Courier Kata

You work for a courier company and have been tasked with creating a code library to calculate the cost of sending an order of parcels.

- The API for the library should be programmatic. There is no need to implement a CLI, HTTP, or any other transport layer
- Try not to peek ahead at future steps and commit your working as you go
- Input can be in any form you choose
- Output should be a collection of items with their individual cost and type, as well as total cost
- In all circumstances the cheapest option for sending each parcel should be selected.
- You are expected to use test driven development
- Take no longer than 2 hours! Do what you can and give us a rough outline of what further changes you might consider making.

Implementation Steps

1) The initial implementation just needs to take into account a parcel's size. For each size type there is a fixed delivery cost

- Small parcel: all dimensions < 10cm. Cost $3
- Medium parcel: all dimensions < 50cm. Cost $8
- Large parcel: all dimensions < 100cm. Cost $15
- XL parcel: any dimension >= 100cm. Cost $25

2) Thanks to logistics improvements we can deliver parcels faster. This means we can charge more money. Speedy shipping can be selected by the user to take advantage of our improvements.

- This doubles the cost of the entire order
- Speedy shipping should be listed as a separate item in the output, with its associated cost
- Speedy shipping should not impact the price of individual parcels, i.e. their individual cost should remain the same as it was before

3) There have been complaints from delivery drivers that people are taking advantage of our dimension only shipping costs. A new weight limit has been added for each parcel type, over which a charge per kg of weight applies

+$2/kg over weight limit for parcel size:

- Small parcel: 1kg
- Medium parcel: 3kg

- Large parcel: 6kg
- XL parcel: 10kg

4) Some of the extra weight charges for certain goods were excessive. A new parcel type has been added to try and address overweight parcels

Heavy parcel (limit 50kg), $50. +$1/kg over

5) In order to award those who send multiple parcels, special discounts have been introduced.

- Small parcel mania! Every 4th small parcel in an order is free!
- Medium parcel mania! Every 3rd medium parcel in an order is free!
- Mixed parcel mania! Every 5th parcel in an order is free!

- Each parcel can only be used in a discount once
- Within each discount, the cheapest parcel is the free one
- The combination of discounts which saves the most money should be selected every time

Example:
6x medium parcel. 3 x $8, 3x $10. 1st discount should include all 3 $8 parcels and save $8. 2nd discount should include all 3 $10 parcels and save $10.

- Just like speedy shipping, discounts should be listed as a separate item in the output, with associated saving, e.g. "-2"
- Discounts should not impact the price of individual parcels, i.e. their individual cost should remain the same as it was before
- Speedy shipping applies after discounts are taken into account

Tips!

- Apply good software design principles.
- Keep your design simple, don't over-engineer.
- TDD all the way.
- Let the structure of your design evolve as you add more tests.
- Start simple. For instance, you might start with a test that if a 1cmx1cmx1cm size parcel, the result should be: `Small Parcel: $3. Total Cost: $3`
- Commit as you go, we like to see your progress and approach.