

Pergunta 1

2 pts

Deseja-se criar um método que, dados três valores do tipo double representando os lados de um triângulo, diga se esse triângulo é equilátero (todos os lados iguais), isósceles não equilátero (dois lados iguais e um diferente) ou escaleno (todos os lados diferentes). Assinale o código em Java que faz isso.



```
static void checkTriangulo(double l1, double l2, double l3){
    if(l1 == l2 && l1 == l3){
        System.out.println("Equilatero");
    } else {
        if(l1 == l2 || l1 == l3 || l2 == l3){
            System.out.println("Isosceles nao equilatero");
        } else {
            System.out.println("Escaleno");
        }
    }
}
```



```
static void checkTriangulo(double l1, double l2, double l3){
    if(l1 == l2 == l3){
        System.out.println("Equilatero");
    } else {
        if(l1 == l2 != l3){
            System.out.println("Isosceles nao equilatero");
        } else {
            System.out.println("Escaleno");
        }
    }
}
```



```
static void checkTriangulo(double l1, double l2, double l3){
    if(l1 == l2 || l1 == l3 || l2 == l3){
        System.out.println("Equilatero");
    }
    if(l1 == l2 && l1 == l3 && l2 != l3){
        System.out.println("Isosceles nao equilatero");
    }
    if(l1 != l2 && l1 != l3 && l2 != l3)
        System.out.println("Escaleno");
}
```



```
static void checkTriangulo(double l1, double l2, double l3){
    if(l1 == l2 && l1 == l3 && l2 == l3){
        System.out.println("Equilatero");
    }
    if(l1 == l2 || l1 == l3 || l2 != l3){
        System.out.println("Isosceles nao equilatero");
    }
    if(l1 != l2 || l1 != l3 || l2 != l3)
        System.out.println("Escaleno");
    }}

```

☐

```
static void checkTriangulo(double l1, double l2, double l3){
    if(l1 == l2 && l1 == l3){
        System.out.println("Equilatero");
    } else {
        if(l1 != l2 || l1 != l3){
            System.out.println("Isosceles nao equilatero");
        } else {
            System.out.println("Escaleno");
        }
    }
}

```

Pergunta 2

2 pts

Deseja-se criar um método que, dado um número inteiro positivo qualquer como entrada, imprima os dígitos desse número na ordem inversa. Por exemplo, se a entrada for o número 123456, deve ser impresso, na tela, 654321. Assuma que o número é positivo (não precisa testar).

☐

```
static void inversor(int i){
    while(val > 0 ){
        System.out.print(val);
        val = val - 10;
    }
}

```

☐

```
static void inversor(int i){
    while(val > 0 ){
        System.out.print(val%10);
        val = val - 10;
    }
}

```

☐

```
static void inversor(int i){
    while(val > 0 ){
        System.out.print(val/10);
        val = val/10;
    }
}
```



```
static void inversor(int i){

    while(val > 0 ){
        System.out.print(val%10);
        val = val/10;
    }
}
```



```
static void inversor(int i){
    while(val > 0){
        System.out.print(val/10);
        val = val%10;
    }
}
```

Texto referente às perguntas 3 e 4:

Em certa universidade, a aprovação dos alunos requer:

- a.** Uma frequência maior ou igual a 70%.
- b.** Uma nota igual ou maior a 5.0, que pode ser obtida:
 - I.** Na primeira avaliação (media1): equivale à média simples de três provas (P1, P2 e P3) aplicadas durante o semestre.
 - II.** Na segunda avaliação (media2): equivale à média simples entre a média da primeira avaliação (media1) e da prova de recuperação (PREC).
- c.** Importante: apenas alunos com média entre 3.0 e 5.0 na primeira avaliação têm autorização para fazer a prova de recuperação. Portanto, a nota da PREC deve ser ignorada tanto para alunos com aprovação direta (i.e. na primeira avaliação) quanto para alunos que não atingirem a nota mínima de 3.0 na primeira avaliação. Deseja-se implementar um programa que seja capaz de dizer se o aluno foi aprovado ou reprovado com base em suas notas e frequências. Para isso, foi desenvolvido o código (incompleto) a seguir. Responda às questões abaixo:

```
public class Avaliacao {

    //Possíveis resultados
    final static int APROVADO_DIRETO = 0; //Aprovado na P1,P2,P3
    final static int APROVADO_2AVALIACAO = 1; //Aprovado após PREC
    final static int REPROVADO_FREQUENCIA = 2; //Reprovado por frequência
```

```

final static int REPROVADO_DIRETO = 3; //Não autorizado a fazer REC
final static int REPROVADO_2AVALIACAO = 4; //Reprovado após fazer REC

static int avaliar (double P1, double P2, double P3, double PREC, int freq){
    double mediaProvas = (P1+P2+P3)/3;
    double mediaRec = (mediaProvas + PREC)/2;
    //RESPOSTA DA PERGUNTA 3
}

public static void main(String[] args) {
    int resultado = avaliar(4.0, 5.0, 6.0, 0.0, 70);

    //RESPOSTA DA PERGUNTA 4

} //fim do main
} // fim da classe

```

Pergunta 3

2 pts

Complete o método “avaliar”, que dá o resultado da avaliação com base nas notas e frequências passadas como parâmetro. O resultado é um valor entre 0 e 4, definido pelas constantes no início da classe.

```

if(mediaProvas >= 5.0){
    return APROVADO_DIRETO;
} else {
    if (mediaRec >= 5.0){
        return APROVADO_2AVALIACAO;
    } else {
        return REPROVADO_2AVALIACAO;
    }
}
if(mediaProvas < 3.0){
    return REPROVADO_DIRETO;
} else {
    return APROVADO_2AVALIACAO;
}

if(freq < 70){
    return REPROVADO_FREQUENCIA;
}

