

AULA PRÁTICA REPETIÇÃO

1. Faça um programa que pergunte a matrícula a nota1 e a nota2 dos alunos de uma turma. Mostre a média aritmética e se o alunos está aprovado (média ≥ 6) ou não. A entrada de dados é finalizada quando uma matrícula ≤ 0 for digitado.
2. Mostrar a tabuada de números informados pelo usuário. A entrada de dados é finalizada quando um número ≤ 0 for digitado.
3. Faça um programa que, a partir das temperaturas mínima e máxima ocorridas, mostre a temperatura média de n(lido) dias. As temperaturas devem ser geradas automaticamente com valores entre -10 e +45
Faça uma função que receba uma temperatura e retorne True se for atípico ou False caso contrário
4. Faça um programa que, a partir das temperaturas mínima e máxima ocorridas, mostre a temperatura média de vários dias. As temperaturas devem ser geradas automaticamente com valores entre -10 e +45. O término do programa ocorre, quando a diferença entre as temperaturas mínima e máxima for inferior a 10. As temperaturas devem ser geradas automaticamente com valores entre -10 e +45 Faça uma função que receba uma temperatura e retorne True se for atípico ou False caso contrário
5. Construa o seguinte jogo usuário x computador: o objetivo de cada rodada é obter mais pontos que o computador.
Inicialmente o programa deve perguntar a quantidade de rodadas.
Em cada rodada:
 - I. Um valor entre 1 e 50 é sorteado para o computador e exibido ao usuário (pontos do computador)
 - II. jogador recebe inicialmente a mesma quantidade de números (cujos valores variam entre 1 e 5) que a rodada, isto é, na 1ª rodada, recebe um número, na 2ª rodada, recebe dois números, na 3ª rodada recebe 3 números e assim sucessivamente.
 - III. O programa deve exibir os números iniciais gerados e a soma destes números, que representa o total de pontos do jogador.
 - IV. A seguir, o jogador pode fazer solicitações de n números (término das solicitações: $n==0$ ou ter feito o dobro da rodada de solicitações). A cada solicitação, o programa mostra os n valores gerados nesta solicitação e a soma destes novos valores é somada ao total de pontos. No entanto, se um dos valores gerados for igual a 1/8 do total de pontos, o jogador “perde” 20 pontos do seu total de pontos
 - V. O jogador perde a rodada caso seu total de pontos “estourar” 50 pontos ou for inferior à pontuação do computador

O jogo é vencido por quem tiver a maior quantidade de rodadas. Caso haja empate, ganha que tem a maior soma total de pontos (soma de pontos de todas as rodadas)
6. O MICODONALDS fez um acordo de compra com vários fabricantes de brinquedos para pacotes surpresas do MicoLanche Dourado. Os fabricantes entregam os brinquedos embalados em caixas de formato retangular de tamanhos variados. Para chamar mais atenção das crianças, o MICODONALDS pretende re-embalar estas caixas em esferas coloridas de plástico. As esferas são fornecidas com quatro diâmetros diferentes: 10, 15, 20 e 25cm.
O diâmetro da esfera capaz de armazenar uma determinada caixa retangular é determinado em função da diagonal desta caixa, pois é sua maior medida. A diagonal de um caixa retangular é calculada por: $D = \sqrt{A^2 + B^2 + C^2}$ onde A, B e C são as dimensões da caixa.
Para encomendar as esferas, o MICODONALDS precisa saber quantas esferas de cada diâmetro ele necessita. Portanto, faça um programa, modularizado, para obter do teclado o código e as dimensões dos brinquedos e exibir, para cada brinquedo, o diâmetro da esfera que o embalará. No final, devem ser exibidas as quantidades totais de cada um dos tipos de embalagem. Término da entrada de dados código negativo
Seu programa será considerado modularizado se tiver pelo menos 2 funções além da principal

Extras:

7. Uma empresa de eventos, que paga R\$30,00 por hora, contratou pessoas de duas formas:

- I. FIXO: 20 dias no mês e em cada dia x horas. Por ex. João foi contratado para trabalhar 10 dias 4 horas por dia e Pedro para trabalhar 20 dias, 8 horas por dia
- II. VARIÁVEL: n dias e em cada dia uma quantidade variável de horas. Por ex., Maria foi contratada para trabalhar 4 dias, no 1º dia, 2 horas, no 2º dia, 5 horas, no 3º, 3 horas e no 4º 1 hora. Zé foi contratado para trabalhar 2 dias, no 1º dia, 12 horas e no 2º dia, 10 horas

Qualquer contratado pode tirar vales durante o período que está trabalhando.

