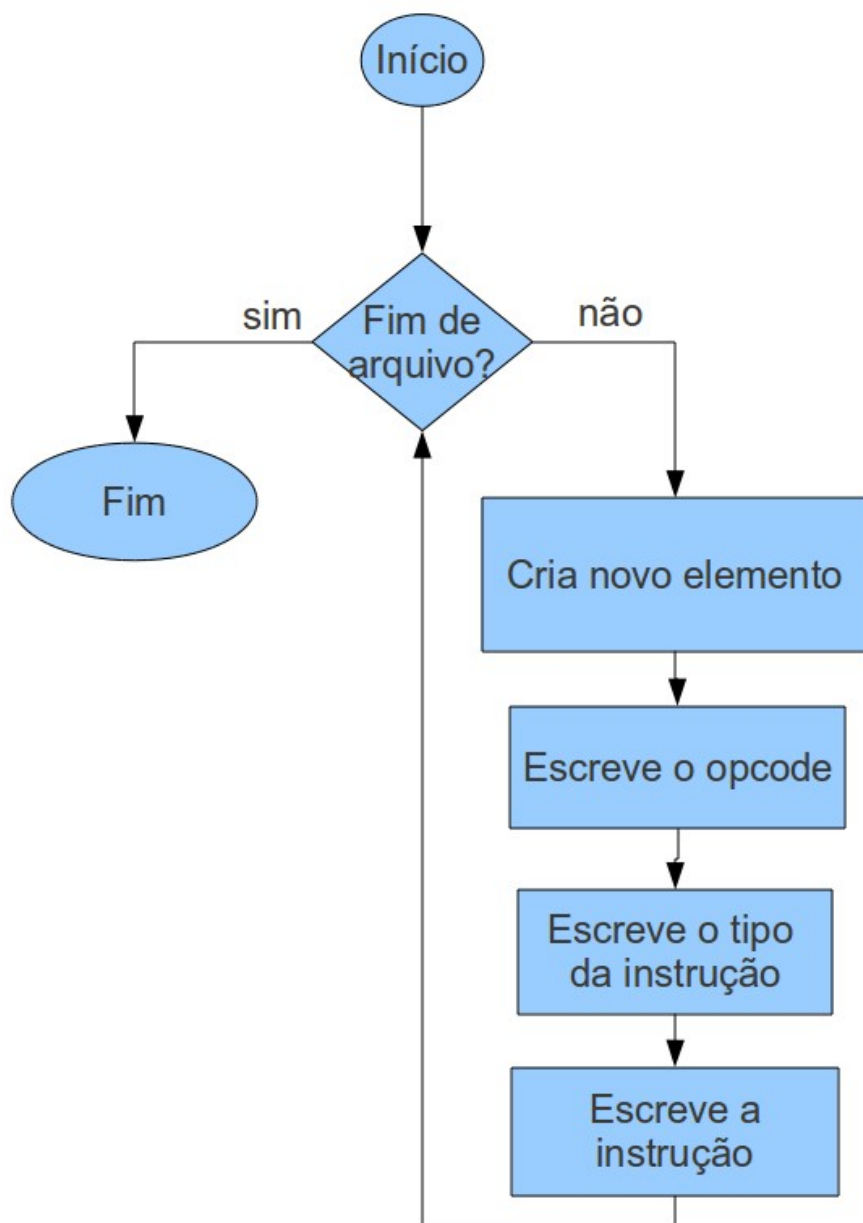


tabela.txt

Arquivo texto com os opcodes, os tipos e os nomes de todas as instruções da linguagem assembly 8085, como abaixo:

```
00 1 NOP  
01 3 LXI B,  
02 1 STAX B  
(...)  
FF 1 RST 7
```

Fluxograma



geracodigo

```
labelS *geracodigo(FILE *codigo, undadosS **p_iniciadosun, codeS  
**p_newcode, tabelaS *p_tabela, dataS **p_newdata);
```

Lê o cabeçalho, lê o program header, lê as instruções (gera labels aleatórias), lê os dados (.data e .bss) e com tudo isso gera as estruturas de dados com o código, os dados e as labels.

Parâmetros

codigo:	Nome do arquivo fonte que será desmontado.
p_iniciadosun:	Posição do início dos dados não inicializados.
p_newcode:	Posição do início da struct de código.
p_tabela:	Início da struct da tabela.
p_newdata:	Posição do início dos dados inicializados.

