

Universidade Federal de Minas Gerais
Bacharelado em Ciência da Computação
Projeto Orientado em Computação

Transporte do Sistema SIC de MS-DOS para ambiente Windows

Área: Compiladores
Orientandos: Marco Rodrigo Costa
 Valeska Gonçalves Russo
Orientadora: Mariza Andrade da Silva Bigonha
Data: 21/06/96

Roteiro da Apresentação

- Introdução
 - Sistema de Implementação de Compiladores — SIC
- Projeto
 - Motivações
 - Etapas Principais
- Conclusões

Sistema de Implementação de Compiladores SIC

- O que é o SIC
 - Ferramenta adequada à escrita de compiladores
 - Implementado em Pascal
 - Plataforma MS-DOS
- Como funciona o SIC
 - Arquivo de entrada: Programa em SIC
 - Compilação em três fases
 - Geração do compilador em Pascal
- Interface do SIC

O Projeto

- Motivações
 - Extensão do compilador para uma nova linguagem
 - Mudança de plataforma

- Etapas Principais
 - Geração do Compilador em C
 - Interface WINDOWS

Geração do Compilador em C

- Primeira Etapa
 - Estudo de Técnicas de Compilação
 - Estudo do Pacote Básico de Gerência de Interface: WINDOW 2.0 (Bigonha, 1990)
 - Estudo do SIC: uso e implementação

Geração do Compilador em C

- Segunda Etapa: Fase de Implementação
 - Transporte do Analisador Sintático para C
 - Tradução do Compilador
 - Levantamento dos Problemas
 - Correspondência entre Pascal e C
 - Alteração da sintaxe do SIC
 - Alteração do Manual do Usuário
 - Alerta ao Projetista de Compiladores

Interface Windows

- Primeira Etapa
 - Estudo do Compilador Delphi
- Segunda Etapa: Fase de Implementação
 - Criação das janelas
 - Transporte do SIC versão C para Delphi
 - Levantamento dos Problemas
 - Programação por Eventos
 - Problema do Segmento de Dados
 - Alteração do Manual do Usuário

