C++ Supermarket Checkout

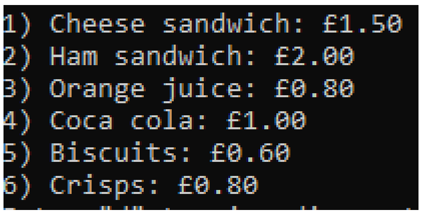
Running the program

Cd into “cmake-build-debug” and run the executable “checkout”.

The program is run through user input in the command line. The receipt is written to a file called “receipt.txt” in the “cmake-build-debug” directory.

Supermarket Stock

The supermarket has the following items available for purchase with the following prices:



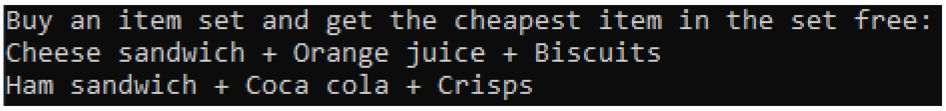
```
1) Cheese sandwich: £1.50
2) Ham sandwich: £2.00
3) Orange juice: £0.80
4) Coca cola: £1.00
5) Biscuits: £0.60
6) Crisps: £0.80
```

More items can be added to the stock by adding to the “generate_stock” function in main.cpp:

Item Discounts

If 3 of the same item is purchased, one of them is given for free.

The supermarket has the following item sets, where if all the items in the set are purchased, the cheapest will be free:



```
Buy an item set and get the cheapest item in the set free:
Cheese sandwich + Orange juice + Biscuits
Ham sandwich + Coca cola + Crisps
```

More item sets can be created by adding to the “generate_sets” function in main.cpp.

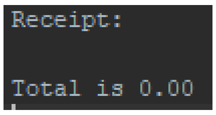
Item sets do not allow for the overlapping of items, i.e. an item cannot appear in multiple item sets. If this were allowed, it would introduce the problem of which item set is the discount applied to. Take a scenario where a ham sandwich appears in 2 item sets, and a customer buys items so that both sets are complete but they only buy a single ham sandwich. Since only a single ham sandwich has been bought, discounts cannot be applied to both item sets, but we have to decide which set the discount is applied to. We could choose the discount based on the order the items are scanned, or we could always choose the smaller discount.

Ultimately, I decided that in a real scenario this would be too confusing to the customers, and it would be hard for them to understand how much they are spending. Therefore, I kept it simple and made it so item sets cannot overlap. Currently, the program assumes item sets contain exactly 3 items, but more items can be allowed in item sets with some changes to the “calculate_set_discount” function in main.cpp.

Tests

Below show screenshots of tests I conducted to cover various scenarios, showing the receipt:

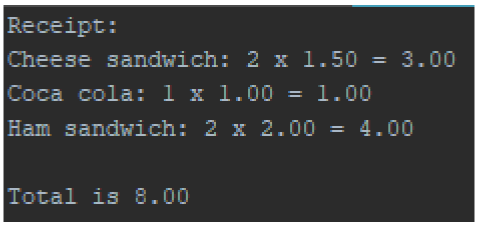
Nothing bought:



```
Receipt:

Total is 0.00
```

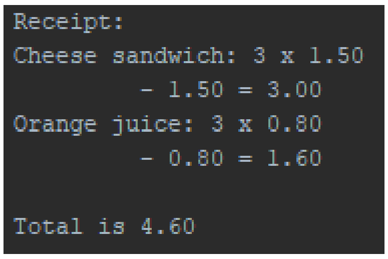
2 ham sandwiches, 2 cheese sandwiches, 1 coca cola (no complete sets):



```
Receipt:
Cheese sandwich: 2 x 1.50 = 3.00
Coca cola: 1 x 1.00 = 1.00
Ham sandwich: 2 x 2.00 = 4.00

Total is 8.00
```

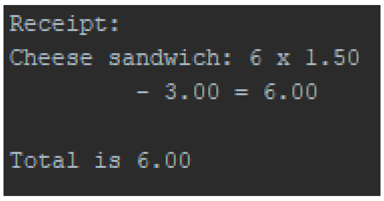
3 ham sandwiches, 3 orange juices:



```
Receipt:
Cheese sandwich: 3 x 1.50
         - 1.50 = 3.00
Orange juice: 3 x 0.80
         - 0.80 = 1.60

Total is 4.60
```

