1. Faça um programa que, utilizando a função ***randint*** para gerar valores entre -100 e 100, mostre:
   1. Para cada número, se ele é positivo, negativo ou zero
   2. Para os números múltiplos de 3 e 4, mostrar todos os números pares entre 0 e ele
   3. A quantidade de zeros gerada
   4. A soma dos números pares
   5. O menor número negativo gerado
   6. O maior número positivo gerado
   7. A média dos números

Término da entrada de valores: quando o valor gerado for 5

A função *randint(valo inicial, valor final)*, do módulo random, escolhe um número aleatoriamente entre valor inicial e final.

1. Faça uma função que receba um número n, inteiro e positivo, e retorne True se este número é primo (isto é, só é divisível por 1 e por ele mesmo) ou False, caso contrário.
2. (D/I) Faça uma função que receba dois valores a e b (inteiros e positivos) e mostre todos os números primos no intervalo formado por eles

Dica: o teste de divisibilidade pode limitar-se até o √n

1. (I) A taxa de crescimento de uma população de bactérias é de 0,3% por hora. No entanto, por motivos desconhecidos, a cada 3 horas, 0,5% da população nascida na última hora, morre. Faça uma função que receba a população inicial e retorne em quantas horas a população de bactérias irá dobrar.
2. A promoção TadeIdadeNova distribui os seguintes prêmios:

* Para os aniversariantes do dia (mesmo dia e mês da promoção): R$ 30,00\* (Mês + Dia)
* Para os aniversariantes do mês da promoção (mesmo mês da promoção): R$ 20,00 \*Mês
* Para os aniversariantes no dia da promoção (mesmo dia da promoção, independente do mês): R$ 0,50 \* Dia

Faça um programa, utilizando a função abaixo, para implementar esta promoção. Inicialmente, deve ler a data da promoção (‘dd/mm’) e a quantidade de participantes. A seguir, ler, para cada participante, sua data de nascimento (‘dd/mm’) e exibir seu prêmio, caso tenha direito.

No final, seu programa deve informar a quantidade de pessoas premiadas, o valor total distribuído na promoção e o maior valor individual distribuído

função **Calcula\_Prêmio** : recebe a data da promoção(‘dd/mm’ e a data de nascimento(‘dd/mm’) e retorna o valor do prêmio de acordo com o critérios acima, caso tenha direito ou 0 caso não tenha direito.

1. Para realizar um debate sobre ao acesso a dados considerados sigilosos em campanhas políticas, um professor de ética resolveu agrupar seus alunos em 3 grupos: o grupo “pró”, o grupo “contra” e o grupo “neutro”. Para saber a qual grupo o aluno pertence, o professor aplicou um questionário cujas respostas podem ser sim (‘s’) ou não (‘n’).
2. Faça uma função que recebe o número de questões, pergunte as respostas de cada uma das questões de um aluno e retorne a quantidade de respostas positivas.

Obs: o aluno digita ‘s’, quando concorda e ‘n’ quando discorda da questão

1. Faça um programa que inicialmente capture o número de respostas do questionário. A seguir, para cada aluno da turma, obtenha sua matrícula e respostas (utilizando a função do item a), e exiba o nome de seu grupo de acordo com o seguinte critério:

2/3 de respostas *sim* 🡺 grupo “pró”

2/3 de respostas *não* 🡺 grupo “contra”

Demais situações 🡺 grupo “neutro”

Término da Entrada de Dados: Matrícula de aluno negativa

1. Uma empresa de turismo realiza excursões familiares para subir a Pedra da Gávea. No entanto, é necessário que as seguintes regras sejam obedecidas:

* grupo de no mínimo 5 pessoas
* pelo menos a metade dos participantes devem ser maior de idade
* não pode haver integrantes com menos de 11 anos
* o integrante mais velho será escolhido o líder da excursão.

Faça um programa que leia o nome e a idade das pessoas de uma excursão familiar e diga se este grupo satisfaz as regras e pode participar da excursão ( informando, também, o nome do seu líder) ou se este grupo não satisfaz as regras e, por isso, não pode participar desta excursão.

Obs: o término da entrada dos participantes desta excursão ocorre quando for digitado um nome == ' '

Dica: conte o número de pessoas, de maiores de idade e de menores de 11 anos. Guarde o valor do nome e da maior idade.

*Modificação*: considere 5 excursões!

1. A tabela abaixo define o preço cobrado por uma importadora de perfumes:

|  |  |  |
| --- | --- | --- |
| Quantidade | Entrega Rápida *(US$/unidade)* | Entrega normal (US$/unidade) |
| <=1000 | 11,10 | 9,10 |
| >1000 e <=5000 | 10,00 | 10,00 |
| >5000 e <=10000 | 6,40 | 6,40 |
| >10000 | 3,30 | 3,30 |

No entanto, a importadora concede um desconto de 30% para os pedidos de entrega rápida com valor total superior a US$ 4.000,00.

a) Faça a função **valor\_unitario**, que recebe a quantidade desejada e o tipo de entrega e retorna o preço unitário cobrado pelo perfume de acordo com a tabela acima (não considerando o desconto!).

b) Faça um programa que obtenha os dados dos pedidos de cada um dos clientes, isto é, a quantidade e o tipo de entrega (R- Entrega Rápida e N - Entrega Normal) desejado. Seu programa deve mostrar para cada cliente, o valor total a pagar pelo pedido (em US$). No final seu programa deve exibir o valor total dos pedidos e quantos clientes receberam desconto.