Valor de retorno

labelS *:	Ponteiro do inicio da posição onde ficará a primeira label criada.
-----------	--

Bibliotecas requeridas

stdio.h
stdlib.h
string.h
desmontador.h

Estrutura de dados

```
typedef struct code{  
    int endereco;  
    int *codigocompleto;  
} tabelaS;
```

0000
21
11
00
21

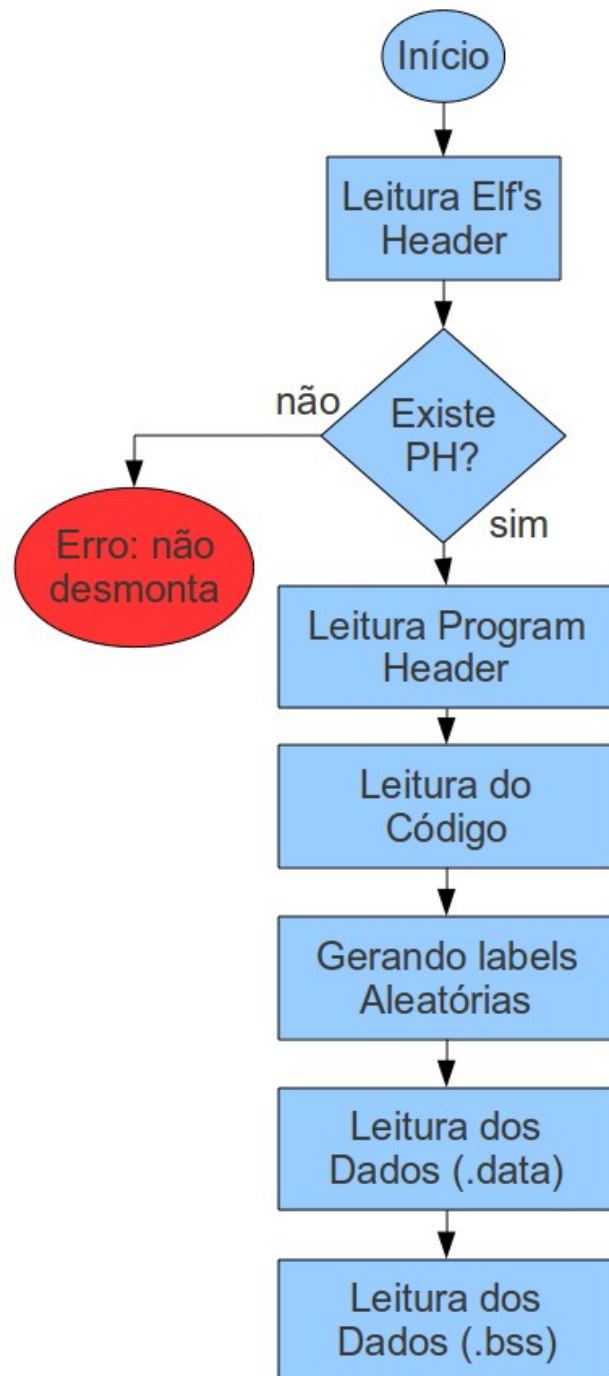
```
typedef struct data{  
    int endereco;  
    int *datacompleto;  
} dataS;
```

0011
00
01

```
typedef struct undados{  
    int endereco;  
    int quantidade;  
} undadosS;
```

0013
02

Fluxograma



gerarlabel

char *gerarlabel(labelS **p_label, int endereco, char *nome);

Gera a estrutura de dados com os endereços e as labels utilizadas no código e retorna a label do endereço requerido nos parâmetros.

Parâmetros

p_label: Início da posição onde ficará o primeira label.
endereco: O endereço da label que se deseja criar.
nome: O nome da label que se deseja criar, se seu valor for zero, a função cria uma label.

Valor de retorno

char *
String da label indicada pelo endereço.

