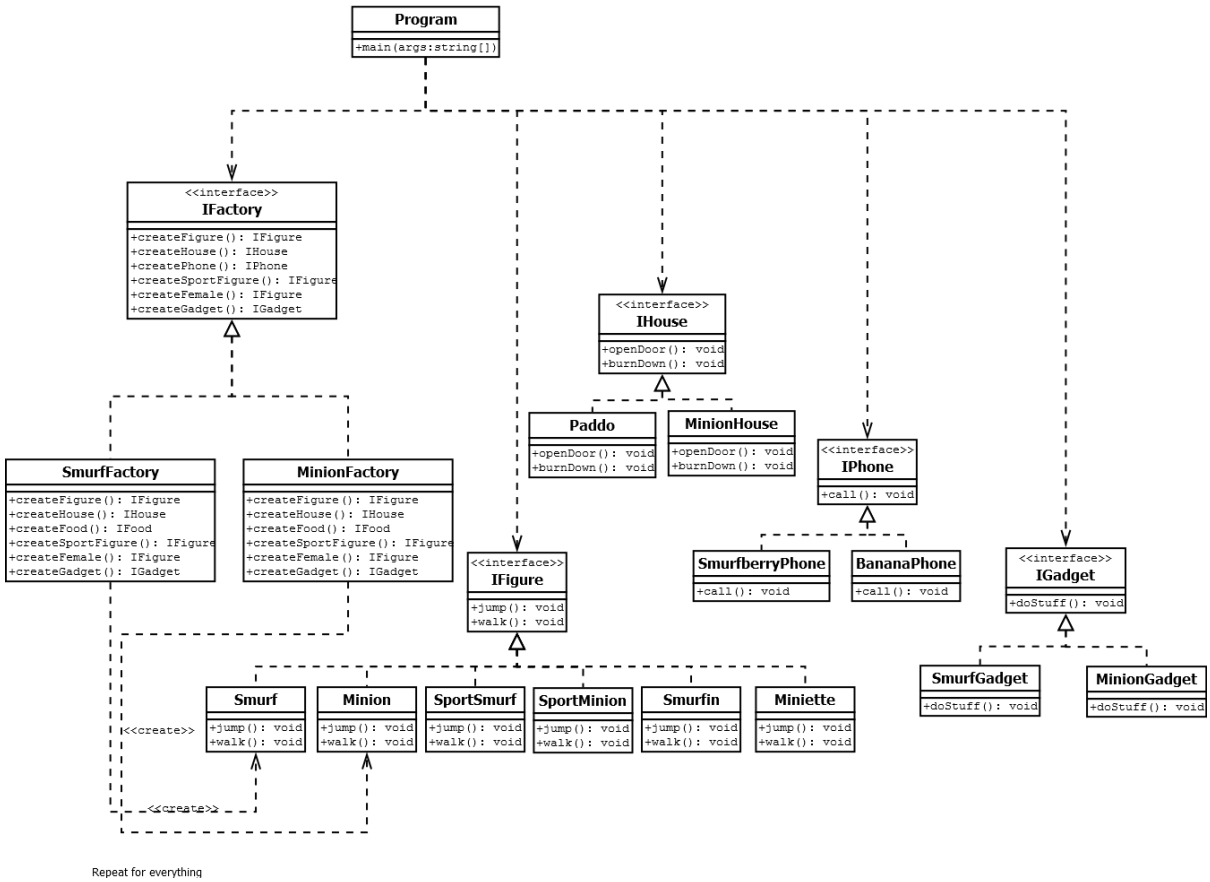
Week 3: Abstract Factory

Minh-Triet Diep - Lars Jaeqx - T61

Het patroon

Het abstract factory patroon zorgt ervoor dat er verschillende families van afhankelijke objecten kunnen worden hergebruikt. Je hoeft hierdoor niet de specifieke implementatie van de families te specificeren.

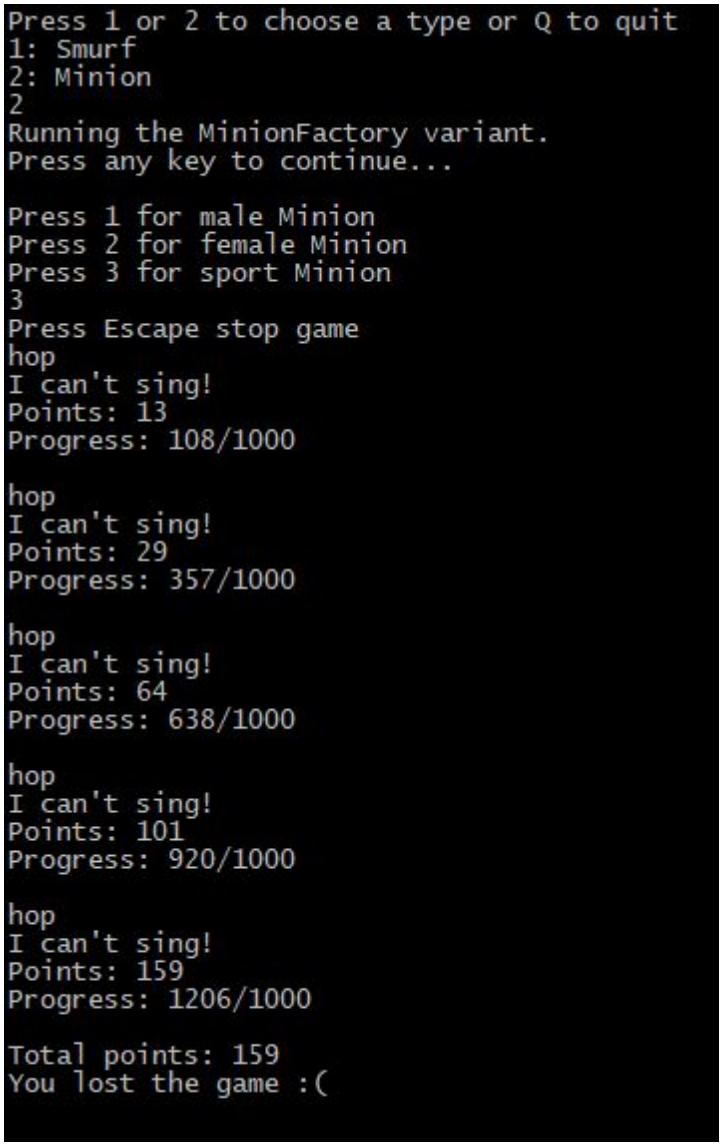
UML



Test run results

De implementatie is een “game” waarin Smurfen of Minions dingen doen en punten verdienen. Voor demonstratiedoeleinden is er op runtime te kiezen tussen de Smurf- of Minionfactory. In de implementatie zelf worden dan met de generieke methodes een smurf of minion-object gemaakt, die onderling kunnen verschillen.

Hier een voorbeeld-run:



Voor en nadelen

Reusability

+ De interfaces kunnen voor elk soort game worden gebruikt.
+ Objecten van verschillende games kunnen worden uitgewisseld.

Maintainability

− Om veranderingen te maken moet dit in elk soort game apart worden aangepast.

Extensibility

+ Je kunt nieuwe soorten game makkelijk toevoegen.
+ Objecten van verschillende games kunnen worden uitgewisseld.
− Een bestaande klasse uitbreiden is niet makkelijk omdat dit dan in alle soorten game moet worden aangepast.

Reflectie

Het maken van dit patroon was relatief eenvoudig. Het implementeren van de klassen was alleen veel werk omdat elke klasse van verschillende games apart moeten worden gemaakt.