**PROVA RECUPERACAO - INDIVIDUAL**

**ALGORITMOS EM JAVA**

**Aluno:**

**Data:**

**Curso:** Jovem Programador – Programador de Sistemas JAVA

**Professor:** Gelvazio Camargo

Elabore os algoritmos em **JAVA** e entregue numa pasta de nome “**recuperacaoprova**” no seu github.

Cada Exercício deve estar num arquivo único contendo como nome o exercício desta forma “exercício\_1\_1\_1.java” para o exercício 1.1.1.

Formação da Nota da Prova de Recuperação

Tabela

Descrição gerada automaticamente

**1 – SEQUENCIAIS**

## 1.1. – SAÍDA SIMPLES - 3 QUESTOES – PESO = 0.10

1.1.1 **- Frase na tela -** Implemente um programa que escreve na tela a frase "O primeiro programa a gente nunca esquece!".

1.1.2 **- Etiqueta -** Elabore um programa que, após limpar a tela, escreve seu nome completo na primeira linha, seu endereço na segunda, e o CEP e telefone na terceira.

1.1.3 **- Frases positivas -** Faça um programa que mostre na tela as frases motivadoras abaixo, que são aquelas que fazem com que muitas ideias sejam levadas adiante e amadureçam ou seja aprofundada.

"Eu vou conseguir!"

"Isso vai funcionar!"

"Se eu continuar tentando, eu vou vencer!"

"Quem estuda, no fim sempre vence!"

## 1.2. MATEMÁTICA - 8 QUESTOES – PESO = 0.10

1.2.1 **- Tempo livre -** Um estudante muito metódico estava matriculado em 6 disciplinas, e dispunha de 1 hora e 20 minutos para estudar. Sua intenção era dividir o tempo disponível igualmente para as 6 disciplinas, e descansar livremente o tempo restante. Faça um programa que calcule o tempo que ele deve dedicar para cada disciplina e o tempo livre.

1.2.2 **- Média de notas -** Monte uma expressão matemática que calcula a média de 4 notas de um período. Faça o cálculo através de um programa, mostrando na tela o resultado.

1.2.3 **- Conversão de temperatura -** Faça um programa que calcula e mostra uma tabela de graus

Celsius/Fahrenheit de 1 a 10 [fórmula: C = 5/9(F-32)]. Por enquanto (sem comandos de repetição), você deverá escrever as instruções para calcular e mostrar cada resultado.

1.2.4 **- Imposto -** Um imposto é calculado com base na seguinte tabela:

|  |  |  |
| --- | --- | --- |
| Até | 1.200,00 | isento |
| de | 1.201,00 a 5.000,00 | 10% |
| de | 5.001,00 a 10.000,00 | 15% |
| acima de | 10.000,00 | 20%. |

Implemente um programa que calcule os impostos a pagar para um valor em cada faixa. Para cada um, mostre uma mensagem que identifique na tela a que se refere cada valor.

1.2.5 **- Tempo dedicado -** Uma pessoa com pouco tempo disponível lê um livro por 15 minutos a cada dia, 5 dias por semana. Monte a fórmula e escreva um programa que calcula e mostra na tela quanto tempo, em horas, a pessoa terá dedicado ao livro ao final de um ano.

1.2.6 **- Cálculo de notas -** Um professor atribui pesos de 1 a 4 para as notas de quatro avaliações. A nota é calculada por meio da média ponderada (N1 + N2\*2 + N3\*3 + N4\*4)/10, onde N1 é a nota da primeira avaliação, N2 a da segunda, etc. Um aluno tirou as seguintes notas: 8 - 7,5 - 10 - 9. Faça um programa que calcula e mostra as notas e a média deste aluno.

1.2.7 **- Funções aninhadas -** Escrever um programa que calcula a raiz de 3, arredonda o resultado e calcula a exponencial do valor resultante. Para expoente use a função **JAVA** Math.pow (base, expoente) e para raiz quadrada use a função JAVA Math.sqrt (base, expoente).

1.2.8 **- Devagar se vai ao longe -** Vou e volto diariamente a pé para o trabalho, que dista aproximadamente 600 m de minha casa. Supondo que trabalho 4 dias por semana, 40 semanas por ano, "bole" a operação matemática que deve ser efetuada para calcular quantos quilômetros, aproximadamente, terei andado ao final de um ano. Elabore um programa que faça as contas e mostre o resultado na tela.