Bibliotecas requeridas

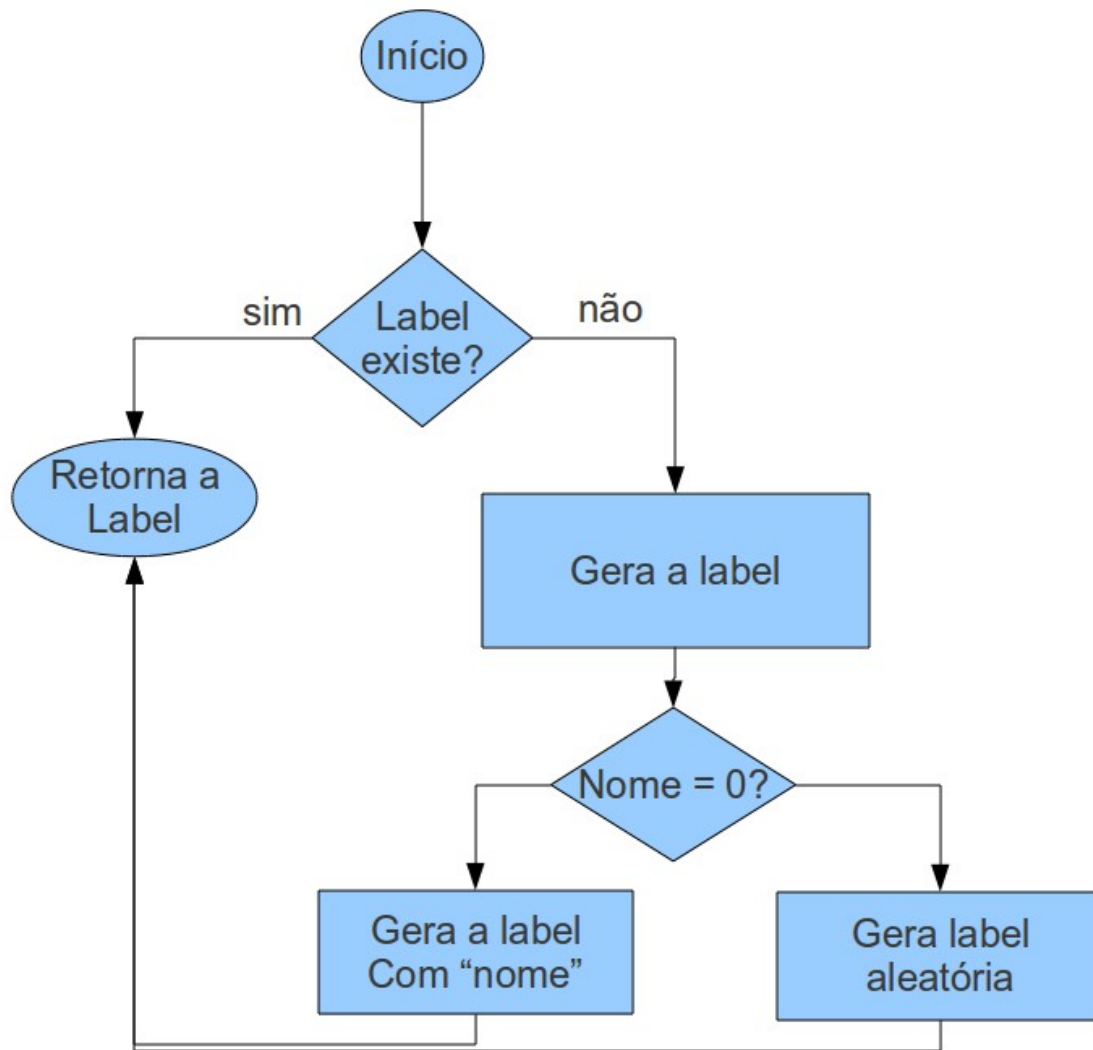
stdio.h
stdlib.h
string.h
desmontador.h

Estrutura de dados

```
typedef struct elementolabel{  
    char *label;  
    int endereco;  
    struct elementolabel *next;  
} labelS;
```



Fluxograma



buscarlabel

labelS *buscarlabel(labelS *pinicio, int endereco);

Busca na estrutura das labels se o endereço já existe, se existir, retorna a label correspondente, senão, retorna NULL.

Parâmetros

p_inicio:	Posição da primeira label da estrutura.
endereco:	Endereço que se está procurando.

Valor de retorno

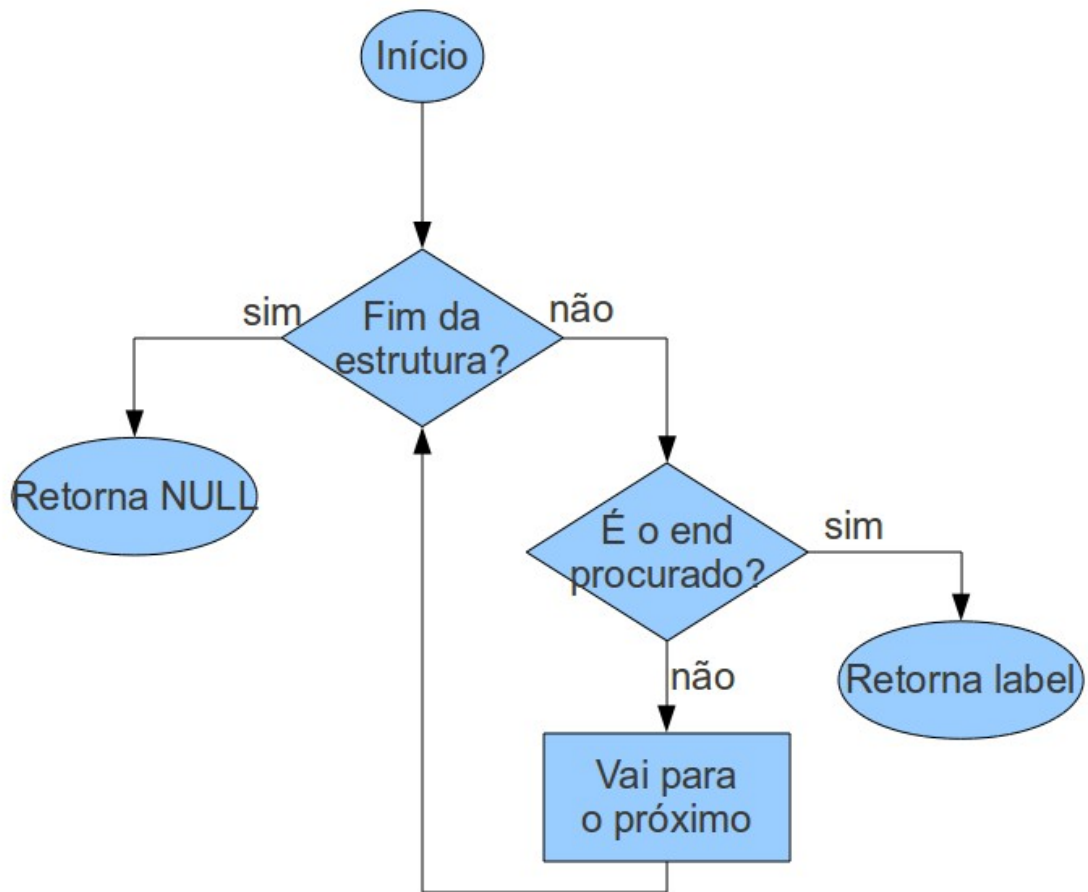
labelS *

Se existir a label, o retorno é a posição da label, se não existir é NULL.

