

BUY NOW - PART 2

João Andrade, up201905589 | Sérgio Estevão, up201905680 | Sofia Germer, up201907461
Turma 2MIEIC03

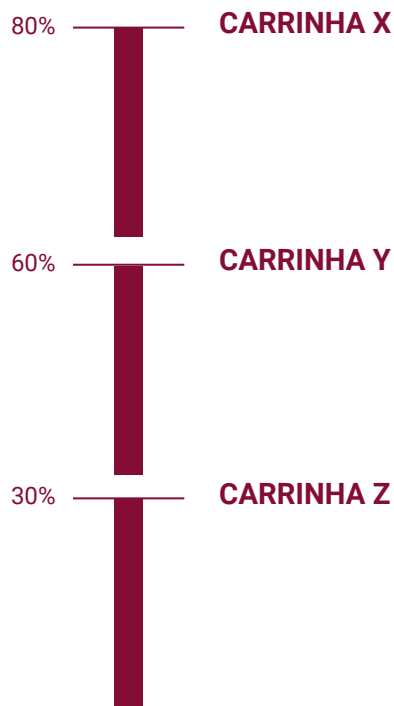
PROBLEMA - Complementar Projeto



Fonte: Google Images, <https://pxhere.com/en/photo/1440159>

- ❖ Adicionar sistema de carrinhas que se baseia numa **fila de prioridades**, sendo consideradas prioritárias as carrinhas com maior ocupação
- ❖ Adicionar um sistema de apoio ao cliente, através de uma **hastable** que guarda as mensagens que cada cliente deseja deixar para posterior tratamento.
- ❖ Implementar um sistema de compra de um produto a um fornecedor, sendo que estes fornecedores se encontram guardados numa **binary tree**.

FILA DE PRIORIDADE



- ❖ As carrinhas (trucks) estão organizadas numa fila de prioridade de forma a que a carrinha no topo seja a que tenha menos espaço vazio (ou seja, segundo o diagrama a primeira carrinha a ser enviada seria a carrinha x)
- ❖ Sempre que uma compra é efetuada (na classe transaction) são enviadas todas as carrinhas cuja ocupação seja superior a 90%
- ❖ Existe também a opção ao manager de enviar todas as carrinhas que ele desejar

Trucks System

```
bool Truck::operator<(const Truck &t) const {  
    return available_volume < t.getAvailableVolume();  
}
```

Overload do operador para que no topo da fila esteja a carrinha de menor volume disponível)

```
----- Trucks available -----  
  
-----  
Truck nrº: 1  
Available Volume: 18 | Capacity: 30  
Truck nrº: 2  
Available Volume: 18 | Capacity: 37  
Truck nrº: 3  
Available Volume: 12 | Capacity: 30  
Truck nrº: 4  
Available Volume: 12 | Capacity: 35  
Truck nrº: 5  
Available Volume: 11 | Capacity: 30  
Truck nrº: 6  
Available Volume: 11 | Capacity: 32  
-----  
Do you wish to ship any of the trucks above? yes  
Which one would you like to ship? 1  
-----  
Truck nrº: 1  
Available Volume: 18 | Capacity: 37  
Truck nrº: 2  
Available Volume: 12 | Capacity: 35  
Truck nrº: 3  
Available Volume: 12 | Capacity: 30  
Truck nrº: 4  
Available Volume: 11 | Capacity: 30  
Truck nrº: 5  
Available Volume: 11 | Capacity: 32  
-----
```

exemplo de interação com o manager
para enviar uma truck específica

Hash Table

- ❖ A hash table é constituída por várias mensagens, sendo que cada uma destas mensagens é identificada pelo email do cliente que a deixou.
- ❖ As mensagens consideram-se iguais se tiverem o membro-dado “message” igual, que representa a mensagem em questão, permitindo assim cada cliente poder deixar mais do que uma mensagem.

```
bool operator() (const Message& ur1, const Message& ur2) const
{
    return ur1.getMessage() == ur2.getMessage();
}
```

Hash Table

- ❖ Devido a uma função de escrita para um ficheiro de texto, o manager pode verificar todas as mensagens deixadas anteriormente.

```
The email from this message is: lll@gm.com which belongs to the user: poooo
Message: aeda
This message has already been delt with!

-----

The email from this message is: geg@gma.com which belongs to the user: eeeee
Message: ola
This message has already been delt with!

-----
```

