



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina Programação de Aplicações Web**  
**Profs. Cristina Nader e Daniel de Oliveira**  
**AP1 2º semestre de 2016.**

Nome: \_\_\_\_\_

—

**Questão 1. [3,0 pontos]** O sítio Hortiflix (<http://hortiflix.com.br/>) foi criado como publicidade para um famoso mercado do Rio de Janeiro. Nesse sítio nomes de filmes famosos são reformulados de forma que façam referência a alguma fruta, legume ou verdura (produto vendido pelo mercado). Por exemplo, “O Poderoso Melão” em referência ao “O Poderoso Chefão” e os “Milhons” em referência aos “Minions”. Como nem todas as pessoas conseguem entender de primeira a qual filme a propaganda se refere resolvemos pedir sua ajuda nessa tarefa. Escreva uma função em PHP chamada **comparacaoFilmes** que recebe duas *strings*. A primeira *string* recebe o nome de um filme original e a segunda a versão do sítio Hortiflix. Sua função deve analisar o quão parecido são as letras contidas em posições correspondentes nos dois títulos de filmes fornecidos e informar um valor total que indica o índice de similaridade entre os títulos. De forma a facilitar, assuma que **\*sempre\*** os títulos dos filmes têm o mesmo comprimento (*strings* de mesmo tamanho). Utilize a regra de pontuação a seguir para comparar cada letra do primeiro título com a letra correspondente no segundo título.

1. Letras Iguais – 10 pontos
2. Mesma letra, mas em um título ela é maiúscula e no outro minúscula – 5 pontos
3. Letras diferentes – 2 pontos
4. Letra em uma *string* e número ou outro símbolo na outra – 0

Exemplo:

`comparacaoFilmes("Minions", "Milhons")` -> Retorna  $10+10+2+2+10+10+10 = 54$

```

<?php
//3 alternativas que poderiam ser usadas (há outras...)
function isNumber($a){
    return preg_match('/\d/', $a);
}

function isLetter($a){
    return preg_match('/[a-zA-Z]/', $a); ;
}

function isUpperLetter($a){
    return preg_match('/[A-Z]/', $a);
}
function comparacaoFilmes($s1, $s2){

    $s1M = strtoupper($s1);
    $s2M = strtoupper($s2);
    $pontos = 0;
    // assumindo que *sempre* os títulos dos filmes têm o mesmo comprimento
    for ($i = 0; $i < strlen($s1); $i++){

        $letraS1 = $s1[$i];
        $letraS2 = $s2[$i];
        $letraS1M = $s1M[$i];
        $letraS2M = $s2M[$i];

        if($letraS1 != ' '){
            //Letras Iguais – 10 pontos
            if($letraS1 == $letraS2){
                $pontos = $pontos +10;
            }
            //Mesma letra, mas em um título ela é maiúscula e
            //no outro minúscula – 5 pontos
            else if($letraS1M == $letraS2M){
                $pontos = $pontos +5;
            }
            //Letras diferentes – 2 pontos
            else if(isLetter($letraS1) && isLetter($letraS2)){
                $pontos = $pontos +2;
            }
            //Letra em uma string e número ou outro símbolo na outra – 0
            //else nao precisa fazer nada
        }
    }
}

```

```
        return $pontos;
    }
    //testes
    echo(comparacaoFilmes('abra cadabra', 'abra cadabra')."<br>");
    echo(comparacaoFilmes('abra cadabra', 'Abra Cadabra')."<br>");
    echo(comparacaoFilmes('abra cadabra', 'abra bidabra')."<br>");
    echo(comparacaoFilmes('abra cadabra', 'abra 23dabra')."<br>");
```

**Questão 2. [4,0 pontos]** Considere a classe a seguir que representa um aluno no sistema acadêmico do CEDERJ.

```
class aluno{  
  
    public $id, $nome, $coeficienteRendimento, $anoEntrada;  
  
};
```

Onde \$id representa o identificador interno do aluno no sistema, \$nome representa o nome do aluno, \$coeficienteRendimento o coeficiente de rendimento do aluno (CR) e \$anoEntrada o ano em que o aluno foi admitido no CEDERJ via vestibular.