Bibliotecas requeridas

stdio.h
estruturas.c

Fluxograma



imprimecodigo

```
void imprimecodigo(FILE *codigo_assembly, labelS *p_label, tabelaS  
*p_tabela, char *linguagemassembly, undadosS *p_dadosbss, codeS  
*p_newcode, dataS *p_newdata);
```

Imprime o código no arquivo alvo.

Parâmetros

codigo_assembly:	Nome do arquivo alvo que será impresso.
p_label:	Posição do início das labels.
p_tabela:	Posição do início da tabela
linguagemassembly:	Nome do programa para ser impresso em .begin.
p_dadosbss:	Posição do início dos dados inicializados.
p_newcode:	Posição do início do código.
p_newdata:	Posição do início dos dados não inicializados.

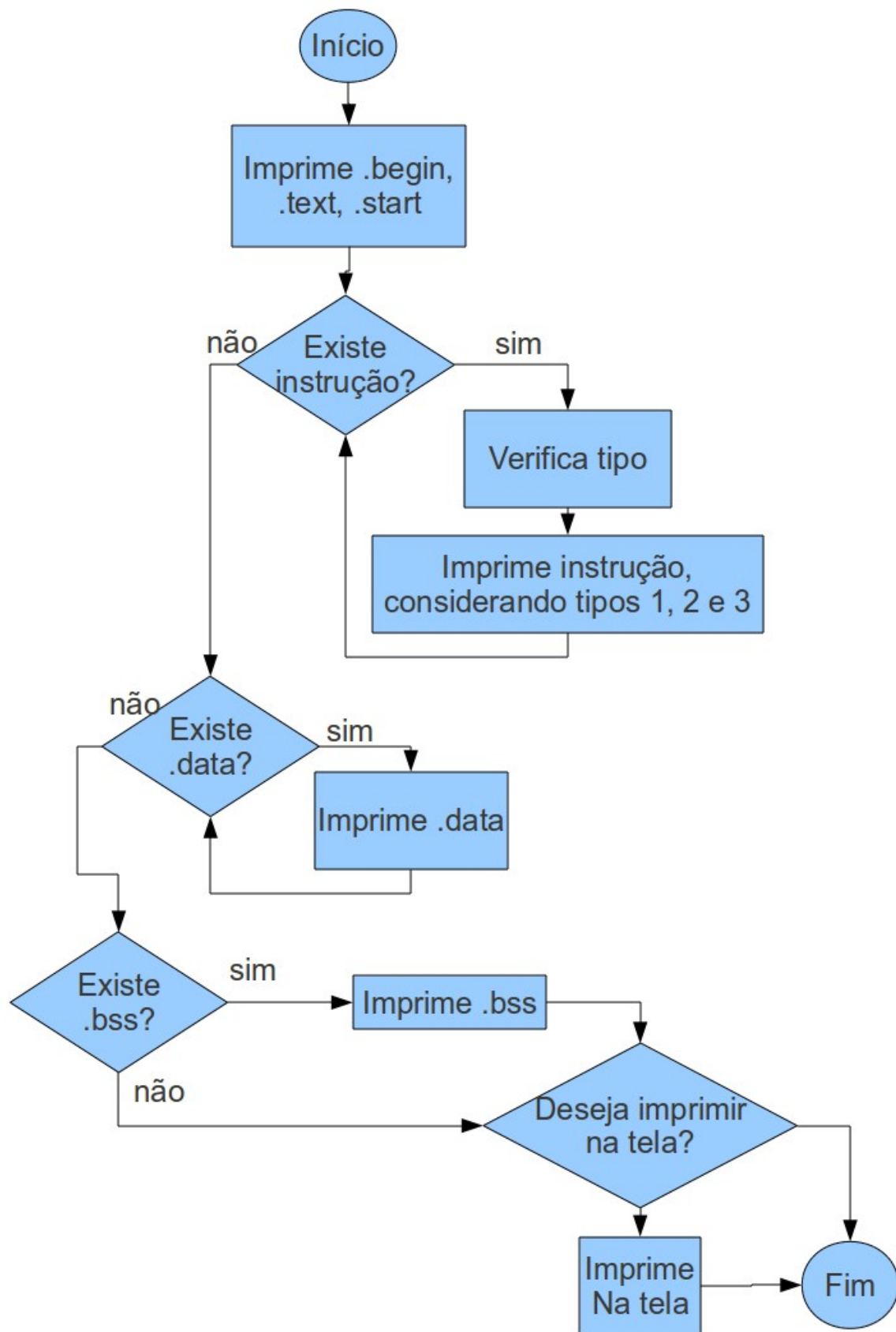
Valor de retorno

Void.