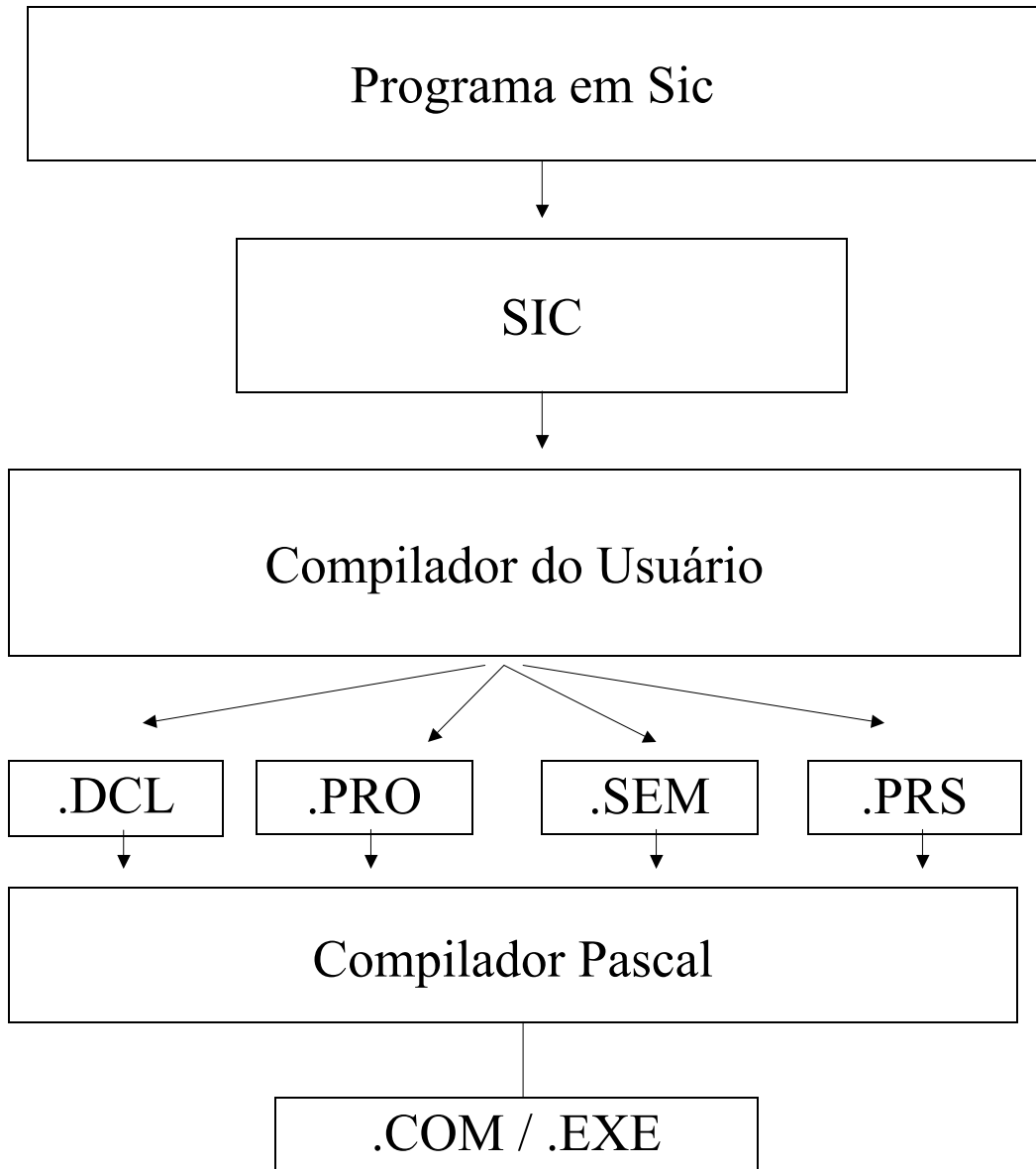
Interface Windows

- Principais Preocupações
 - Interface simples
 - Realimentação
 - Minimização dos erros

Conclusões

- Principais Resultados
 - Nova Plataforma de Desenvolvimento de Compiladores
 - Ambiente mais amigável
- Possíveis Extensões
 - Transporte dos demais analisadores sintáticos para C
 - Acoplamento das versões Pascal e C
 - Criação de um Help-on-line
 - Incorporação do Editor de Texto

Sistema de Implementação de Compiladores SIC



Sistema de Implementação de Compiladores SIC

- Trechos de um programa em SIC (versão Pascal):

```
%%COMPILER PROGRAM exemplo; %%BATCH
```

```
%%TOKENS
```

```
"id"    = xid ;  
"cte"   = xcte ;  
"program " = xprogram    ...
```

```
%%STACK 50 OF ATTRIBUTES
```

```
exp    = (r,tipo:INTEGER) { exp.r := 0; exp.tipo := INTEIRO };  
"id"   = (valor :INTEGER) { "id".valor := 1 }  
"cte"  = (valor :INTEGER)    ...
```

```
%%CONSTANTS
```

```
MAXMSG  = 20;  
MAXERROR = 11;
```

```
%%TYPES
```

```
erros = packed array[1..MAXMSG] of char;    ...
```

