

Trabalho Prático

A construção de um compilador para uma linguagem gráfica que executa scripts de animação

Parte IV - Implementação do Gerador de Código

Data de entrega: 04/06/2013

Nesta fase, deverão ser adicionadas ações semânticas que gerem código para o computador **GCX**, transformando o analisador semântico em um tradutor. O tradutor deverá gerar um arquivo texto contendo os mnemônicos em **ASSEMBLY GCX**, conforme a tabela de instruções. Além disso, deverá ser gerado um arquivo de inteiros, contendo o código-objeto correspondente, o qual será executado com o auxílio do simulador do **GCX**.

1. Modifique a tabela de símbolos, acrescentando o campo **endereço** que armazenará a posição de memória inicial da área reservada para cada identificador.
2. Crie variáveis globais para os contadores de programa (código), memória de dados, rótulos, temporários e uma variável para a base de temporários. Lembre-se de incrementá-las sempre que reservar memória, criar temporário, rótulo ou gravar instruções. A base dos temporários é determinada após o bloco de declarações. O contador de temporários deve ser reinicializado sempre que um comando chamar uma expressão.
3. Crie uma lista de rótulos que deverá ter campos para o endereço de desvio (memória de código relativa ao registrador CS) e para a posição do arquivo de código-objeto onde deverá ser gravado (processo de retrocorreção).
4. Implemente a geração de código para declarações. Teste esta parte antes de passar para a próxima.
5. Implemente a geração de código para expressões e para o comando PAUSE. Teste as expressões, começando por expressões simplificadas até expressões mais complexas.
6. Implemente a geração de código para o comando WHILE. Teste o comando junto com expressões relacionais.
7. Implemente a geração de código para os comandos restantes. Teste o tradutor completo.

8. Implemente um procedimento para fazer a retrocorreção, gravando os rótulos em aberto nas posições correspondentes no arquivo-objeto.
9. Teste o código executável gerado, com o auxílio do simulador do GCX.

Pontos extras: Implemente uma rotina de otimização de código baseada, no mínimo, na técnica de *peephole*.

O que entregar:

- Códigos-fontes
- Esquema de tradução

Obs: Leia as especificações gerais contidas no documento “Descrição do trabalho”.