Bibliotecas requeridas

stdio.h
string.h
desmontador.h

Fluxograma



buscartabela

tabelaS *buscartabela(tabelaS *pinicio, int chave);

Busca na estrutura das labels se o endereço já existe, se existir, retorna a label correspondente, senão, retorna NULL.

Parâmetros

pinicio:	Posição da primeira estrutura da tabela.
chave:	Opcode que se está procurando.

Valor de retorno

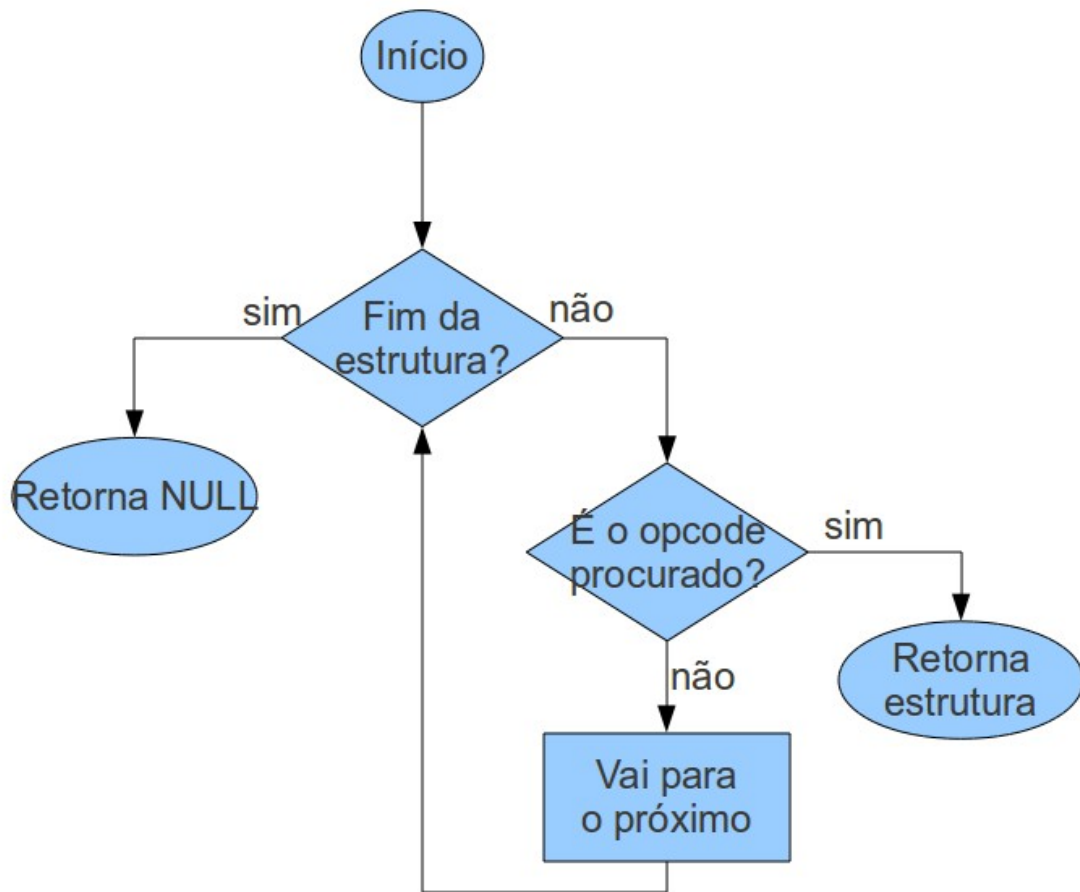
tabelaS *

Se existir o opcode, retorna a posição do elemento da tabela procurado, se não existir é NULL.

Bibliotecas requeridas

stdio.h
estruturas.c

Fluxograma



fechaarquivos

fechaarquivos(FILE **file1, FILE **file2, FILE **file3);

Fecha todos os arquivos necessários para desmontar o arquivo fonte.

Parâmetros

file1:	Arquivo que se deseja fechar.
file2:	Arquivo que se deseja fechar.
file3:	Arquivo que se deseja fechar.

Valor de retorno

void.

Bibliotecas requeridas

stdio.h

Fluxograma



desmontarcomdebug

desmontarcomdebug(char *linguagemmaquina, char *linguagemassembly);

Chama todas as funções necessárias para desmontar o arquivo pelo Section Header.

Parâmetros

linguagemmaquina: Nome do arquivo aonde se encontra a linguagem de máquina a ser desmontada.

linguagemassembly: Nome do arquivo alvo aonde se desmontará o arquivo..