O término da entrada de dados ocorre quando for digitado uma quantidade solicitada <=0

1. Construa um programa, utilizando adequadamente as funções dos itens a e b, para adivinhar um dia e mês (diaG, mesG) gerados aleatoriamente. O valor gerado para o mês pode variar de 1 a 12 e do dia, de 1 a 28.

Para ajudar, o programa deve exibir:

“\*\*\*” : quando o dia ϵ [diaG – 5 , diaG + 5]

“~~~”: quando o dia ∉ [diaG – 5 , diaG + 5]

“ok” : quando o dia for igual ao diaG

“\*\*\*” : quando o mes ϵ [mesG – 5 , mesG + 5]

“~~~”: quando o mes ∉ [mesG – 5 , mesG + 5]

“ok” : quando o mes for igual ao mesG

1. Construa uma função que receba dois valores x e y e retorne True, se x ϵ [y – 5 , y + 5] ou False, caso contrário
2. Construa uma função que receba dois valores x e y e exiba

‘ok’, se x == y,

“\*\*\*”, se x ϵ [y – 5 , y + 5] e

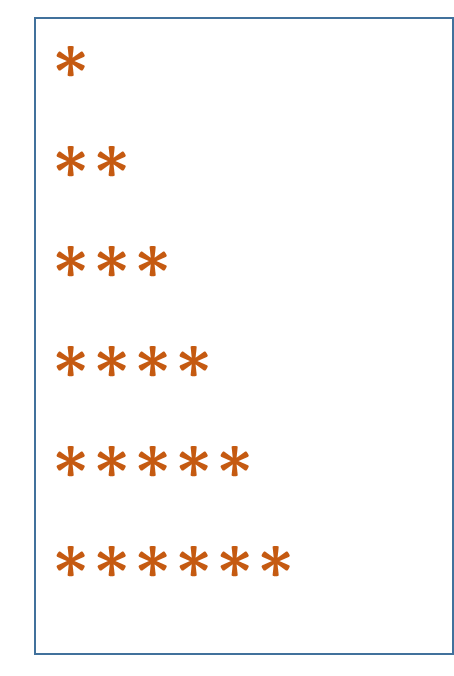
“~~~” se x ∉ [y – 5 , y + 5]

1. Uma lanchonete aceita 4 formas de pagamento: cartão de crédito, cartão de débito, dinheiro e ticket refeição. Faça um programa para exibir o valor a ser devolvido (troco) **a um cliente** ou a mensagem “Não há troco”. Inicialmente, o seu programa deverá ler o valor da despesa a ser paga (real) pelo cliente e o tipo de pagamento: ‘Débito’- para cartão de débito, ‘Crédito’-para cartão de crédito, ‘Dinheiro’ - para dinheiro ou ‘Vale’- para ticket. Se o pagamento for realizado por cartão (débito ou crédito), não há troco. Para os demais tipos de pagamento, pode ou não haver troco. Se for pagamento em dinheiro, será lido o valor pago. Se o pagamento for com tickets, devem ser lidos a quantidade de tickets e o valor de cada ticket (repare que os valores podem ser diferentes).
2. Modifique o programa para atender vários clientes. Término valor == 0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Entrada** | | | | | **Saída** |
| 120.00 | Crédito | | | | Não há troco |
| 120.00 | Débito | | | | Não há troco |
| 120.00 | Dinheiro | 150.00 | | | Troco: 30.00 |
| 120.00 | Vale | 2 | 80 | 50 | Troco: 10.00 |
| 120.00 | Vale | 2 | 70 | 50 | Não há troco |

1. A função exibeLinha, mostra na tela uma linha com n símbolos c. A saída do bloco principal abaixo é:

|  |  |  |
| --- | --- | --- |
| exibeLinha | Bloco principal | Saída |
| def exibeLinha(c,n):  **linha = c\*n**  **print(linha)**  return | i = 1  while (i <= 4):  exibeLinha('\*',4)  print() #muda a linha  i = i + 1 |  |

Usando a função exibeLinha, construa o seguinte desenho ASCII: 

1. Faça um programa que obtenha a velocidade máxima permitida em uma avenida e a placa 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 quando necessária.

