

Primeira Prova de Análise e otimização de Código  
- DCC888 -  
Ciência da Computação

Nome: \_\_\_\_\_  
“Eu dou minha palavra de honra que não trapacearei neste exame.”

Número de matrícula: \_\_\_\_\_

As regras do jogo:

- A prova é sem consulta.
- Quando terminar, não entregue nada além do caderno de provas para o instrutor.
- Quando escrever código, a sintaxe correta é importante.
- Cada estudante tem direito a fazer uma pergunta ao instrutor durante a prova. Traga o caderno de provas quando vier à mesa do instrutor.
- A prova termina uma hora e quarenta minutos após seu início.
- Seja honesto e lembre-se: **você deu sua palavra de honra.**

Alguns conselhos:

- Escreva sempre algo nas questões, a fim de ganhar algum crédito parcial.
- Você pode sacrificar sua pergunta para saber porque os seres humanos riem, segundo *Robert Heilen*.
- Se não entender a questão, e já tiver gasto sua pergunta, escreva a sua interpretação da questão junto à resposta.
- A prova não é difícil, ela é divertida, então aproveite!

Tabela 1: Pontos acumulados (para uso do instrutor)

Questão 1	Questão 2	Questão 3	Questão 4

1. Abaixo vemos um sistema de restrições extraído de uma linguagem de programação que possui ponteiros.

$h = \&c$	$e = \&g$	$b = c$
$h = \&g$	$h = a$	$c = b$
$a = \&e$	$f = d$	$b = a$
$d = *h$	$*e = f$	$f = \&a$

(a) (3 Pontos) Desenhe o grafo de restrições que seria criado para os comandos acima, logo antes de começarmos a resolver a análise. Estamos assumindo, aqui, uma análise segundo o estilo de *Andersen*.

(b) (3 Pontos) Mostre o grafo de restrições ao final da análise, quando todas as arestas candidatas já houverem sido avaliadas. Não é necessário contrair ciclos.

(c) (3 Pontos) Escreva uma tabela relacionando cada variável com as posições de memória para onde ela pode apontar. As posições para onde uma variável pode apontar são dadas ao término da avaliação da análise de ponteiros.

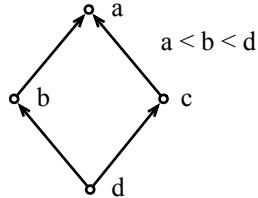
(d) (1 Ponto - Extra) Descreva como morreram pelo menos 37 ladrões da estória Ali Babá e os Quarenta Ladrões.

2. Alguns professores pesquisadores no Brasil recebem a chamada *bolsa de produtividade* em pesquisa. A hierarquia de pesquisadores é formada por cinco níveis: pesquisador nível 2, 1d, 1c, 1b e 1a. O nível mais alto é 1a, e o mais baixo é 2. Um pesquisador é frequentemente contactado pelo CNPq, a agência que financia a bolsa, para que ele avalie projetos de pesquisa. O professor pode avaliar o projeto, ou pode pedir à algum outro pesquisador, cujo *ranking* seja menor que o seu, para avaliar o projeto. Para evitar sobrecargas de professores, um professor, tendo a oportunidade de solicitar uma avaliação a vários colegas, deve escolher aquele colega de maior *ranking*.

A título de exemplo, podemos assumir que em nosso departamento temos os seguintes professores:  $\{\text{Laender} \leftrightarrow 1a, \text{Nivio} \leftrightarrow 1a, \text{Wagner} \leftrightarrow 1b, \text{Renato} \leftrightarrow 1c, \text{Mi} \leftrightarrow 1d, \text{Fer} \leftrightarrow 2, \text{Edu} \leftrightarrow 2, \text{Flip} \leftrightarrow 2\}$ . Nesse exemplo, o professor *Laender*, cujo *ranking* é o maior possível, pode solicitar uma avaliação a qualquer colega, menos ao professor *Nívio*, que possui o mesmo *ranking*. O professor *Laender*, nesse caso, deve escolher o professor *Wagner*, pois ele possui o maior *ranking* dentre os colegas disponíveis. Os professores de nível 2, se acaso forem sobrecarregados com projetos, podem enviar alguns desses projetos para a secretária *Soninha*, que os remeterá de volta ao CNPq. Nesta questão e na próxima você vai elaborar uma análise estática que determina o caminho que um professor pode tomar para encontrar um avaliador para seus projetos. Começaremos determinando um semi-reticulado para a sua análise. Lembre-se, um semi-reticulado é uma estrutura algébrica  $(S, \leq, \wedge, \perp)$ , sendo  $S$  um conjunto,  $\leq$  uma ordem parcial sobre os elementos do conjunto,  $\wedge$  um operador de encontro que fornece o maior limite inferior de seus operandos, e  $\perp$  um elemento que é menor que todos os outros.

(a) (2 Pontos) Determine o conjunto  $S$  sobre o qual existe o seu reticulado.