- 1) Escreva uma função em PHP chamada **ordenaAluno** que recebe um *array* de alunos e o ordena por ordem de CR dos alunos.
- 2) Escreva uma função em PHP chamada **mediaAlunoAno** que recebe um *array* de alunos e um ano de entrada e apresenta a média dos CRs de todos os alunos que foram admitidos nesse ano.
- 3) Declare a classe chamada **universidade** que contém um *array* de alunos e os métodos **ordenaAluno** e **mediaAlunoAno** que você definiu anteriormente.
- 4)

```
<html>
<head>
<title>Online PHP Script Execution</title>
</head>
<body>
<?php

class aluno{
    public $id, $nome,$coeficienteRendimento, $anoEntrada ;
    ;
    function exhibe(){
        $a=$this;
        $s = $a->id.' '.$a->nome.' '.$a->coeficienteRendimento.' '.$a->anoEntrada.'<br>';
        return $s;
    }
}

//testes:
$pedro = new aluno();
$pedro->id = 1;
$pedro->nome = 'Pedro';
$pedro->coeficienteRendimento = '8.0';
$pedro->anoEntrada = 2015;
$joao = new aluno();
$joao->id = 2;
$joao->nome = 'Joao';
$joao->coeficienteRendimento = '9.0';
$joao->anoEntrada = 2016;
$paulo = new aluno();
$paulo->id = 3;
$paulo->nome = 'Paulo';
$paulo->coeficienteRendimento = '7.5';
$paulo->anoEntrada = 2016;

//Declare a classe chamada universidade que contém um array de alunos
//e os métodos ordenaAluno e mediaAlunoAno que você definiu anteriormente.
class universidade{
    //recebe um array de alunos e um ano de entrada e apresenta a média dos CRs de todos os
    alunos que foram admitidos nesse ano.
    function mediaAlunoAno($als, $ano){
        $media = 0;
        $nAlunosAno = 0;
        foreach ($als as $aluno) {
            if($aluno->anoEntrada == $ano){
                $media = $media + $aluno->coeficienteRendimento;
            }
        }
    }
}
```

```

        $nAlunosAno = $nAlunosAno+1;
    }
}
if ($nAlunosAno ==0)
    return 0;
return $media/$nAlunosAno;
}

//recebe um array de alunos e o ordena por ordem de CR dos alunos.
function ordenaAlunos(&$als){
    for ($i = 0; $i < count($als)-1; $i++){
        for ($j = 1; $j< count($als); $j++){
            if($als[$i]->coeficienteRendimento > $als[$j]->coeficienteRendimento){
                $temp = $als[$i];
                $als[$i] = $als[$j];
                $als[$j] = $temp;
            }
        }
    }
    return $als;
}

public $alunos;
};
$u = new universidade();
$u->alunos= array($pedro, $joao, $paulo);

foreach ($u->alunos as $aluno){
    echo $aluno->exibe();
}

$u->ordenaAlunos($u->alunos);
foreach ($u->alunos as $aluno){
    echo $aluno->exibe();
}
echo $u->mediaAlunoAno($u->alunos, 2016);
?>
</body>
</html>

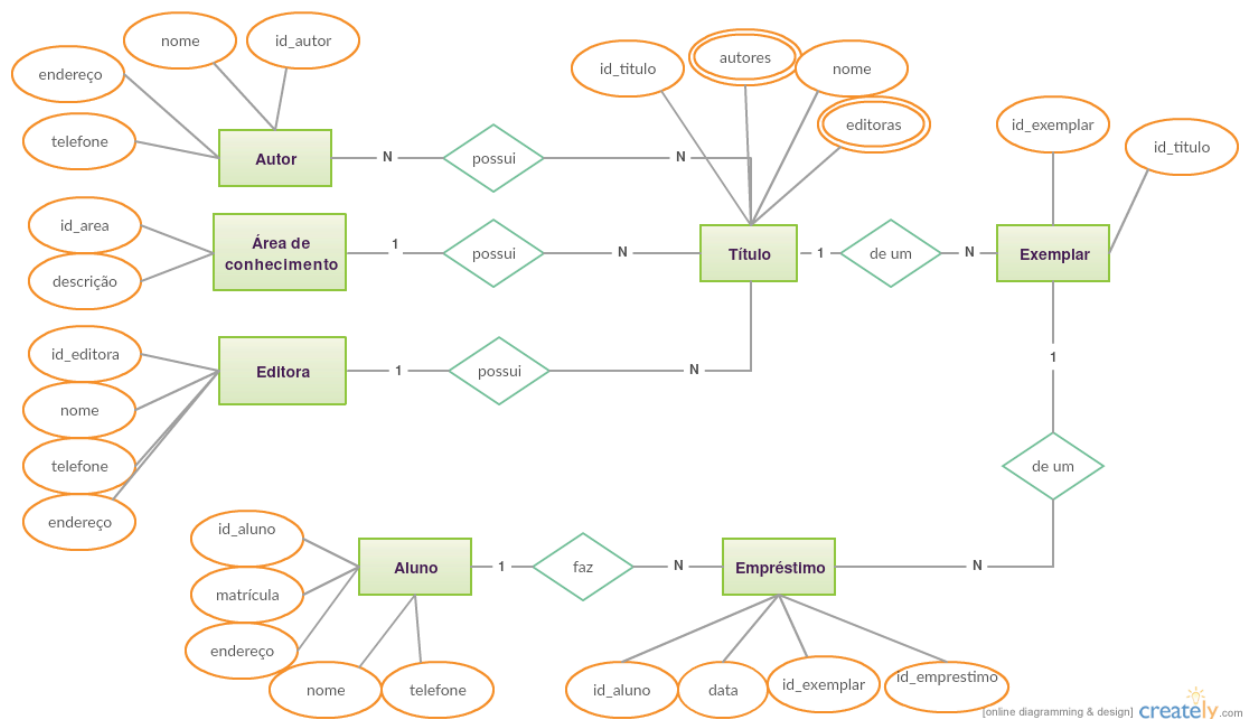
```

