resolução das questões da lista:

1: Mova o azul para o h2, mande o verde pro h3, coloque o azul em cima do verde, mande o amarelo para o h2, coloque o verde e o azul em cima do amarelo na ordem, mova o vermelho para o h3, mande o azul para o h1, mande o verde em cima do vermelho, coloque o azul em cima do amarelo, coloque o verde no h1, coloque o azul em cima do verde, coloque o amarelo em cima do vermelho, mande o azul para o h2, coloque o verde em cima do amarelo e pra fechar coloque o azul em cima do verde.

2: a) 255

b) você deve multiplicar o número anterior mais 1. exemplo: 3\*2 + 1= 7.

3: 1 copo mais 1 garrafa= 1 jarra

1 jarra= 1 copo mais um prato

2 jarras= 3 pratos

x copos= 1 garrafa

r= 2 copos

copo= 0,5

garrafa= 1

4: Como nenhuma mulher se sentou ao lado do marido, os homens e as mulheres devem estar sentados alternadamente. Portanto, a ordem dos assentos é: homem, mulher, homem, mulher, homem, mulher, homem, mulher.

1. Como na frente de Basílio estava Dionísio, eles estão sentados um de frente para o outro. Portanto, eles devem estar sentados da seguinte forma: Basílio, mulher, Dionísio, mulher.
2. À direita da esposa de Basílio se sentou Carlos. Como as mulheres estão sentadas entre os homens, a esposa de Basílio deve estar sentada à esquerda de Carlos. Portanto, a ordem dos assentos agora é: Basílio, mulher, Carlos, mulher, Dionísio, mulher.
3. Agora, só temos Armando e duas mulheres (esposas de Carlos e Dionísio) para colocar nos assentos restantes. Como nenhuma mulher pode se sentar ao lado do marido, a esposa de Carlos deve estar sentada à direita de Dionísio e a esposa de Dionísio à direita de Basílio. Portanto, a ordem dos assentos agora é: Basílio, esposa de Carlos, Carlos, esposa de Dionísio, Dionísio, esposa de Basílio.
4. Finalmente, Armando e a esposa de Armando ocupam os dois assentos restantes. No entanto, como Armando e a esposa de Armando não podem estar juntos, precisamos trocar a esposa de Armando com a esposa de Carlos. Portanto, a ordem final dos assentos é: Basílio, esposa de Armando, Carlos, esposa de Dionísio, Dionísio, esposa de Basílio, Armando, esposa de Carlos.

5: Primeiro, escolha um item do pote rotulado como “Ambas”. Como sabemos que todos os potes estão rotulados incorretamente, este pote só pode conter “Maçãs” ou “Laranjas”, mas não ambos.

Se você tirar uma maçã, então você sabe que o pote deve ser rotulado como “Maçãs”. Agora, você tem dois potes restantes: um rotulado como “Maçãs” e outro como “Laranjas”. Mas você sabe que nenhum desses potes pode ser o que seu rótulo atual diz (porque todos os rótulos estavam originalmente errados), então o pote rotulado como “Maçãs” deve conter “Laranjas” e o pote rotulado como “Laranjas” deve conter “Ambas”.

Se você tirar uma laranja, então você sabe que o pote deve ser rotulado como “Laranjas”. Agora, você tem dois potes restantes: um rotulado como “Maçãs” e outro como “Ambas”. Mas você sabe que nenhum desses potes pode ser o que seu rótulo atual diz (porque todos os rótulos estavam originalmente errados), então o pote rotulado como “Maçãs” deve conter “Ambas” e o pote rotulado como “Ambas” deve conter “Maçãs”

6: Não fiz

7: Os algoritmos são um conjunto de instruções passo a passo que descrevem como realizar uma tarefa. Eles podem ser representados de três maneiras principais: descrição narrativa, fluxograma e pseudocódigo.

