

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

Questão 1) (4.0 pontos)

Escreva um programa que receba, como parâmetro de entrada, uma frase com palavras e pontuações válidas ('.' e ',') juntas, imprima a frase com as palavras separadas e informe o número de pontuações existentes na frase fornecida. Por exemplo, se a chamada do programa for:

```
java Coloca_espaco_em_branco PROGIII,primeira,segunda,terceira.
```

A resposta será:

```
PROGIII, primeira, segunda, terceira.
```

4

RESPOSTA:

```
public class Coloca_espaco_em_branco{
    public static boolean Pertence(char c, String s){
        char[] a = s.toCharArray();
        for(int i = 0; i < s.length(); i++) if(a[i] == c) return true;
        return false;
    }

    public static String colocaEspaco(String s){
        char[] a = s.toCharArray();
        String resp = "";
        for(int i = 0; i < s.length(); i++){
            resp = resp + a[i];
            if(Pertence(a[i], ".,;")) resp += ' ';
        }
    }
}
```

```

    return resp;
}

public static void main(String[] args){
    String resp = colocaEspaco(args[0]);
    System.out.println(resp + "\n" + (resp.length() - args[0].length()));
}
}

```

Questão 2) (3.0 pontos)

Suponha o método main() abaixo, seguido do suposto resultado, o qual instancia e manipula uma URL:

Código:
<pre> public class AP1_2013_2_Q2 { public static void main(String[] args) { URL url1 = new URL("http", "www.cederj.edu.br", 8080, "extensao"); if (url1.valida()) System.out.println(url1); else System.out.println("URL mal formada!"); } } </pre>
Resultado:
<pre> http://www.cederj.edu.br:8080/extensao </pre>

Sua resposta deve fazer com que o código acima funcione.

- α) Declare a classe URL, seus campos e seu construtor. Uma url é composta pelo seu tipo (http, ftp, etc ...), seu domínio (endereço www, por exemplo), uma porta (número que indica a qual aplicação uma comunicação é endereçada) e um recurso (string com o nome de uma imagem, uma página ou o nome de algum outro arquivo).
- β) Defina um método toString(), o qual converte os dados de um objeto URL na string com o formato fornecido acima: <http://www.cederj.edu.br:8080/extensao> . Este método é importante para que a primeira chamada do método println() acima funcione.
- χ) Defina o método valida(), o qual indica se uma URL é válida. Nesta questão uma URL é válida se o protocolo é “http” ou “ftp”, o domínio não contém nenhum dos caracteres à seguir (“@”, “#” e “\$”), a porta varia entre 0 e 65535 e o tamanho máximo da URL não excede 2000 caracteres.

RESPOSTA:

```

class URL {
    String protocolo;
    String dominio;
    int porta;
    String recurso;
}

```

```

    public URL(String protocolo, String dominio, int porta, String recurso) {
        this.protocolo = protocolo;
        this.dominio = dominio;
        this.porta = porta;
        this.recurso = recurso;
    }

    public boolean valida() {
        return ((protocolo.equalsIgnoreCase("http") ||
        protocolo.equalsIgnoreCase("ftp")) && (!dominio.contains("@") || !dominio.contains("#") || !
        dominio.contains("$")) && (porta >= 0 && porta <= 65535) && this.toString().length() <
        2000);
    }

    public String toString() {
        return protocolo + "://" + dominio + ":" + porta + "/" + recurso;
    }
}

```

Questão 3) (3.0 pontos)

Suponha a classe abaixo cujos campos possuem nomes auto-explicativos:

```

class Veiculo {
    int numRodas;
    String marca;
    float peso;
    int potencia;
    int qtdLugares;
}

```

- Defina um construtor para esta classe.
- Defina um método chamada cargaUtil(), o qual retorna a soma do peso do veículo mais a soma do peso dos passageiros. Para tal, considere o peso médio de uma pessoa de 70kg.
- Defina um método que retorne a relação peso/potência de um veículo. Esta é dada dividindo a carga útil pela potência do veículo.
- Defina uma classe Passeio como um tipo especializado de Veículo. Veículos deste tipo possuem tipicamente 4 rodas e 5 lugares.
- Instancie 2 objetos da classe Passeio: um da marca VW, com peso de 1080kg e 95 cv (cavalos de potência; outro da marca Renault, com peso 1117 e 102 cv. Compare a relação peso/potência entre eles e imprima o nome do que possui a menor relação.

RESPOSTA:

```

class Veiculo {
    int numRodas;
    String marca;
    float peso;
    int potencia;
    int qtdLugares;
}

```

```

    public Veiculo(int numRodas, String marca, float peso, int potencia, int lugares) {
        this.numRodas = numRodas;
        this.marca = marca;
        this.peso = peso;
        this.potencia = potencia;
        this.qtdLugares = lugares;
    }

    public float cargaUtil() {
        return this.peso + this.qtdLugares * 70;
    }

    public float rpp() {
        return this.cargaUtil() / this.potencia;
    }
}

class Carro extends Veiculo {
    public Carro(String marca, float peso, int potencia) {
        super(4, marca, peso, potencia, 5);
    }
}

public class AP1_2013_2_Q3 {
    public static void main(String[] args) {
        Carro logan = new Carro("Logan", 1080, 95);
        Carro cobalt = new Carro("Cobalt", 1117, 102);
        System.out.println("O carro com menor relação peso/potência é o: " +
(logan.rpp() < cobalt.rpp() ? logan.marca : cobalt.marca));
    }
}

```