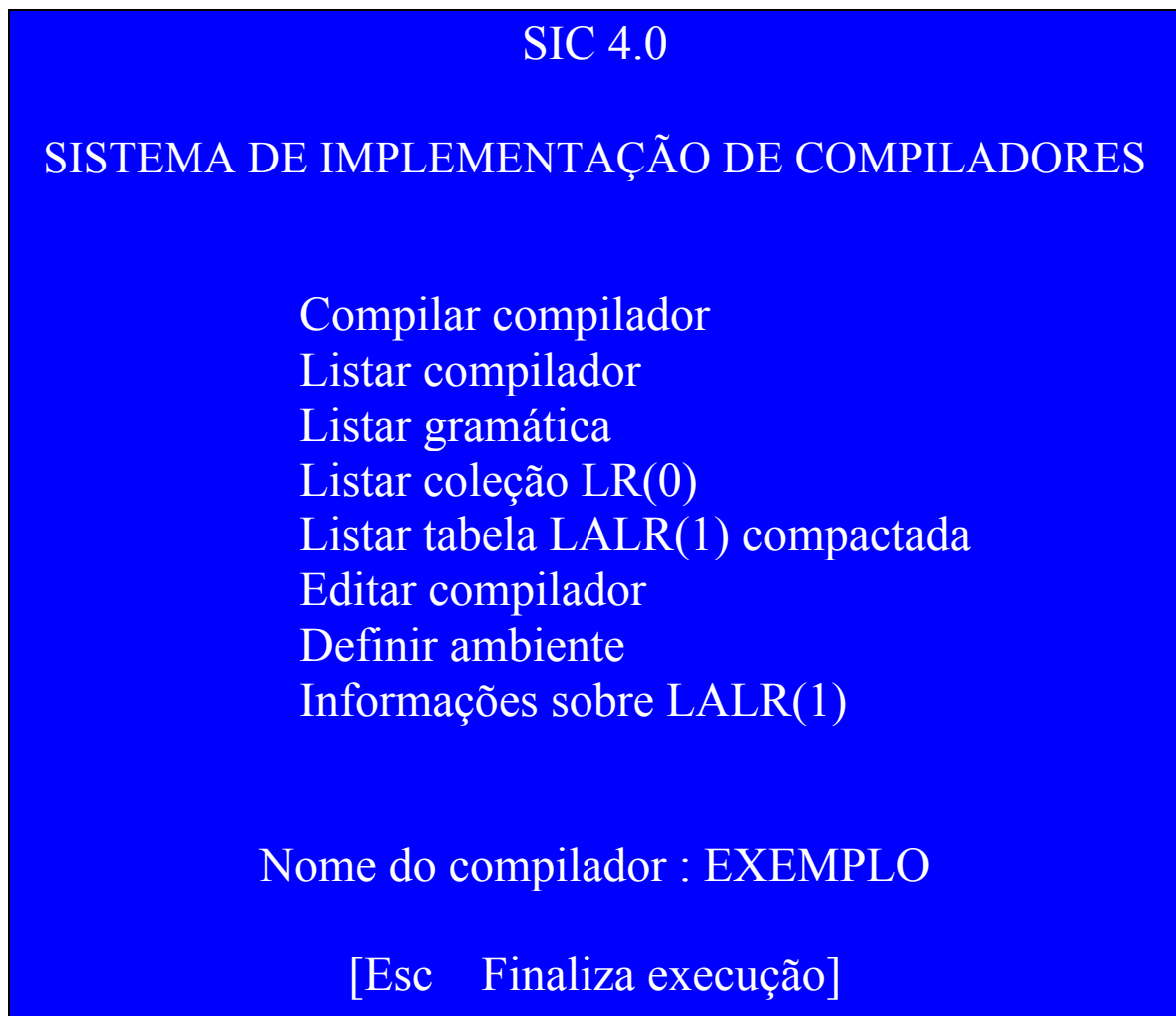
```
%%GRAMMAR program AND SEMANTICS
```

```
program = proghead dcls cmdc  
        {SALVA(1,proxq-1);  
        GEN(CPROGEND,0,0,0); };
```

```
proghead = "program"  
        { toff := 0;  
        offset := 0;  
        proxq := 1;  
        GEN(CPROGBEGIN,0,0,0); };
```

Sistema de Implementação de Compiladores SIC

- Janela Principal do SIC



Geração do Compilador em C

- Tradução do Compilador:

- Versão Pascal:

```
procedure ATRIBUTOS(var fsymb:SYMSET);
```

```
...
```

```
    WRITELNE(typesic); write(typesic, 'YYBUFFER = record');  
    WRITELNE(typesic); write(typesic, ':14,bufsimb : integer;');  
    WRITELNE(typesic); write(typesic, ':14,buflinha: integer;');  
    WRITELNE(typesic); write(typesic, ':14,bufpos : 0..255;');  
    WRITELNE(typesic); write(typesic, ':14,bufsem : YYTATR;');  
    WRITELNE(typesic); write(typesic, ':10,end;');
```

```
...
```

- Versão C:

```
procedure ATRIBUTOS(var fsymb:SYMSET);
```

```
...
```

```
    WRITELNE(typesic); write(typesic, 'typedef struct { ');  
    WRITELNE(typesic); write(typesic, ':14,int bufsimb; ');  
    WRITELNE(typesic); write(typesic, ':14,int buflinha; ');  
    WRITELNE(typesic); write(typesic, ':14,int bufpos; ');  
    WRITELNE(typesic); write(typesic, ':14,YYTATR bufsem; ');  
    WRITELNE(typesic); write(typesic, ':10,} YYBUFFER;');
```

```
...
```

Geração do Compilador em C

- Correspondência entre Pascal e C:
 - Eliminação de declarações de labels
 - Arranjos em Pascal definidos de [MIN...MAX]:
Arranjos em C iniciados em 0.
 - Criação de procedimento para inclusão de protótipos no arquivo de declarações.
 - Registro Variante: Union em C
 - Alteração da ordem dos identificadores

Geração do Compilador em C

- Exemplo de alteração da ordem dos identificadores:

- Versão Pascal:

- Definição de atributos no arquivo de entrada:

```
exp = (r, tipo : integer)
```

- Código gerado no compilador Pascal do usuário:

```
CASE YY500Y of ...
```

```
2:(exp : record r,tipo:INTEGER end) ;
```

```
...
```

```
END;
```

- Versão C:

- Definição de atributos no arquivo de entrada:

```
exp = (r, tipo : int)
```

- Código gerado no compilador C do usuário:

```
typedef union
```

```
{ ...
```

```
struct { int r , tipo; } exp;
```

```
...
```

```
} YYTATR;
```

Geração do Compilador em C

- Um ponto sensível a geração de erros:
 - Na definição de rotinas semânticas (versão Pascal):

```
exp = "id"
{ ...
if FOIDECLARADO("id".valor) and
  ((classe = VARIAVEL) or (classe = PAR)) then
  exp.tipo := ts["id".valor].tipotam;
else begin
  ERRO(7,"id".YYLINHA,"id".YYPOS) ;
  exp.tipo := INTEIRO;
end;
};
```

- Na definição de rotinas semânticas (versão C):

```
exp = "id"
{ ...
if (FOIDECLARADO("id".valor) &&
  ((classe == VARIAVEL) || (classe == PAR)))
  exp.tipo = ts["id".valor].tipotam;
else {
  ERRO(7,"id".YYLINHA,"id".YYPOS) ;
  exp.tipo = INTEIRO;
}
};
```


Geração do Compilador em C

- Alteração da sintaxe do SIC:
 - Única Alteração: Seção de Atributos
 - Na versão Pascal a identificação do tipo dos atributos é composta por apenas uma palavra. Por exemplo:

