

PIPCA – Técnicas de Programação – 2017/01 – Prof. João Gluz

Atividade Adicionais da Avaliação 1 – Resolução de Problemas de Programação em Prolog

5. Grafos

Identificando o Caminho Crítico de um Projeto

O caminho crítico é o caminho mais longo de um projeto, ou seja, aquele formado pela sequência de atividades cujas durações das atividades somadas definem o prazo total estimado para o projeto. Se o prazo final for excedido, é porque pelo menos alguma das atividades do caminho crítico não foi concluída no prazo estimado.

O caminho crítico pode ser encontrado pela construção de um modelo gráfico (um grafo dirigido sem laços, também denominado de rede) baseado nas dependências que existem entre os objetivos ou atividades terminais da estrutura analítica de projeto e na duração estimada para cada uma dessas atividades.

Existem dois tipos de modelos gráficos possíveis, um com as atividades nos nodos (sigla em inglês AoN – *Activities on Nodes*) e o outro com as atividades nos arcos (sigla em inglês AoA – *Activities on Arrows*). Em ambos casos o caminho crítico pode ser localizado através de um algoritmo que percorre o grafo e identifica o caminho mais longo que vai do nodo de origem ao nodo de fim de projeto. Neste trabalho será considerada apenas a representação AoN.

Considerando a estrutura dos termos que representam os objetivos (atividades) terminais de uma estrutura analítica de projeto, tal como definida na seção anterior:

objterm(O,R,C,T)

onde O identifica o objetivo correspondente ao nodo, R é uma lista de identificadores de recursos (pessoas, equipamentos, materiais, etc.) necessários para atingir o objetivo, C é um valor numérico com a estimativa de custo necessário para atingir o objetivo e T uma estimativa em dias do tempo necessário para alcançar o objetivo.

Agora, desenvolva e teste os seguintes predicados:

- *extrai_grafo_aon*(AG,G):

Este predicado extrai da estrutura analítica de projeto, representada na árvore genérica AG, um grafo identificado por G, onde cada nodo deste grafo corresponde a um objetivo terminal (que também é uma atividade terminal) e os arcos entre estes nodos correspondem a relações de dependência por recursos entre essas atividades.

Para que o objetivo/atividade terminal O1 definido por *objterm*(O1,R1,C1,D1) seja dependente do objetivo/atividade terminal O2 definido por *objterm*(O2,R2,C2,D2), basta que o identificador de O2 apareça na lista de recursos R1 necessários para que o objetivo O1 seja finalizado.

O predicado *extrai_grafo_aon*(AG,G) adiciona os nodos necessários na base de fatos através do predicado *assert*(F), onde F é um fato Prolog.

Além de adicionar fatos no formato:

arco(G,O1,O2)

que indica que há um arco do nodo correspondente ao objetivo O1 para o nodo correspondente ao objetivo O2 no grafo G, o predicado *extrai_grafo_aon*(AG,G) também adiciona nodos no formato:

nodo(G,O,D)

que indica que N é um nodo do grafo G e que o prazo (duração) estimado para o objetivo O.

- *caminho_critico*(G,L)

Predicado que identifica o caminho critico do projeto representado pelo grafo G. A lista L retorna os nodos deste caminho crítico.

- *prazo_caminho_critico*(G,P)

Predicado que retorna em P o prazo estimado para o caminho critico do projeto representado pelo grafo G.