Valor de retorno

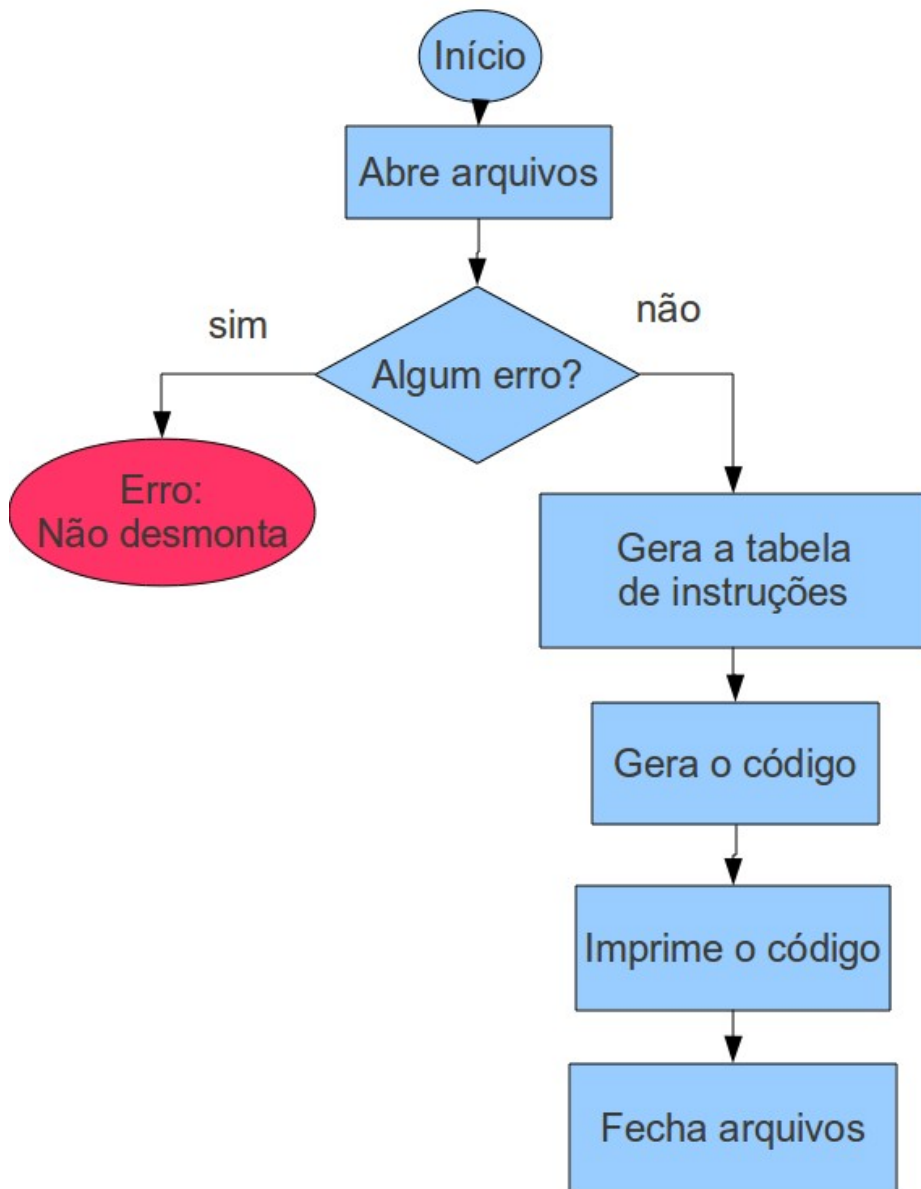
Void.

Bibliotecas requeridas

stdio.h

desmontador.h

Fluxograma



geracodigo

```
labelS *geracodigocomdebug(FILE *codigo, undadosS  
**p_iniciodadosun, codeS **p_newcode, tabelaS *p_tabela, dataS  
**p_newdata);
```

Lê o cabeçalho, lê o section header, lê as strings, gera as labels, lê as instruções, lê os dados (.data e .bss) e com tudo isso gera as estruturas de dados com o código, os dados e as labels.

Parâmetros

codigo:	Nome do arquivo fonte que será desmontado.
p_iniciodadosun:	Posição do início dos dados não inicializados.
p_newcode:	Posição do início da struct de código.
p_tabela:	Início da struct da tabela.
p_newdata:	Posição do início dos dados inicializados.