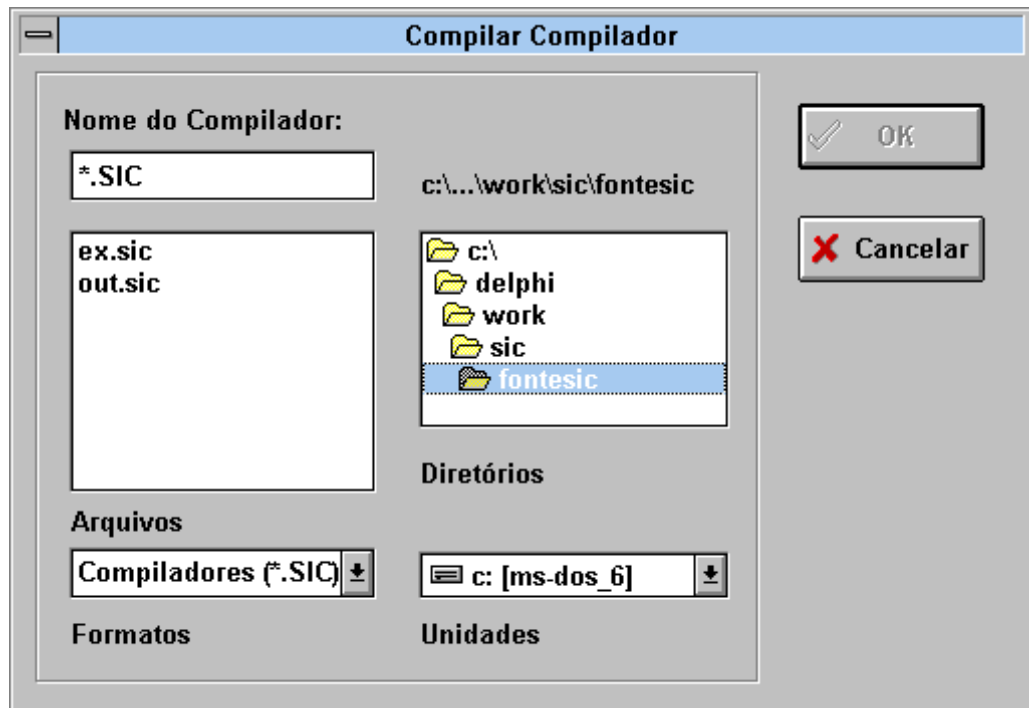
exp = (r, tipo : **integer**)

- Na versão C a identificação do tipo dos atributos pode ser composta por mais de uma palavra. Por exemplo:

exp = (r, tipo: **unsigned int**)

Interface Windows

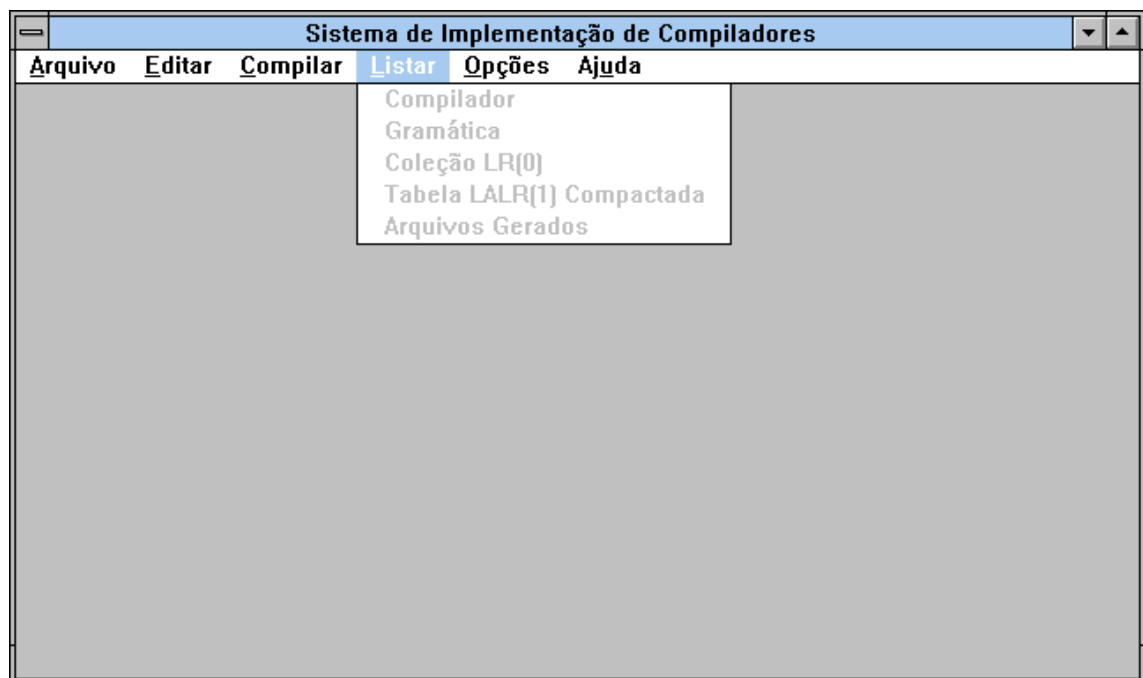
- Janela Compilar Compilador



- Minimização dos Erros:
 - Na abertura da janela, o botão ok fica desabilitado e são listados apenas arquivos de extensão SIC:

Interface Windows

- Minimização dos Erros:
 - Somente após a compilação de um compilador é que certas funções como listar compilador, listar gramática, etc ficam disponíveis:



Sistema de Implementação de Compiladores SIC

- Trechos de um programa em SIC (versão C):

```
%%COMPILER %%BATCH
```

```
%%TOKENS
```

```
"id"    = xid ;  
"cte"   = xcte ;  
"program" = xprogram
```

```
%%STACK 50 OF ATTRIBUTES
```

```
exp    = (r, tipo:int) { exp.r = 0; exp.tipo = INTEIRO; };  
"id"   = (valor :int) { "id".valor = 1; }  
"cte"  = (valor :int)
```

```
%%CONSTANTS
```

```
#define MAXMSG 20  
#define MAXERROR 11
```

```
%%TYPES
```

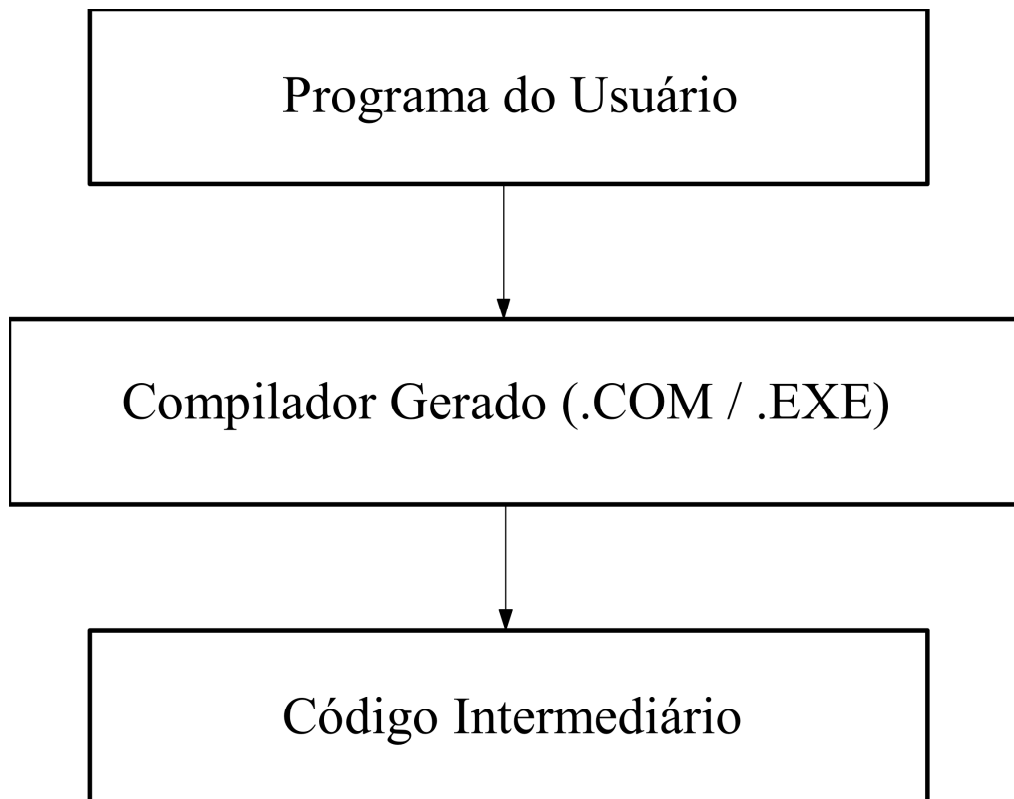
```
typedef char erros[MAXMSG+1];
```

```
%%GRAMMAR program AND SEMANTICS
```

```
program = proghead dcls cmdc  
        {  
            SALVA(1,proxq-1);  
            GEN(CPROGEND,0,0,0);  
        };
```

```
proghead = "program"  
        {  
            toff = 0;  
            offset = 0;  
            proxq = 1;  
            GEN(CPROGBEGIN,0,0,0);  
        };
```