**2.VARIÁVEIS E ENTRADA DE DADOS**

2.1. SAÍDA SIMPLES - 3 QUESTOES – PESO = 0.10

2.1.1 **- Mensagem -** Implemente um programa que leia três linhas de mensagens de até 15 caracteres cada uma e mostra-as na tela uma frase por linha.

2.1.2 **- Etiqueta -** Escreva um programa que lê do teclado seu nome completo, endereço, CEP e telefone, e mostra seu nome na primeira linha, seu endereço na segunda, e o CEP e telefone na terceira.

2.1.3 **- Triângulo com iniciais -** Escrever um programa que lê um caractere, as iniciais de um nome (3 caracteres), uma linha e uma coluna e depois desenha na tela um triângulo equilátero formado com o caractere, tendo dentro as iniciais lidas. O caractere no ápice do triângulo deve estar na linha e coluna lidas, e a altura do triângulo deve ser no máximo 5 linhas.

### 

### 2.2. MATEMÁTICA – 2 QUESTOES – PESO = 0.10 e 0.20

2.2.1 – Peso – 0.10 - **Média aritmética** - Escrever programa que lê três notas inteiras e calcula a sua média aritmética.

2.2.2 – Peso – 0.10 - **Média geométrica** - Elabore um programa que lê três valores e calcula a média geométrica dos números lidos (divisão do produto pela quantidade de valores).

2.2.3 – Peso – 0.10 - **Média ponderada** - Implemente um programa que lê três valores e calcule a média ponderada para pesos 1, 2 e 3, respectivamente (multiplique cada nota pelo seu peso, some os produtos e divida o resultado pela soma dos pesos).

2.2.4 – Peso – 0.10 - **Aritmética básica** - Implemente um programa que lê dois números quaisquer e informa sua soma, diferença, produto e quociente.

2.2.5 – Peso – 0.20 - **Funções matemáticas** - Elabore um programa que lê um número (suponha que será positivo) e informa seu quadrado, raiz e exponencial. Para expoente use a função JAVA Math.pow(base, expoente) e para raiz quadrada use a função JAVA Math.sqrt (base, expoente).

2.2.6 – Peso – 0.20 - **Nota final** - O critério de notas de uma faculdade consiste de uma nota de 0 a 10 em cada bimestre, sendo a primeira nota peso 2 e a segunda peso 3. Elabore um programa que lê as notas bimestrais e calcula a nota do semestre.

3.ALTERNATIVAS E DECISÃO

### 3.1. SAÍDA SIMPLES – 3 QUESTOES – PESO = 0.5

3.1.1 **- Menu principal -** Faça um programa de menu que mostra na tela, sob o título de "Menu

Principal", três opções: "1 - Fim", "2 - Cadastro" e "3 - Consulta", lê do teclado a opção desejada pelo usuário e mostra uma mensagem confirmando a opção escolhida ou uma mensagem de erro, se a opção for inválida.

3.1.2 **- Múltipla escolha 1 -** Elaborar uma questão de múltipla escolha, de uma disciplina que esteja cursando ou um tema de interesse, com um enunciado e cinco alternativas, sendo uma correta ou incorreta. Escrever um programa que mostra a questão na tela, pede a resposta correta e informa ao usuário se este acertou ou errou.

3.1.3 **- Múltipla escolha 2 -** Enriqueça o programa acima da questão de múltipla escolha, incluindo uma outra questão de outro tema. No início do programa, ofereça ao usuário a escolha de qual questão quer responder.

### 3.2. MATEMÁTICA – 8 QUESTOES – PESO = 0.20

3.2.1 **- Maior de 2 -** Elaborar programa que lê dois números quaisquer e mostra na tela uma mensagem indicando qual é o maior, ou se são iguais.

3.2.2 **- Maior de 3 -** Faça um programa que lê três números diferentes e mostra na tela uma mensagem indicando qual é o maior.

3.2.3 **- Divisão -** Escrever um programa que lê dois números e efetua uma divisão, mas somente se o divisor for diferente de zero; quando isto ocorrer, é mostrada uma mensagem de erro apropriada.

3.2.4 **- Aprovação 1 -** Elaborar programa que lê uma disciplina e respectiva nota final, múltipla de 0,5, e informa o que ocorreu. Se a nota for de 5 a 10, aprovado; se 4 ou 4,5, segunda época e, caso contrário, reprovado.

3.2.5 **- Aprovação 2 -** Altere o programa acima para que, se a nota estiver fora da faixa válida, seja emitida uma mensagem de erro.