1. Descrição Narrativa: É a descrição escrita do algoritmo em linguagem natural (português, inglês, etc.).
   * Vantagens: Fácil de escrever e entender.
   * Desvantagens: Pode ser ambíguo e não estruturado.
2. Fluxograma: É uma representação gráfica do algoritmo usando formas geométricas padronizadas.
   * Vantagens: Visualmente claro e fácil de seguir.
   * Desvantagens: Difícil de modificar e não é prático para algoritmos complexos.
3. Pseudocódigo: É uma representação do algoritmo que se assemelha a uma linguagem de programação, mas não segue rigorosamente sua sintaxe.
   * Vantagens: Fácil de converter em código real e estruturado.
   * Desvantagens: Requer algum conhecimento de programação.

Agora, vamos representar um algoritmo para converter graus Celsius para Fahrenheit nas três formas:

Descrição Narrativa:

O usuário insere um valor em graus Celsius. O algoritmo multiplica esse valor por 9/5 e adiciona 32 para converter para Fahrenheit. Se a temperatura em Fahrenheit for maior que 100, o algoritmo exibe "Você está com febre". Caso contrário, exibe "Relaxa, tá tudo bem".

Fluxograma:

INÍCIO

|

V---> Insira a temperatura em Celsius

|

V---> Fahrenheit = Celsius \* 9/5 + 32

|

V---> Se Fahrenheit > 100

| |

| V

| ---> Exibir "Você está com febre"

| |

V V

---> Senão

|

V

---> Exibir "Relaxa, tá tudo bem"

|

V

FIM

Pseudocódigo:

INÍCIO

LEIA Celsius

Fahrenheit = Celsius \* 9/5 + 32

SE Fahrenheit > 100 ENTÃO

ESCREVA "Você está com febre"

SENÃO

ESCREVA "Relaxa, tá tudo bem"

FIM SE

FIM

8: Linguagem Natural: É a linguagem que usamos no dia a dia para nos comunicarmos uns com os outros. Exemplos de linguagens naturais incluem português, inglês, espanhol, etc. Elas são ricas e complexas, permitindo expressar uma ampla gama de ideias e emoções.

1. Linguagem de Máquina: É a linguagem que o computador entende diretamente. É uma linguagem de baixo nível que consiste em strings binárias (0s e 1s) que representam instruções para o processador do computador.
2. Linguagem de Programação: É uma linguagem formal projetada para expressar instruções que podem ser executadas por um computador. Elas permitem que os programadores escrevam algoritmos de maneira que possam ser compreendidos e executados por um computador. Exemplos incluem Python, Java, C++, etc.
3. Compilador: É um programa que traduz código de uma linguagem de programação (linguagem de alto nível) para uma linguagem de máquina (linguagem de baixo nível). O resultado é um arquivo executável que pode ser executado pelo sistema operacional.
4. Interpretador: É semelhante a um compilador, mas em vez de traduzir todo o programa de uma vez e depois executá-lo, o interpretador traduz e executa cada linha de código sequencialmente. Isso permite uma interação mais imediata, mas pode ser mais lento para programas grandes.

9: Linguagens de Baixo Nível: São linguagens que estão mais próximas da linguagem de máquina, ou seja, do hardware. Elas são mais difíceis de ler e escrever para os humanos, mas oferecem mais controle sobre o hardware do computador. Existem dois tipos principais de linguagens de baixo nível:

* + Linguagem de Máquina: É a linguagem de nível mais baixo, consistindo em código binário (0s e 1s). Por exemplo, um comando para adicionar dois números pode parecer “11011010 01101011”.
  + Linguagem Assembly: É um pequeno passo acima da linguagem de máquina. Usa abreviações ou mnemônicos para representar operações e locais de memória. Por exemplo, um comando para adicionar dois números pode parecer “ADD R1, R2”.

1. Linguagens de Alto Nível: São linguagens que estão mais próximas da linguagem humana (mais fáceis de ler e escrever) e mais distantes da linguagem de máquina. Elas são projetadas para serem mais fáceis de usar, embora ofereçam menos controle sobre o hardware do computador. Exemplos de linguagens de alto nível incluem Python, Java, C++, etc. Por exemplo, um comando para adicionar dois números em Python seria simplesmente “sum = num1 + num2”.