Sistema de Implementação de Compiladores SIC



Sistema de Implementação de Compiladores SIC

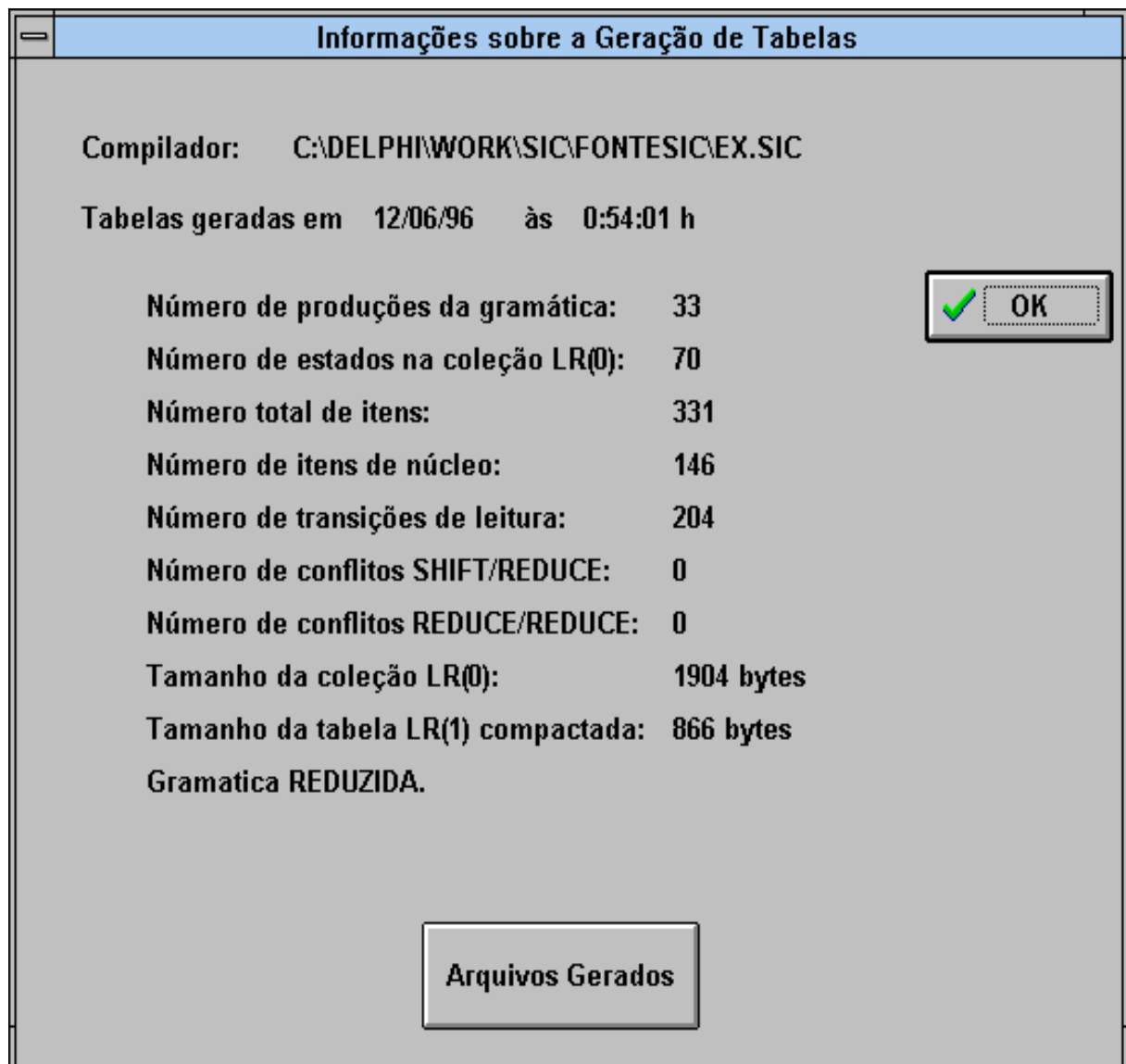
- Exemplo de programa do usuário
- Entrada para o compilador :

```
program
i:integer;
j:integer
begin
  i:=0;
  j:=2
end
```

