

Universidade Federal de Rio Grande do Norte
Centro de Tecnologia
Departamento de Engenharia da Computação e Automação
DCA 0201 - Paradigmas de Programação, 2012.1
Lista 1: Avaliação da 1^a unidade.

Professor: Samuel Xavier de Souza
Assistente: Kayo Gonçalves e Silva

15 de março de 2012

Instruções para resolução

- Procure responder corretamente todas as questões da lista.
- **Suas respostas serão validadas de forma oral por amostragem**—geralmente de 2 à 3 defesas orais.
- Se não conseguir responder alguma questão, **procure esclarecer as dúvidas há tempo** em sala de aula com o professor, pelo SIGAA, com um colega, ou por e-mail. Se necessário, é possível marcar um horário para tirar dúvidas na sala do professor.
- **Não serão aceitas resoluções ”mágicas”**. Sua nota nesse caso será 0 (zero).
- Procure entregar a resolução da lista de forma organizada. Isso pode favorecer a sua nota.
- Os códigos dos programas requisitados deverão aparecer no corpo da resolução da questão.
- A resolução da lista deverá ser entregue em **formato PDF em apenas 1 (um) arquivo**. Para evitar problemas com má conversão, é aconselhado que se utilize um processador de textos que suporte o formato PDF (como o OpenOffice) ou que se utilize de uma impressora de PDF (ex. pdfcreator).
- Procure **preparar sua defesa oral para cada questão**. Explicações diretas e sem rodeios favoreceram a sua nota.
- **Obedeça o prazo de entrega da lista. Se não entregar há tempo não poderá fazer a prova oral**. Isso significa que deverá entrar com um requerimento junto ao departamento para repor a prova. Sem requerimento não há reposição de prova.
- A defesa oral leva apenas de 10 a 15 minutos. As provas são feitas em sequência em horários fixados com antecedência. Por uma questão de ordem e respeito mútuo entre alunos, **não será tolerado que o aluno chegue atrasado para a sua prova**.

Prolog

1. (1 ponto) Defina um predicado `uniao(L,K,M)`, que seja verdadeiro sse M é a união de L e K sem repetições.
2. (1 ponto) Defina um predicado `unir(L,K,M)` que, dada duas listas de inteiros ordenadas L e K, seja verdadeiro sse M é uma lista ordenada contendo todos os elementos de L e K. Obs.: podem haver repetições.
3. (1 ponto) Defina um predicado `apagar(L,K,M)` que é verdadeiro sse M é o resultado de apagar o item L da lista K; o que muda se, nesse predicado, pretende-se apagar todos os itens?
4. (2 pontos) Considere a seguinte representação usando listas em Prolog para um arranjo de resistores.
 - resistores em serie são representados por suas resistências em sequência na lista; Ex.

---r1---r2---r3--- ==> [r1, r2,r3]

- resistores em paralelo são representados por uma ou mais listas consecutivas como elementos de uma lista; Ex.

---r1-----
---|-----|---r4--- ==> [[r1],[r2,r3],r4]
---r2---r3---

Defina o predicado `resitequiv(X,Y)` que calcula a resistência equivalente Y de um circuito resistivo qualquer representado pela lista X.

LISP

5. (1 ponto) Defina uma função que receba uma lista de listas de dois elementos inteiros e forneça a soma dos produtos, ou seja:
 - Entrada: ((1 2)(3 4)(5 6))
 - Saída: 44, o que corresponde a $1*2 + 3*4 + 5*6$A lista de entrada poderá ser composta por uma quantidade indefinida de dois elementos inteiros. Caso a lista de entrada seja vazia, a saída será 0 (zero).
6. (1 Ponto) Escreva um método que receba uma lista e diga se ela contém elementos repetidos. Isto é, a função retornará true ou false (nil).
7. (1 Ponto) Desenvolva uma função chamada muda-n. Essa função receberá um número n, um elemento e uma lista e substituirá o n-ésimo elemento da lista por esse elemento. O primeiro elemento da lista equivale a n igual a zero. O resultado será a lista já modificada. Caso n seja maior que o tamanho da lista, preencher com NIL as posições do vetor que não existam.
 - Ex1.: Pedir para mudar o elemento da posição n=0 para o elemento 44 da lista (1 2 3 4) resultará (44 2 3 4).
 - Ex2.: Pedir para mudar o elemento da posição n=4 para o elemento 44 da lista (1 2 3 4) resultará (1 2 3 4 44).
 - Ex3.: Pedir para mudar o elemento da posição n=5 para o elemento 44 da lista (1 2 3 4)

resultará (1 2 3 4 NIL 44).

Ex3.: Pedir para mudar o elemento da posição $n=7$ para o elemento 44 da lista (1 2 3 4) resultará (1 2 3 4 NIL NIL NIL 44).

8. (2 Pontos) Escreva o método de ordenação bubble sort recursivo. A resposta do método será uma lista ordenada. Não utilizar métodos de ordenação já pré-definidos.