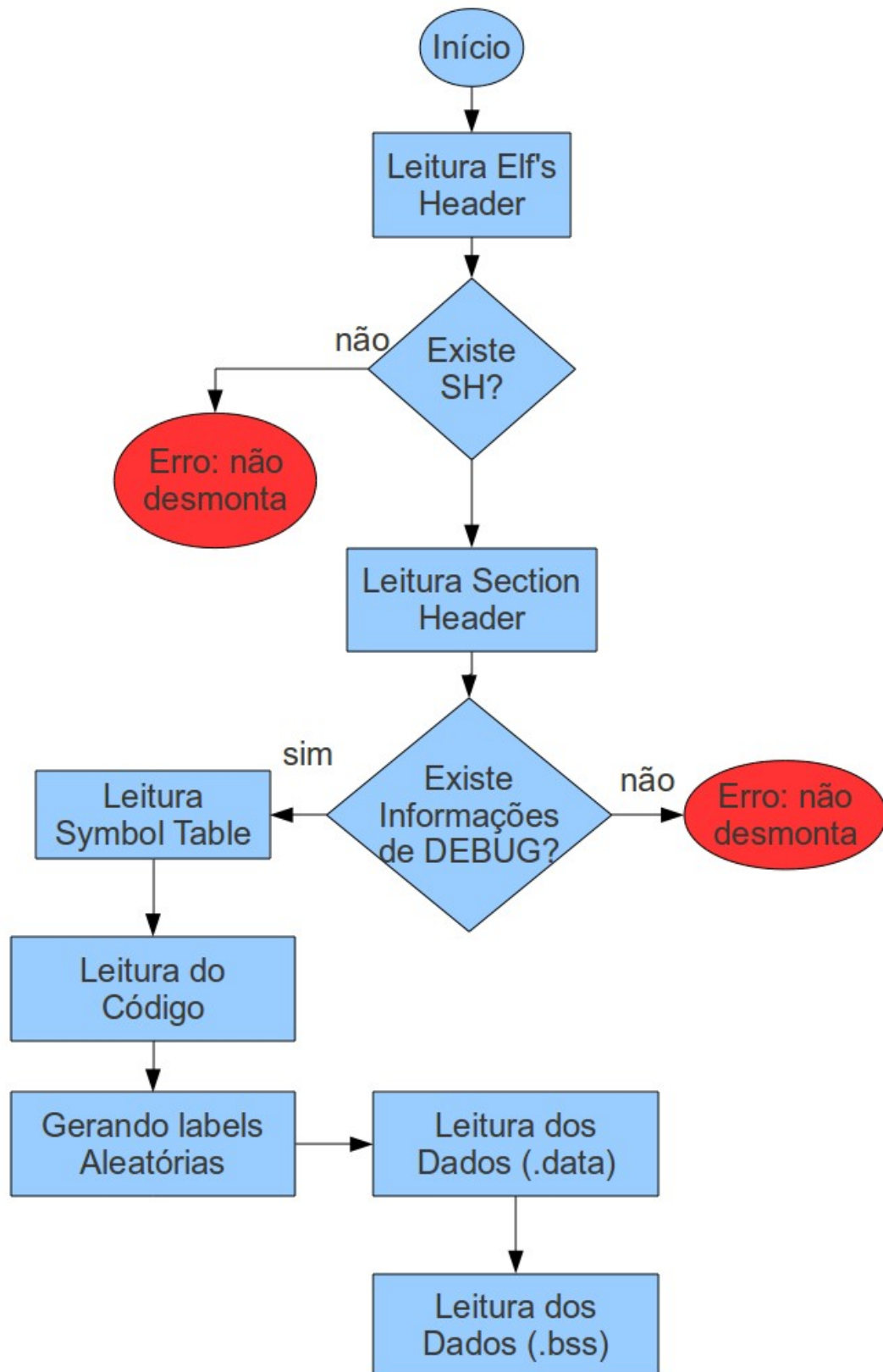
Valor de retorno

labelS *:	Ponteiro do inicio da posição onde ficará a primeira label criada.
-----------	--

Bibliotecas requeridas

- stdio.h
- stdlib.h
- string.h
- desmontador.h

Fluxograma



imprimenatela

```
void imprimenatela(labelS *p_label, tabelaS *p_tabela, char  
*linguagemassembly, undadosS *p_dadosbss, codeS *p_newcode, dataS  
*p_newdata);
```

Imprime o código na tela.

Parâmetros

p_label:	Posição do início das labels.
p_tabela:	Posição do início da tabela
linguagemassembly:	Nome do programa para ser impresso em .begin.
p_dadosbss:	Posição do início dos dados inicializados.
p_newcode:	Posição do início do código.
p_newdata:	Posição do início dos dados não inicializados.