```

```
if(mediaProvas >= 5.0 || freq >= 70){
    return APROVADO_DIRETO;
} else {
    if (mediaRec >= 5.0){
        return APROVADO_2AVALIACAO;
    } else {
        return REPROVADO_2AVALIACAO;
    }
    if(mediaProvas < 3.0 && mediaRec < 5.0){
        return REPROVADO_DIRETO;
    } else {
        return REPROVADO_2AVALIACAO;
    }
}

if (freq < 70){
    return REPROVADO_FREQUENCIA;
}
```

☐

```
if(freq < 70){
    return REPROVADO_FREQUENCIA;
}

if(mediaProvas >= 5.0 && mediaRec >= 5.0){
    return APROVADO_DIRETO;
} else {
    if(mediaProvas < 3.0 && mediaRec < 5.0){
        return REPROVADO_DIRETO;
    } else {
        return APROVADO_2AVALIACAO;
    }
}

return REPROVADO_2AVALIACAO;
```

☒

```
if(freq < 70){
    return REPROVADO_FREQUENCIA;
}

if(mediaProvas >= 5.0){
    return APROVADO_DIRETO;
} else {
    if(mediaProvas < 3.0){
        return REPROVADO_DIRETO;
    } else {
        if (mediaRec >= 5.0){
            return APROVADO_2AVALIACAO;
        } else {
            return REPROVADO_2AVALIACAO;
        }
    }
}

}
```

☐

```
if(mediaProvas >= 5.0 && freq > 70){
    return APROVADO_DIRETO;
} else {
    if(mediaRec <= 3.0){
        return REPROVADO_DIRETO;
    } else {
        if (mediaRec < 5.0){
            return APROVADO_2AVALIACAO;
        } else {
            return REPROVADO_2AVALIACAO;
        }
    }
}
```

Pergunta 4

2 pts

Complete o método “main” para que ele imprima na tela o resultado da avaliação do aluno com base no valor de “resultado”.

☐

```
while(resultado >= 0){
    System.out.println("O resultado eh: " + resultado);
    resultado = resultado - 1;
}
```

☐

```
switch(resultado){
    case APROVADO_DIRETO:
        System.out.println("Aprovado direto.");
    case APROVADO_2AVALIACAO:
        System.out.println("Aprovado na 2a avaliacao.");
    case REPROVADO_FREQUENCIA:
        System.out.println("Reprovado por frequencia.");
    case REPROVADO_DIRETO:
        System.out.println("Reprovado sem direito a recuperacao.");
    case REPROVADO_2AVALIACAO:
        System.out.println("Reprovado apos a recuperacao.");
    default:
        System.out.println("Erro! ");
}
```



```
switch(resultado){
    case APROVADO_DIRETO:
        System.out.println("Aprovado direto.");
        break;
    case APROVADO_2AVALIACAO:
        System.out.println("Aprovado na 2a avaliacao.");
        break;
    case REPROVADO_FREQUENCIA:
        System.out.println("Reprovado por frequencia.");
        break;
    case REPROVADO_DIRETO:
        System.out.println("Reprovado sem direito a recuperacao.");
        break;
    case REPROVADO_2AVALIACAO:
        System.out.println("Reprovado apos a recuperacao.");
        break;
}
```

```
int auxiliar = 0;
while(auxiliar <= 4){
    if (auxiliar == resultado){
        System.out.println("Aprovado: " + auxiliar);
    } else {
        System.out.println("Reprovado: " + auxiliar);
    }
    auxiliar = auxiliar + 1;
}
```

```
while (resultado <= 4)
    switch(resultado){
        case APROVADO_DIRETO:
            System.out.println("Aprovado direto.");
            break;
        case APROVADO_2AVALIACAO:
            System.out.println("Aprovado na 2a avaliacao.");
            break;
        case REPROVADO_FREQUENCIA:
            System.out.println("Reprovado por frequencia.");
            break;
        case REPROVADO_DIRETO:
            System.out.println("Reprovado sem direito a recuperacao.");
            break;
        case REPROVADO_2AVALIACAO:
            System.out.println("Reprovado apos a recuperacao.");
        }
    resultado = resultado + 1;
}
```

Pergunta 5**2 pts**

Deseja-se implementar um método que imprima a tabuada de um número inteiro positivo n qualquer passado como entrada, ou seja, imprima o valor da multiplicação de n por todos os números entre 1 e 10. Assinale a alternativa que indica uma implementação correta desse método.



```
static void tabuada(int n){
    int i = 10;
    while (i <= 10){
        System.out.println(n + " x " + i + " = \t" + n*i);
        i = n + 1;
    }
}
```



```
static void tabuada(int n){
    int i = n%10;
    while (i <= 10){
        System.out.println(n + " x " + i + " = \t" + n*i);
        i = i - 1;
    }
}
```



```
static void tabuada(int n){
    while (n >= 1){
        System.out.println(n + " x " + n + " = \t" + n*n);
        n = n - 1;
    }
}
```



```
static void tabuada(int n){
    int i = 1;
    while (i <= 10){
        System.out.println(n + " x " + i + " = \t" + n*i);
        i = i + 1;
    }
}
```




```
static void tabuada(int n){  
    int i = n;  
    while (i <= 10){  
        System.out.println(n + " x " + i + " = \t" + n*i);  
        i = n + 1;  
    }  
}
```

Salvo em 12:33

Enviar teste