**Questão 3 [3,0 pontos]** Você foi contratado para modelar um sistema de controle de bibliotecas de uma grande universidade. O primeiro passo nesse desenvolvimento é criar o banco de dados da

aplicação. Desenvolva um modelo Entidade-Relacionamento (ER) que apoie as seguintes características do sistema:

- A biblioteca dispõe de livros, que são chamados de **títulos**.
- Cada título possui um nome, autores e editoras associadas.
- Cada título pertence a uma **área de conhecimento** e possui um código único de identificação.
- Cada título pode possuir vários **exemplares**.
- Cada exemplar possui um código único de identificação.
- Cada título pode ter vários **autores** e um mesmo autor pode ter escrito vários títulos.
- Um autor possui código, nome, telefone e endereço.
- As **editoras** possuem código, nome, telefone e endereço.
- As áreas de conhecimento possuem código e uma descrição.
- **Alunos** tomam livros **emprestados** por uma semana.
- A data de empréstimo é fundamental no processo.
- Cada aluno possui uma matrícula, um nome, um telefone e um endereço.

Após elaborar o modelo ER, escreva o comando SQL para criar a **tabela aluno** de acordo com as características presentes no modelo ER que você elaborou.



CREATE TABLE Aluno

```
(  
Id_aluno int primary key,  
Matricula int,  
Endereco varchar(255),  
Nome varchar(255),  
Telefone varchar(255)  
);
```