As linguagens de alto nível precisam ser traduzidas para a linguagem de máquina antes de poderem ser executadas, o que é feito por compiladores ou interpretadores. As linguagens de baixo nível, por outro lado, podem ser executadas diretamente pelo hardware do computador.

10: Python:

Python

Código gerado por IA. Examine e use com cuidado. [Mais informações em perguntas frequentes](https://www.bing.com/new#faq).

x = 2

y = 6

resultado = x \* y

print(resultado)

Java:

Java

Código gerado por IA. Examine e use com cuidado. [Mais informações em perguntas frequentes](https://www.bing.com/new#faq).

public class Main {

public static void main(String[] args) {

int x = 2;

int y = 6;

int resultado = x \* y;

System.out.println(resultado);

}

}

As diferenças entre as linguagens de programação existem por várias razões. Cada linguagem de programação foi projetada com um conjunto específico de objetivos em mente, seja para facilitar a escrita de certos tipos de programas, para rodar eficientemente em certos tipos de hardware, ou para ser fácil de aprender e usar. Além disso, as linguagens de programação evoluem ao longo do tempo, com novas linguagens sendo criadas para resolver problemas que as linguagens mais antigas não conseguem resolver de forma eficiente. Portanto, a sintaxe e a estrutura de cada linguagem de programação podem variar significativamente.

11: As linguagens de programação geralmente têm um conjunto de instruções primitivas que incluem:

1. Atribuição: Atribui um valor a uma variável. Por exemplo, x = 5.
2. Entrada/Saída: Recebe dados do usuário ou exibe dados para o usuário. Por exemplo, print() para saída e input() para entrada em Python.
3. Aritmética: Realiza operações matemáticas. Por exemplo, +, -, \*, /.
4. Condicional: Executa diferentes blocos de código com base em uma condição. Por exemplo, if...else.
5. Looping: Executa um bloco de código várias vezes. Por exemplo, for, while.
6. Goto: Desvia a execução para outra parte do programa.

Todas essas instruções primitivas estão presentes na maioria das linguagens de programação modernas, incluindo Python. No entanto, a instrução goto é uma exceção notável. Python não suporta a instrução goto porque pode tornar o fluxo de controle do programa difícil de entender e levar a um código confuso e propenso a erros, conhecido como “código espaguete”.

12: Variáveis são espaços de memória que armazenam valores que podem ser alterados durante a execução de um programa. Elas são identificadas por nomes, que devem seguir certas regras e convenções.

No Python, os nomes das variáveis devem começar com uma letra (a-z, A-Z) ou um sublinhado (\_), seguido por qualquer número de letras, dígitos (0-9) ou sublinhados. Eles não podem começar com um dígito e não podem conter espaços, traços ou outros caracteres especiais. Além disso, existem algumas palavras reservadas que não podem ser usadas como nomes de variáveis porque têm um significado especial para a linguagem Python (por exemplo, print).

Aqui está a avaliação dos nomes que você forneceu:

* print: Não é permitido, pois print é uma palavra reservada em Python.
* customer\_list: É permitido, segue as regras de nomenclatura.
* num-people: Não é permitido, pois contém um traço (-), que não é permitido em nomes de variáveis.
* $teste: Não é permitido, pois começa com um caractere especial ($).
* 2\_numero: Não é permitido, pois começa com um dígito.

13: Constantes em Python são variáveis cujos valores são fixos e não mudam durante a execução de um programa. Tecnicamente, Python não tem suporte nativo para constantes. No entanto, é uma prática comum tratar qualquer variável com um nome em letras maiúsculas como uma constante.

Por exemplo:

Python

Código gerado por IA. Examine e use com cuidado. [Mais informações em perguntas frequentes](https://www.bing.com/new#faq).

PI = 3.14159

GRAVITY = 9.8

Neste caso, PI e GRAVITY são tratados como constantes e, por convenção, não devem ser alterados depois de definidos. No entanto, Python não impedirá que você altere o valor dessas “constantes”, pois, do ponto de vista da linguagem, elas são apenas variáveis normais. A ideia de tratá-las como constantes é mais uma convenção entre os programadores do que uma regra imposta pela linguagem.