A entrada de dados é finalizada ao ser digitado uma placa == ‘’ (apenas o enter)

No final seu programa deve mostrar o valor médio das multas e a placa do infrator que pagou a maior multa (considere que não há empate)

a) Faça uma função que receba um valor de hora válido e retorne True se esta hora for dentro do horário comercial (8 às 18) , ou False 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

1. a) Faça uma função que retorne a soma dos divisores de um número recebido

b) Faça uma função para exibir todos os pares de números amigáveis entre n e m recebidos. Um par de números é amigável quando cada um deles é igual à soma dos divisores do outro.

c) Faça um programa que pergunte ao usuário pares de números (x,y) torne-os os pontos iniciais e finais de um intervalo e exiba todos os pares de números amigáveis entre x,y. Caso não haja um número amigável no intervalo informado, enviar mensagem adequada.

Término da entrada de dados: x==y

1. O controle de qualidade de uma fábrica de componentes eletrônicos avalia diversas características para aceitá-los ou rejeitá-los. Para cada característica avaliada verifica sua classificação: Excelente, Aceitável, Rejeitado.

Avaliações fazem referência à área e as dimensões (comprimento e largura medida em milímetros) em relação às medidas do componente padrão, de acordo com a tabela abaixo:

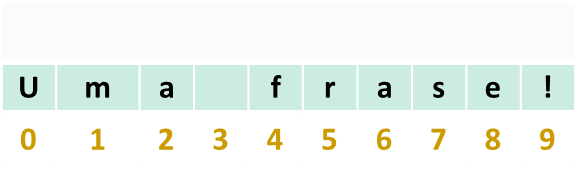
|  |  |  |
| --- | --- | --- |
| NOTA | REGRAS EM RELAÇÂO ÀS MEDIDAS DO COMPONENTE PADRÃO | |
|  | AREA | LARGURA e COMPRIMENTO |
| Excelente | Área com variação menor ou igual a 1 mm. | Altura e largura com variação menor que 1 mm. |
| Bom | Área com variação menor ou igual a 1 mm. | Altura ou largura com variação menor que 1 mm. |
| Rejeitado | Área com variação menor ou igual a 1 mm. | Altura ou largura com variação superior a 1 mm. |
| Rejeitado | Área com variação maior que 1 mm. |  |

1. Implemente uma função que recebe como parâmetro duas medidas em milímetros e retorna a diferença absoluta entre elas.
2. Utilizando, **obrigatoriamente**, a função auxiliar do item anterior, implemente uma função que recebe como parâmetro as medidas padrões e do componente e retorna a nota de classificação.
3. Escreva um programa para classificar n (lido) componentes eletrônicos. Seu programa deve capturar como entrada as dimensões (altura e comprimento) do componente padrão e as dimensões de cada componente a ser avaliado. Após utilizar as funções dos itens anteriores para realizar os cálculos, seu programa deve imprimir, como saída, a classificação de cada componente. No final, mostrar a quantidade de produtos rejeitados.
4. Foram entrevistadas 1590 pessoas sobre a popularidade de dois candidatos à presidência: candidato A e candidato B. Cada entrevistado deu uma nota (número inteiro) de 0 a 5 a cada um dos dois candidatos. Faça um programa que leia as duas notas dadas por cada entrevistado e exiba:

* A soma das notas obtidas pelo candidato A e a soma das notas obtidas pelo candidato B;
* O total de entrevistados que deram nota 0 (zero) aos 2 candidatos.

**ITERANDO SOBRE STRINGS:**

Uma string é uma sequência de caracteres. As posições dos caracteres na sequência são numeradas, iniciando em 0



A função **len** retorna o tamanho da string. Selecionamos caracteres de uma string usando o operador de indexação [ ]. Exemplos: print(frase[2]) 🡪 a print(frase[-2]) 🡪 e

1. Escreva uma função que receba uma string e um caractere e retorne a quantidade de ocorrências do caractere na string

Exemplo: 'abacate' e 'a' --> 3

'carnaval' e 'o' --> 0

1. (D/I)Escreva uma função que receba duas strings e retorne a substring com os primeiros caracteres iguais, caso existam ou '' caso contrário

Exemplos: 'abacate' e 'aberto' --> 'ab'

'carnaval' e 'carne' --> 'carn'

'camelo' e 'cedo' --> 'c'

'camelo' e 'maçã' --> ''

1. (D/I)Escreva uma função que receba duas strings, s e t, e retorne o número da posição de s onde inicia a 1ª ocorrência de t em s, ou False caso contrário.

Exemplos: 'Biobanana' e 'an' --> 4

'Biobanana' e 'on' --> False

1. (D/I)Escreva uma função que receba uma palavra e retorna True se seus caracteres estão em ordem lexicográfica e False, caso contrário.

Ex: ano --> True bola --> False

1. (D/I) Um palíndromo é uma seqüência de caracteres cuja leitura é idêntica se feita da direita para esquerda ou vice−versa. Por exemplo: OSSO e OVO são palíndromos. Em textos mais complexos os espaços e pontuação são ignorados. A frase SUBI NO ONIBUS é o exemplo de uma frase palíndroma onde os espaços foram ignorados. Faça um programa que leia uma frase, mostre−a e diga se é um palíndromo ou não.