BST - BINARY SEARCH TREE

```
BST<supplier> Shop::getAllSuppliers(){  
    BST<supplier> ret = vectorProducts.at( n: 0).getSuppliers();  
    for(int c = 1; c < vectorProducts.size(); c++){  
        auto tree = vectorProducts.at(c).getSuppliers();  
        BSTItrIn<supplier> it(tree);  
        while (!it.isAtEnd()){  
            ret.insert(it.retrieve());  
            it.advance();  
        }  
    }  
    return ret;  
}
```

Exemplo de utilização da BST : Atualiza a main tree de suppliers

CRUD - ALTERAÇÕES

```
1 12 ; 30
2 11 ; 30
3 18 ; 30
4 12 ; 35
5 11 ; 32
6 18 ; 37
```

trucks.txt

Adicionamos este ficheiro de texto que é lido quando a Shop é criada (`read_trucks_file()`) e é atualizado no final do programa (`update_trucks_file()`)

```
1  acucar ; 33 ; 2
2  aqua ; 84 ; 1
3  amendoins ; 40 ; 3
4  arroz ; 48 ; 4
5  banana ; 0 ; 5
6  batatas-frita ; 26 ; 3
7  bolachas ; 34 ; 4
```

products.txt

Adicionamos a todos os ficheiros de produtos (tanto da online shop como de cada loja real) o atributo volume, visto que produtos diferentes necessitam de embalagens de tamanhos diferentes.

```
1 lll@gm.com;fantastico;1
2 ctvybui@gm.com;projeto aeda;0
3 geg@gma.com;aeda;1
4 gege@gm.com;ola;0
5
```

messagesbackup.txt

Adicionamos todas as mensagens já tratadas ou não para um ficheiro, para poderem ficar no histórico da loja. Assim o manager poderá tratar das mensagens antes e após reinício da loja.

```
1 Espama;232565865;leite;0.43;50
2 Braganca;789654121;leite;0.45;200
3 yutte;785456545;leite;0.98;600
4 Pat;8523254565;leite;1.12;200
5 Thomas;7412545658;ovos;1.5;1500
6 Thomas;7412545658;ovos;1.5;1470
7 Poll;7458545854;leite;1.5;200
8 Simon;412123552;ovos;1.5;100
9 Robb;858585858;ovos;1.55;200
10 Arthur;123020202;ovos;2.5;200
11 Jaime;454545454;ovos;3;500
12
```

supplier.txt

Adicionamos informação sobre cada supplier

SPECIAL FEATURE

```
=====
      _   _   _   _   _   _   _   _   _   _
    |_| |_| |_| |_| |_| |_| |_| |_| |_| |_| |_|
    |_) |_| |_| |_| |_| |_| |_| |_| |_| |_| |_|
    |_<|_| |_| |_| |_| |_| |_| |_| |_| |_| |_|
    |_) |_| |_| |_| |_| |_| |_| |_| |_| |_| |_|
    |___/ \___/ |_| |_| |_| |_| |_| |_| |_| |_|

=====

1. Client's mode
2. Manager mode
3. Exit

=====

Make your choice: 2
Enter password: aeada
Password Accepted... Welcome!

=====

1. Change Profit Margin
2. Check Shop Statistics
3. Handle Client's Messages
4. Manage Trucks
5. Manage Suppliers
6. Go Back

=====

Make your choice: 5
```

- ❖ Escolhemos destacar o nosso método de organização da loja (em modo cliente e modo manager) uma vez que consideramos que o cliente não deveria ter a permissão para realizar determinadas ações.
- ❖ Adicionamos as seguintes funcionalidades ao manager:
 1. como gerir carrinhas de encomendas
 2. lidar com um sistema de apoio ao cliente
 3. gerir fornecedores da loja.

DIFICULDADES ENCONTRADAS

1. Uma vez que não nos foi divulgada a avaliação da primeira parte do projeto tornou-se complicado saber o que foi bem feito e o que poderíamos ter feito melhor, sendo que isso poderia proporcionar uma maior consciência sobre o que é pretendido no projeto.
2. Devido à data de entrega colidir com a entrega de outros dois projetos de unidades curriculares distintas, tornou a gestão de tempo muito complicada e não conseguimos dedicar o tempo que gostaríamos ao projeto.