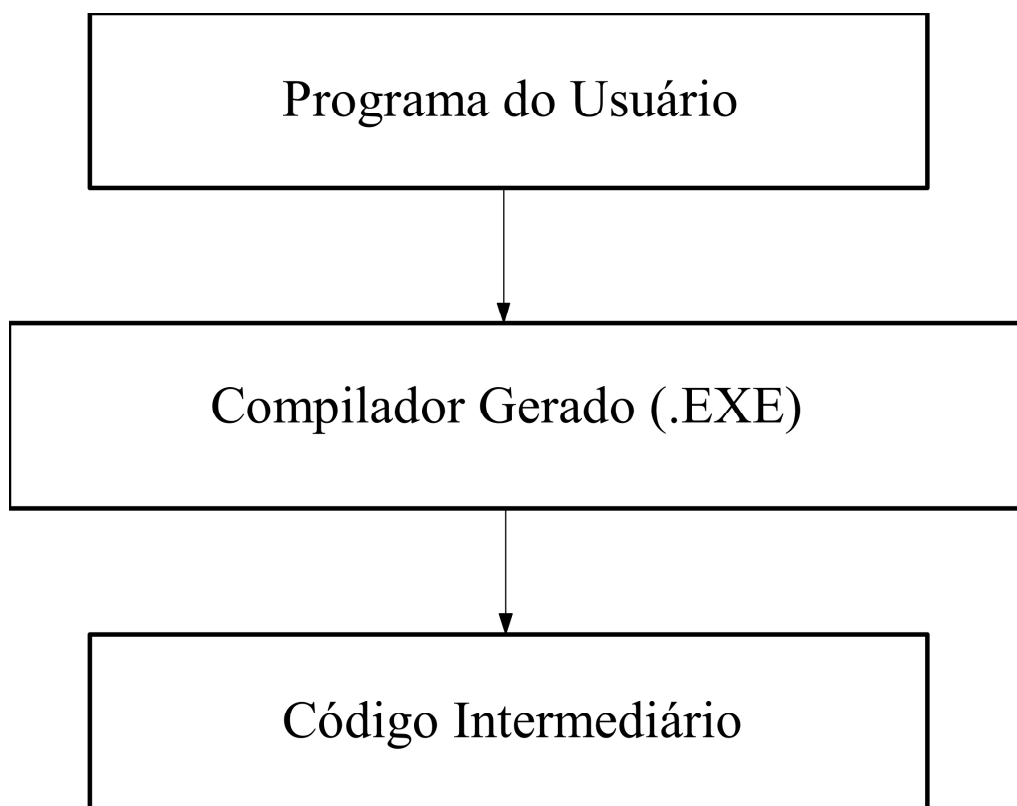

j 1 1 1 1
Parse completed, sintaxe ok

Interface Windows

- Janela com informações sobre a geração das tabelas:



Sistema de Implementação de Compiladores SIC



Geração do Compilador em C

- Alerta ao Projetista de Compiladores
 - Pontos sensíveis a geração de erros na especificação do compilador:

- Definição de rotinas semânticas

{ }

- Definição das palavras reservadas do SIC

%%TOKENS, %%GRAMMAR, etc.

Sistema de Implementação de Compiladores SIC