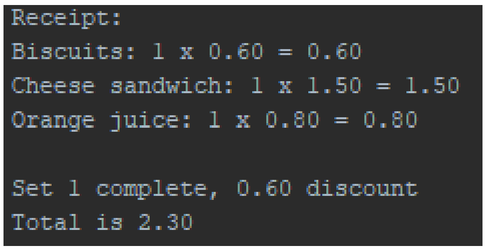
6 ham sandwiches:



```
Receipt:
Cheese sandwich: 6 x 1.50
         - 3.00 = 6.00

Total is 6.00
```

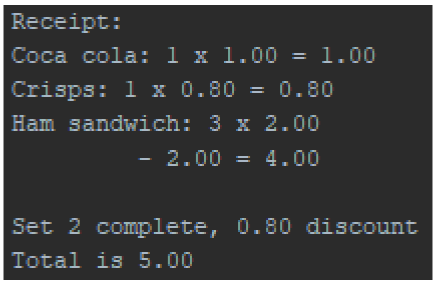
1 ham sandwich, 1 orange juice, 1 biscuits (item set 1 complete):



```
Receipt:
Biscuits: 1 x 0.60 = 0.60
Cheese sandwich: 1 x 1.50 = 1.50
Orange juice: 1 x 0.80 = 0.80

Set 1 complete, 0.60 discount
Total is 2.30
```

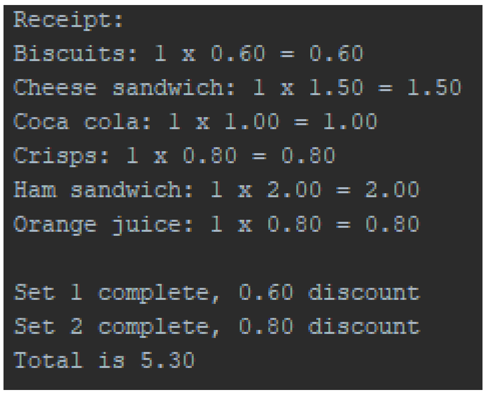
3 cheese sandwich, 1 coca cola, 1 crisps (item set 2 complete):



```
Receipt:
Coca cola: 1 x 1.00 = 1.00
Crisps: 1 x 0.80 = 0.80
Ham sandwich: 3 x 2.00
         - 2.00 = 4.00

Set 2 complete, 0.80 discount
Total is 5.00
```

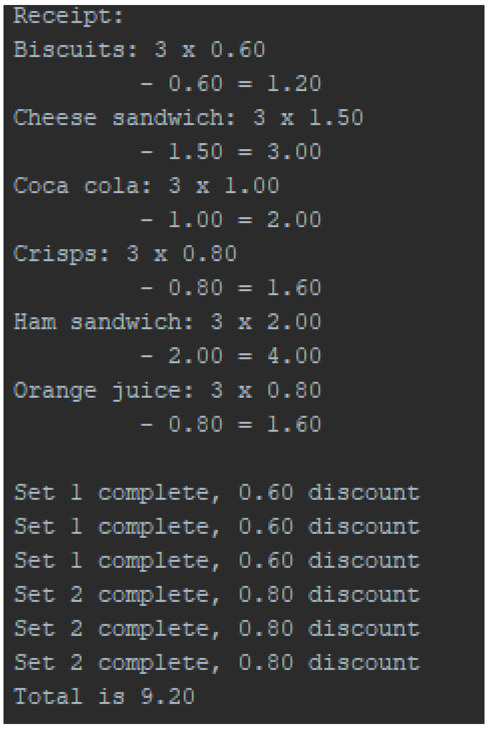
1 of everything (both item sets complete):



```
Receipt:
Biscuits: 1 x 0.60 = 0.60
Cheese sandwich: 1 x 1.50 = 1.50
Coca cola: 1 x 1.00 = 1.00
Crisps: 1 x 0.80 = 0.80
Ham sandwich: 1 x 2.00 = 2.00
Orange juice: 1 x 0.80 = 0.80

Set 1 complete, 0.60 discount
Set 2 complete, 0.80 discount
Total is 5.30
```

3 of everything:



```
Receipt:
Biscuits: 3 x 0.60
         - 0.60 = 1.20
Cheese sandwich: 3 x 1.50
         - 1.50 = 3.00
Coca cola: 3 x 1.00
         - 1.00 = 2.00
Crisps: 3 x 0.80
         - 0.80 = 1.60
Ham sandwich: 3 x 2.00
         - 2.00 = 4.00
Orange juice: 3 x 0.80
         - 0.80 = 1.60

Set 1 complete, 0.60 discount
Set 1 complete, 0.60 discount
Set 1 complete, 0.60 discount
Set 2 complete, 0.80 discount
Set 2 complete, 0.80 discount
Set 2 complete, 0.80 discount
Total is 9.20
```