

LIPAD – Linguagem de Programação para Algoritmos Distribuídos

O Ambiente Computacional Virtual (ACV) [1] é uma ferramenta de apoio didático, que auxilia o aprendizado de disciplinas como: Teoria de Linguagens, Linguagens Formais e Autômatos e Compiladores entre outras. Ele proporciona a montagem de uma topologia de rede do tipo barramento, onde n computadores virtuais podem ser acoplados. Tendo a LIPAD a finalidade de escrever programas que possam ser executados sobre esta topologia virtual.

A LIPAD é uma linguagem de programação descrita para implementar algoritmos que possam ser executados por um ou mais processadores. Contendo um mecanismo de sincronismo entre processos através de troca de mensagens para proporcionar a exclusão mútua da área compartilhada de memória.

Sua sintaxe é baseada no Pascal, porém contendo também comandos que proporcionam o sincronismo entre processos e a decomposição do algoritmo em partes a serem executadas por diferentes processadores. Esta distribuição do algoritmo pela rede ficará totalmente a cargo do programador, devendo este ter total conhecimento da linguagem que está utilizando.

As variáveis definidas pela linguagem pertencem a dois grupos: variáveis não-compartilhadas e variáveis compartilhadas. Cada processo aloca na memória uma área para os seus dados, onde localiza as *variáveis não-compartilhadas*, e fica a cargo do compilador reservar uma área da memória para as *variáveis compartilhadas*. Acessos para leitura e escrita nesta memória podem ser feitos quando um processo necessitar de uma variável compartilhada, utilizando mecanismos de comunicação entre, invisíveis ao programador.

Existe também o conceito de *blocos distribuídos*, que são executados por um processador especificado no início do bloco.

O compilador da LIPAD utiliza todos os recursos de hardware disponibilizados pela máquina virtual (MVR) do ACV [2], podendo ainda ser extensível a melhorias como: otimização de código e orientação a objetos.

Este compilador é composto basicamente por módulos que fazem a compilação total, em fases bem definidas, buscando uma maior modularidade e facilidade de desenvolvimento.

Utiliza-se a tradução do programa escrito em LIPAD para um código intermediário (MEPA) [3], por ser este mais simples de ser traduzido para uma máquina qualquer, o que facilita a construção do tradutor para código final. As fases subsequentes geram a linguagem de máquina reconhecida pela MVR e em seguida o código executável.

No caso de programas que utilizem os mecanismos de distribuição de processamento fornecidos pela LIPAD, o compilador gera múltiplos códigos executáveis que serão executados por processadores específicos, conforme especificado no código fonte.

Na implementação deste compilador foram escolhidas ferramentas que propiciassem os resultados esperados. Foram escolhidos a ferramenta Lex para implementar o analisador léxico, a ferramenta Yacc para implementar a gramática da LIPAD e a linguagem de programação C++ para implementar a biblioteca de suporte.