Faça um programa, utilizando a função **totaliza_horas_variaveis**, para calcular o **salário que deve ser pago** às pessoas contratadas de acordo com a seguinte fórmula:

Salário a pagar= (número total de horas trabalhadas * 30) – (total de vales pagos)

Considere que:

- Para cada contratado é perguntado seu nome e o tipo de contrato (1 ou 2), a quantidade de vales retirados e o valor de cada vale.
- A seguir, é perguntado:
 - para os fixos (tipo 1): a quantidade de dias trabalhados (n) e a quantidade de horas em cada dia (x)
 - para os variáveis (tipo 2): a quantidade de dias trabalhados (n) e para cada dia a quantidade de horas em cada dia.

No final, seu programa deve mostrar o nome do contratado com maior salário.

Término da entrada de dados: nome vazio

Obs: Considere que todos os dados digitados estão corretos.

Função **totaliza_horas_variaveis**: calcula a soma das horas trabalhadas pelos contratos do tipo 2. A função deve receber como parâmetro um número inteiro com a quantidade de dias trabalhados. Esta função lê do teclado a quantidade de horas trabalhadas em cada dia e retorna a soma das horas trabalhadas.

8. No jogo de Pedra, Papel, Tesoura, os jogadores devem simultaneamente esticar a mão, na qual cada um formou um símbolo (que significa pedra, papel ou tesoura). Então, os jogadores comparam os símbolos para decidir quem ganhou, da seguinte forma:

- Pedra ganha da tesoura (amassando-a ou quebrando-a).
- Tesoura ganha do papel (cortando-o).
- Papel ganha da pedra (embrulhando-a).

A pedra é simbolizada por um punho fechado; a tesoura, por dois dedos esticados; e o papel, pela mão aberta. Caso dois jogadores façam o mesmo gesto, ocorre um empate, e geralmente se joga de novo até desempatar.

Faça um programa que implemente o jogo com n(lido) partidas entre uma pessoa e o computador. O jogador digita 1 para Pedra, 2 para Tesoura e 3 para Papel.

9. Escreva um programa que imprime na tela um menu simulando as seguintes operações em um caixa eletrônico de um banco:

"Digite a opcao da operacao que deseja fazer:

1 para ver o saldo atual da conta,

- 2 para fazer um depósito;
- 3 para fazer um saque.”
- 4 para finalizar

Inicialmente, seu programa deve solicitar ao usuário como entrada, via teclado, sua conta e o saldo inicial da conta. Após o saldo fornecido, o programa deve imprimir o menu acima e executar a opção selecionada até a opção 4 ser digitada

Após o término da execução de cada operação de depósito e saque, o saldo atual deve ser impresso.

Obrigatoriamente, seu programa deve definir uma função para o menu.

resp_menu: A função mostra as opções do menu, captura e valida a opção escolhida, retornando-a

MODIFIQUE seu programa para processar vários clientes. Término: conta == 0

10. Faça um programa em C que obtenha a velocidade máxima permitida em uma avenida e a velocidade que cada motorista infrator estava dirigindo. Calcule e mostre o valor de cada multa recebida. A multa é calculada do seguinte modo:

- 50 reais se o motorista ultrapassou a velocidade limite em até 10km/h (ex.:vel max: 50km/h; motorista a 60km/h ou a 56km/h);
- 100 reais, se o infrator ultrapassar 11 a 30 km/h a veloc. permitida
- 200 reais, se estiver acima de 31km/h da velocidade permitida fora do horário comercial
- 400 reais, se estiver acima de 31km/h da velocidade permitida no horário comercial mais 2,00 real por km excedido.

Por exemplo, vel max: 50km/h; motorista a 120km/h às 11hs.

Multa: $400,00 + 70 * 2,00$

A hora em que ocorreu a multa deve ser perguntada qdo necessária.

A entrada de dados é finalizada ao ser digitado uma velocidade ≤ 0

a) Faça uma função que receba um valor de hora válido e retorne 1 se esta hora for dentro do horário comercial (8 às 18) , ou 0 caso contrário.

b) Faça uma função que receba a velocidade máxima e a velocidade do motorista, retornando o valor da multa. Esta função deve chamar a função anterior

11. Faça um programa que controle o saldo bancário de um cliente. O programa lê o valor do saldo anterior e em seguida lê as operações realizadas na conta. As operações podem ser as seguintes:

Saque em dinheiro (código 10);

Depósito (código 33);

Pagamento de cheque (código 4).

O programa lê o código das operações e o valor, realiza as atualizações na conta, imprimindo uma mensagem ao usuário caso seu saldo se torne negativo.

O programa deve continuar a leitura até que o código da operação seja zero. Códigos diferentes dos definidos devem ser ignorados.

Desafio: Ao final do processamento o programa deve imprimir o saldo atual do cliente, a média de saques e o valor do maior pagamento.