Seja respeitoso com o código que cria, siga as diretrizes abaixo:

* **ENTREGA MANUSCRITADOS 11 EXERCÍCIOS PARA OS DE RECUPERAÇÃO, PARA OS DEMAIS ALUNOS ENTREGA DO LINK DO REPOSITÓRIO E 6 ATIVIDADES MANUSCRITA DE SUA ESCOLHA.**
* Usar capa modelo, alterar nome, disciplina, nome da atividade e ano (- 5 pontos)
* O documento deve ter **margem superior** e **esquerda** de 3cm, **margem inferior** e **direita** de 2cm. **Alinhamento** Justificado, **Deslocamento** mudar para **Primeira Linha** de 1,25cm, e **espaçamento entre linhas** de 1,5cm
* Variáveis sempre iniciam com letra minúscula “**nota” (- 25 % na questão)**
* Variáveis com dois nomes o segundo inicia com maiúsculo “**notaBimestraUm” (- 25 % questão)**
* Para cada atividade um arquivo **“atvUm.ts” (- 5% questão) e uma cancelada**
* Atribua nomes que façam sentido para as variáveis, se vai armazenar idade o nome da variável é **“idade”** ou **“age”. (- 5% questão)**
* Escolha o tipo correto para variável **string, number** ou **boolean (- 25 % questão)**
* Variáveis, nome de arquivos e diretórios não pode ter espaço entre os nomes e acentuação (não pode acentuar variáveis, arquivos, diretórios ...) **(-50% questão)**
* **Essas regras são básicas na programação fazem parte de boas práticas, violação dessas práticas a nota é penalizada.**
* Mantenha o código endentado **(- 10 pontos da lista de exercícios)**
* Toda atividade tem que ter o enunciado e nome do aluno completo comentado, conforme exemplo atividade 1 **(- 10% questão)**
* **Os descontos por questão ficam limitado ao máximo de 45 pontos**
* **O professor pode perguntar sobre alguma questão (ou todas) a fim de confirmar o aprendizado, caso não se confirme é zerada a questão.**

/\* (5 pontos)

1 - Crie um programa que peça um número ao usuário e armazene ele na variável x. Depois peça outro número e armazene na variável y.

Mostre esses números. Em seguida, faça com que x passe a ter o valor de y, e que y passe a ter o valor de x e mostre os números.

Nome Aluno:

\*/

/\* (5 pontos)

2 - Escreva um programa que receba três números inteiros e mostre eles em ordem decrescente.

Nome Aluno:

\*/

/\*10 pontos

3 - Faça um algoritmo que leia um valor e informe se ele está dentro de

um intervalo específico (por exemplo, entre 10 e 20, 20 e 30, até o 100).

Nome Aluno:

\*/

/\*10 pontos

4 - Desenvolva um programa que calcule a média de uma sequência de números informados pelo usuário. O programa deve continuar solicitando números até que o usuário indique que não deseja mais continuar. Ao final, exiba a média dos valores digitados.

Nome aluno:

\*/

/\*10 pontos

5 - Leia um conjunto de 5 notas e mostre a maior, a menor e a média.

Nome Aluno

\*/

/\*10 pontos

6 - Leia 10 números inteiros e informe, ao final:

* A quantidade de números pares digitados e quais foram eles;
* A quantidade de números ímpares digitados e quais foram eles.

Nome aluno:

\*/

/\*10 pontos

7 - Desenvolva um programa que leia números até que a soma dos valores

digitados seja maior que 100.

Nome aluno:

\*/

/\* 10 pontos

8 - Leia a idade e o sexo de 50 pessoas e mostre a média de idade para

cada sexo.

Nome aluno:

\*/

/\* 10 pontos

9 – Pedir ao usuário dois números e imprimir nesse intervalo os números ímpares.

Nome aluno:

\*/

/\*10 pontos

10 – Crie um programa que receba os dados de dez pessoas e calcule a média de altura das pessoas maiores de idade.

Nome aluno:

\*/

/\*10 pontos

11 – Simule a compra de um carro com as seguintes condições:

Exiba no prompt uma lista com 4 modelos de carros, cada um com um valor fixo.

O usuário deve escolher um dos carros disponíveis e selecionar a forma de pagamento:

* À vista: aplicar um desconto de 15% sobre o valor do carro.
* Parcelado: aplicar um acréscimo de 13% sobre o valor do carro.

Ao final, exiba o valor final da compra de acordo com a escolha do usuário.

Nome Aluno:

\*/