



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Programação II

AD1 - 1º semestre de 2015

Questão 1 (2,5 pontos)

Construa expressões regulares em PHP para validar as sintaxes das seguintes entradas de dados (escreva apenas as expressões):

- a. Código postal (CEP)
- b. Letras em uma sequência de DNA de qualquer tamanho
- c. Endereço na internet. A expressão regular deve aceitar domínios sem o prefixo “www”, além de tratar o caso de sites seguros por certificado (https). Exemplos:
`http://www.domínio.com`
`https://www.domínio.com`
`http://domínio.com`
`http://domínio.com.br`
`http://www.domínio.qualquer.subdomínio.edu.jp`
- d. Placas de automóveis nacionais. Exemplos:
`ADF 1234`
`KXE 4567`
`DEF 0988`

Questão 2 (1,5 pontos)

Suponha um vetor armazenando dados de alunos associando a matrícula ao nome do aluno, como por exemplo:

```
$alunos = array(  
    12345 => "João da Silva",  
    12567 => "Maria Almeida",  
    12677 => "Bernardo Nóbrega",  
    13567 => "Arnaldo Albuquerque"  
);
```

Escreva uma função PHP que recebe o vetor de alunos desordenado e ordena o mesmo pelo número de matrícula usando recursão. Sua função deve subdividir o vetor em duas metades durante a recursão. Considere uma maneira eficiente de dividir o vetor durante o processo de ordenação recursiva.

Questão 3 (2,5 pontos)

Desenvolva um algoritmo (sequência de passos por escrito) e o implemente como uma função PHP para encontrar a melhor aproximação inteira da raiz quadrada de um determinado número também inteiro recebido como parâmetro de entrada. Atenção: não utilize métodos prontos do PHP de cálculo de raiz quadrada.

Exemplos:

- a melhor aproximação inteira da raiz quadrada do número 10 é 3;
- a melhor aproximação inteira da raiz quadrada do número 16 é 4;
- a melhor aproximação inteira da raiz quadrada do número 7 é 3;
- a melhor aproximação inteira da raiz quadrada do número 110 é 10;

Questão 4 (3,5 pontos)

Escreva uma função que recebe uma string e retorna **TRUE** caso hajam palavras repetidas na string ou **FALSE** caso não encontre palavras idênticas. Considere que a string pode conter qualquer número de palavras e que as palavras são separadas por caracteres de espaço ou de pontuação (tais como , , ! , ? , (,) , - , : , ; , .). Portanto, sua função deve identificar as palavras contidas na string e verificar se existem duas ou mais iguais. Considere ainda que a mesma palavra pode ser escrita usando letras maiúsculas ou minúsculas e ainda assim ser considerada a mesma palavra.