3.2.6 **- Aprovação 3 -** Altere o programa acima para que leia também a quantidade de aulas ministradas e a quantidade de faltas do aluno. Se o aluno não obteve 75% de frequência, ele está reprovado, independentemente da nota.

3.2.7 **- Equação do segundo grau -** Elaborar programa que lê os coeficientes a, b e c de uma equação de segundo grau e, antes de calcular as raízes, calcula o delta. Se este for negativo, informa que a equação não tem solução real. Se for zero, mostra a única raiz. Se positivo, mostra as duas raízes.

3.2.8 **- Tipo de triângulo -** Em um triângulo, cada lado é menor do que a soma dos outros dois.

Escreva um programa que lê três valores e informa se estes não podem constituir um triângulo ou, caso contrário, se o triângulo formado é equilátero (três lados iguais), isósceles (dois lados iguais) ou escaleno (lados diferentes).

4.REPETIÇÃO

### 4.1. SAÍDA SIMPLES – 2 QUESTOES – PESO = 0.20

4.1.1 **- Egocentrismo 1 -** Implemente um programa que mostra seu nome na tela 10 vezes.

4.1.2 **- Egocentrismo 2 -** Implemente um programa que mostra seu nome e sobrenome na tela 15 vezes.

### 4.2. MATEMÁTICA – 15 QUESTOES – PESO = 0,40

4.2.1 **- Aprovação -** Elaborar programa que lê uma disciplina e respectiva nota (de 0 a 10, com uma casa decimal), e informa se o aluno passou na disciplina, repetindo o ciclo até que a nota lida seja zero. O aluno passa quando tira 7 ou mais.

4.2.2 **- Raiz quadrada 1 -** Implemente um programa que repetidamente calcula e mostra a raiz quadrada de 4 até 100.

4.2.3 **- Raiz quadrada 2 -** Altere o programa acima para que ele verifique se o usuário entrou um valor positivo ou zero. Se sim, a raiz é calculada, caso contrário é mostrada uma mensagem de erro.

4.2.4 **- Idade média -** Um professor, após obter informações de uma turma, deseja saber a média de idade. Escrever um programa que lê as idades até que a idade lida seja zero, quando então é mostrada a média (o zero não é considerado para a média).

4.2.5 **- Estatística de notas -** Faça um programa que lê uma quantidade qualquer de notas de 0 a 10

(não permitir fora desta faixa) e, ao final, mostra quantas notas foram digitadas, a média e a quantidade com valor abaixo de 5.

4.2.6 **- Maior -** Escrever um programa que lê números inteiros até que o número lido seja zero, quando então é mostrado o maior número lido.

4.2.7 **- Maior e menor -** Alterar o programa anterior para que mostre também o menor número lido.

4.2.8 **- Números inteiros 1 -** Escrever um programa que lê um número inteiro e mostra na tela os números inteiros de 1 até o número lido.

4.2.9 **- Números inteiros 2 -** Alterar o programa acima de forma que seja lido também o número inicial.

4.2.10 **- Soma de pares -** Implemente um programa que calcula a soma dos números pares compreendidos entre dois números lidos.

4.2.11 **- Ímpares múltiplos 1 -** Escreva um programa que soma todos os números ímpares múltiplos de três situados na faixa de 1 a 1000.

4.2.12 **- Ímpares múltiplos 2 -** Altere o programa acima de forma que a faixa seja informada pelo usuário, e os números ímpares múltiplos de três sejam mostrados em ordem decrescente.

4.2.13 **- Tabuada -** Elabore um programa que lê um número de 1 a 9 e mostra a tabuada de multiplicação do número. Por exemplo, para o 5:

5 x 1 = 5

5 x 2 = 10

...

5 x 10 = 50

Após mostrar uma tabuada, o programa pergunta se o usuário deseja ver outra. Se a resposta for positiva (por exemplo, 'S'), ele faz tudo de novo, caso contrário termina.

4.2.14 **- Numerador -** Implemente um programa que mostre na tela os números inteiros entre dois números lidos do teclado, organizados em 10 linhas e 10 colunas:

21 22 23 24 25 26 27 28 29 30

31 32 33 34...

4.2.15 **- Eleição -** Para computar o vencedor de uma eleição deve ser feito um programa. Há 3 candidatos, e os votos dos eleitores foram codificados da seguinte forma:

1, 2 ou 3: votos para os respectivos candidatos

0: voto em branco

4: voto nulo

Escrever o programa, que deve fornecer o número do vencedor da eleição (suponha que não pode haver empates), as quantidades de votos brancos e nulos e o número de eleitores que compareceram às urnas.