Descrição

Durante a pandemia, a utilização de aplicativos para fazer pedidos em restaurantes e supermercados se tornou muito comum.

Neste problema, você terá que simular o recebimento de pedidos de um restaurante que faz vendas por aplicativo.

Para fazer essa simulação de recebimento de pedidos, você terá que implementar seis classes: Venda, Pedido, Produto, Pizza, Hamburguer e Japonesa.

OBS: Os arquivos venda.hpp, pedido.hpp e produto.hpp não devem ser modificados.

A classe Produto armazena as seguintes informações:

| **Produto** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| # quantidade: int |
| # valor\_unitario: float |

As classes Pizza, Hamburguer e Japonesa deverão ser subclasse de Produto e possuir as seguintes especificações:

| **Pizza** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| - sabor: string |
| - pedacos: int |
| - borda\_recheada: bool |
| + Pizza(string sabor, int pedacos, bool borda\_recheada, int qtd, float valor\_unitario) |

| **Hamburguer** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| - tipo: string |
| - artesanal: bool |
| + Hamburguer(string tipo, bool artesanal, int qtd, float valor\_unitario) |

| **Japonesa** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| - combinado: string |
| - sushis: int |
| - temakis: int |
| - hots: int |
| + Japonesa(string combinado, int sushis, int temakis, int hots, int qtd, float valor\_unitario) |

A classe Pedido deverá ter uma lista de produtos, o endereco de entrega e os seguintes métodos deverão ser implementados (modifiquem apenas o arquivo pedido.cpp):

| **Pedido** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| - produtos: std::list<Produto\*> |
| - endereco: string |
| + adicionaProduto(Produto\* p): void |
| + calculaTotal(): float |
| + resumo(): string |
| + setEndereco(string endereco): void |

Breve descrição:

* void adicionaProduto(Produto\* p); // adiciona um produto ao pedido
* float calculaTotal(); // calcula e retorna o valor total do pedido
* string resumo(); // retorna um resumo do pedido (uma descrição de todos os produtos que fazem parte do pedido e o endereço de entrega no final)
* setEndereco(string endereco); // atualiza o endereço de entrega do pedido

A classe Venda, terá uma lista de pedidos recebidos e os métodos a seguir (modifiquem apenas o arquivo venda.cpp):

| **Venda** |
| --- |
| ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------ |
| - pedidos: std::list<Pedido\*> |
| + adicionaPedido(Pedido\* p): void |
| + imprimeRelatorio(): void |

Breve descrição:

* void adicionaPedido(Pedido\* p); // adiciona um pedido à lista de pedidos recebidos
* void imprimeRelatorio(); // imprime a lista completa de todos pedidos processados, o total de vendas e a quantidade de pedidos recebidos

Help

<https://www.cplusplus.com/doc/tutorial/classes/>

<https://www.cplusplus.com/reference/list/list/>

<https://www.cplusplus.com/reference/string/string/>

<https://www.cplusplus.com/doc/tutorial/inheritance/>

<https://www.cplusplus.com/doc/tutorial/polymorphism/>

<http://www.cplusplus.com/forum/beginner/222475/>

**Formato de entrada**

Os pedidos são compostos por uma lista de produtos e um endereço de entrega. A tag pedido indica o início de um novo pedido e a tag endereco indica o final do pedido. Cada produto é representado por duas linhas. A primeira linha informa o tipo do produto (pizza, hamburguer ou comida japonesa) e a segunda linha informa os atributos do produto. Os atributos de cada produto seguem a mesma ordem dos seus construtores.

**Formato de saída**

Após o processamento de todos os pedidos, o método imprimeRelatorio() da classe Venda é executado. O relatório deverá imprimir uma descrição detalhada de todos os pedidos que foram processados, a quantidade de pedidos e o total de vendas. Assim como no exemplo a seguir.

**Exemplo de entrada**

pedido

pizza

Calabresa,4,1,2,40

pizza

Frango com catupity,6,0,1,55.90

hamburguer

X-tudo,1,2,25

endereco

Av. Pres. Antonio Carlos, 6627 - Pampulha, Belo Horizonte - MG

pedido

hamburguer

X-salada,0,4,15

japonesa

Combo 1,4,5,6,1,70

endereco

Rua Reitor Pires Albuquerque, ICEx - Pampulha, Belo Horizonte - MG

**Exemplo de Saída**

Pedido 1

2X Pizza Calabresa, 4 pedacos e borda recheada.

1X Pizza Frango com catupity, 6 pedacos e sem borda recheada.

2X Hamburguer X-tudo artesanal.

Endereco: Av. Pres. Antonio Carlos, 6627 - Pampulha, Belo Horizonte - MG

Pedido 2

4X Hamburguer X-salada simples.

1X Comida japonesa - Combo 1, 4 sushis, 5 temakis e 6 hots.

Endereco: Rua Reitor Pires Albuquerque, ICEx - Pampulha, Belo Horizonte - MG

Relatorio de Vendas

Total de vendas: R$ 315.90

Total de pedidos: 2

Depurando localmente

Caso você queira testar seu VPL localmente, você pode baixar os arquivos necessários [aqui](https://drive.google.com/file/d/1EYhG21AeE4KD7ao-udpHxVXwIrDuUgYB/view?usp=sharing), e usá-los com diferentes arquivos de entrada, assim:

g++ \*.cpp -o main

./main < input.txt