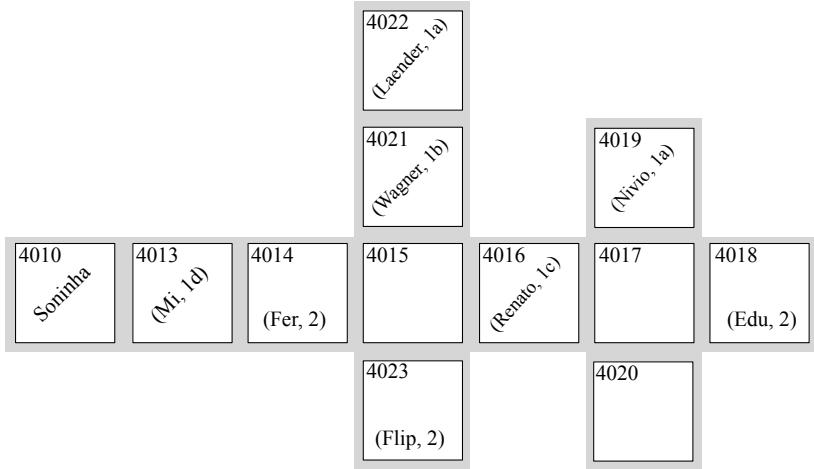
(b) (3 Pontos) Determine a ordem parcial do reticulado. Para isso, use um diagrama, de forma tal que uma aresta  $u \rightarrow v$  nesse grafo indica que  $v < u$ . Por exemplo:



(c) (2 Pontos) Qual é o limite inferior,  $\perp$ , desse reticulado?

(d) (3 Pontos) Determine o operador de encontro de informação, isto é,  $\wedge$ . Use uma tabela quadrada para determinar esse operador. As linhas e colunas da tabela devem conter todos os elementos do conjunto  $S$ .

3. Essa questão também refere-se ao reticulado de professores pesquisadores. Os professores do departamento de ciência da computação estão alocados em salas segundo a planta abaixo:



No que se segue, criaremos um sistema de restrições para um professor saber em que direção caminhar para encontrar alguém que lhe avalie um projeto. Não é necessário encontrar o menor caminho.

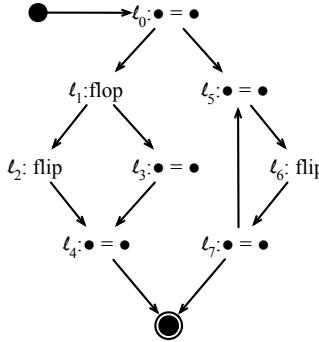
- (a) (3 Pontos) Uma análise estática precisa associar informação a alguma estrutura bem definida. Por exemplo, podemos associar informação a pares *variable*  $\times$  *Program Point*, ou diretamente a variáveis, no caso de uma análise esparsa. Isso, obviamente, se estamos falando de programas. Agora, contudo, estamos falando do mapa do departamento de computação. Defina um grafo ao qual podemos associar informações, e mostre como informação será associada. Note que informação pode ser associada a vértices, arestas, pares de vértices, conjuntos de arestas, etc. Por *informação*, referi-mo-nos a um ponto no reticulado definido na questão anterior.
- (b) (4 Pontos) Defina um sistema de equações para resolver essa análise estática. Você precisa associar uma ou mais equações a cada ponto da estrutura criada no item anterior. Veja bem: a propagação de constantes esparsa associa uma única equação com cada variável. A análise de vida das variáveis associa duas, uma para calcular o conjunto *IN*, e outra para calcular o conjunto *OUT*. Seja cuidadoso acerca de quantas equações estarão associadas a cada ponto.
- (c) (3 Pontos) Explique como uma solução do seu sistema de equações pode indicar a um professor em que direção caminhar para encontrar quem lhe avalie um projeto.

4. Esta questão refere-se à análise descrita abaixo, que lida com uma linguagem muito simples. Essa linguagem possui somente três tipos de instrução:

- Instruções vazias,  $\bullet = \bullet$
- Instruções do tipo flip.
- Instruções do tipo flop.

As equações de fluxo de dados que descrevem a interpretação abstrata de cada tipo de instrução são mostradas à esquerda da figura abaixo.

Funções de Transferência
$\ell; \bullet = \bullet$
$\text{IN}[\ell] = \text{OUT}[\ell]$
$\ell; \text{flip}$
$\text{IN}[\ell] = (\text{OUT}[\ell] \setminus \{\text{flip}\}) \cup \{\text{flip}\}$
$\ell; \text{flop}$
$\text{IN}[\ell] = (\text{OUT}[\ell] \setminus \{\text{flop}\}) \cup \{\text{flop}\}$
Operador de encontro:
$\text{OUT}[\ell] = \bigcup_{p \text{ in pred}} \text{IN}[\ell_p]$



- (a) (3 Pontos) Mostre o resultado da análise em questão para os rótulos do programa à direita da figura acima. Não é necessário mostrar os conjuntos OUT. Mostre somente os conjuntos IN associados a cada rótulo. Você pode escrevê-los abaixo, ou diretamente na figura acima.
- (b) (2 Pontos) Essa é uma análise do tipo *forward*, ou é uma análise do tipo *backward*? Justifique a sua resposta.
- (c) (1 Pontos) Essa é uma análise do tipo *may*, ou é uma análise do tipo *must*? Justifique a sua resposta.
- (d) (4 Pontos) Mostre que a função de transferência associada à instrução flip, vista na figura acima, é monotônica. Dica: pense em tabelas.