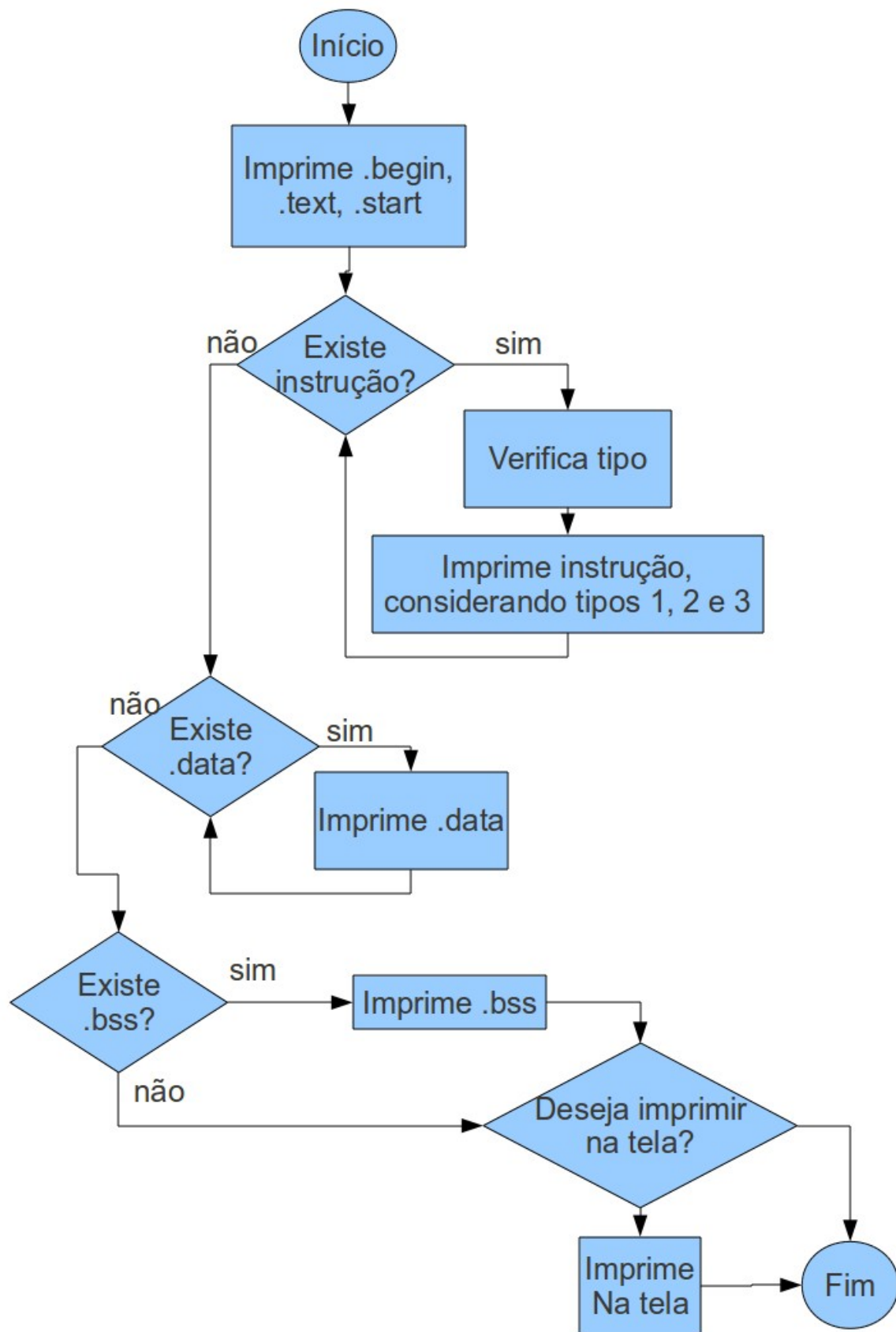
Valor de retorno

Void.

Bibliotecas requeridas

stdio.h
string.h
desmontador.h

Fluxograma



BIBLIOTECAS

stdio.h
stdlib.h
string.h
desmontador.h

MAKEFILE

CFLAGS = -c -g -ansi -Wall
OBJS = desmontador.o fechaarquivos.o desmontar.o geratabela.o geracodigo.o gerarlabel.o
buscartabela.o estruturas.o buscarlabel.o imprimecodigo.o imprimenatela.o estruturas_elf.o
geracodigocomdebug.o desmontarcomdebug.o

desmontador: \$(OBJS)
gcc -o desmontador \$(OBJS)

fechaarquivos.o: fechaarquivos.c
gcc \$(CFLAGS) fechaarquivos.c

desmontar.o: desmontar.c
gcc \$(CFLAGS) desmontar.c

desmontarcomdebug.o: desmontarcomdebug.c
gcc \$(CFLAGS) desmontarcomdebug.c

geratabela.o: geratabela.c
gcc \$(CFLAGS) geratabela.c

geracodigo.o: geracodigo.c
gcc \$(CFLAGS) geracodigo.c

geracodigocomdebug.o: geracodigocomdebug.c
gcc \$(CFLAGS) geracodigocomdebug.c

gerarlabel.o: gerarlabel.c
gcc \$(CFLAGS) gerarlabel.c

buscartabela.o: buscartabela.c
gcc \$(CFLAGS) buscartabela.c

buscarlabel.o: buscarlabel.c
gcc \$(CFLAGS) buscarlabel.c

estruturas.o: estruturas.c
gcc \$(CFLAGS) estruturas.c

estruturas_elf.o: estruturas_elf.c
gcc \$(CFLAGS) estruturas_elf.c

imprimecodigo.o: imprimecodigo.c


```
gcc $(CFLAGS) imprimecodigo.c
```

```
imprimenatela.o: imprimenatela.c  
gcc $(CFLAGS) imprimenatela.c
```

```
desmontador.o: desmontador.c desmontador.h  
gcc $(CFLAGS) desmontador.c
```

```
clean:  
rm -f desmontador *.o codigo_assembly.txt
```

```
roteiroDEBUG:  
./desmontador -g Roteiro01  
./desmontador -g Roteiro02  
./desmontador -g Roteiro03  
./desmontador -g Roteiro04  
./desmontador -g Roteiro05
```

```
roteiro:  
./desmontador Roteiro01  
./desmontador Roteiro02  
./desmontador Roteiro03  
./desmontador Roteiro04  
./desmontador Roteiro05
```

```
extrasDEBUG:  
./desmontador -g Extra01  
./desmontador -g Extra02  
./desmontador -g Extra03  
./desmontador -g Extra04  
./desmontador -g Extra05  
./desmontador -g Extra06  
./desmontador -g Extra07  
./desmontador -g Extra08  
./desmontador -g Extra09  
./desmontador -g Extra10
```