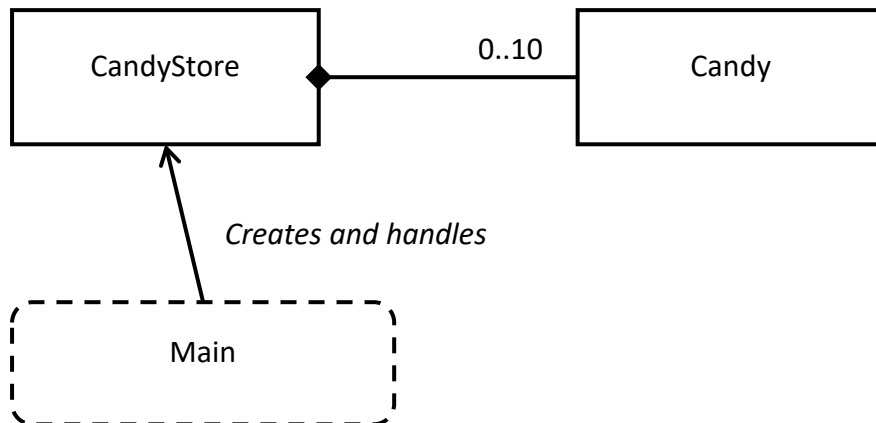


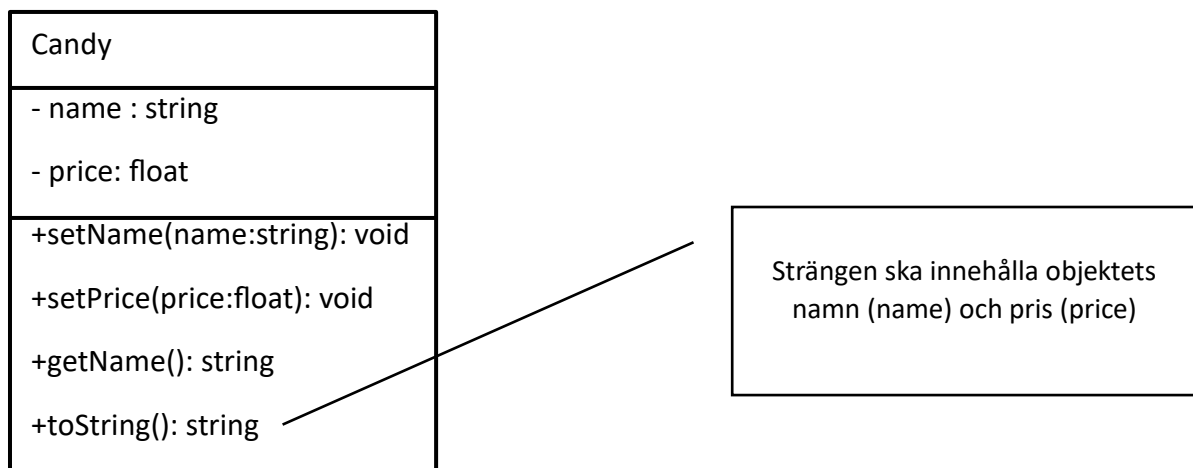
Övergripande klassdiagram

Du ska implementera klasserna **Candy** och **CandyStore** enligt relationen i klassdiagrammet:



Detaljerade klassdiagram för respektive klass

Candy



Det vara möjligt att

- skapa ett Candy-objekt givet *name* och *price*
- skapa ett Candy-objekt utan att bifoga något alls
- jämföra Candy-objekt genom användande av <operatorn vilken ska baseras på *price*
- jämföra Candy-objekt genom användande av ==operatorn vilken ska baseras på både *price* och *name*

Alla medlemsvariabler ska vara **privata**.

Inga andra funktioner eller variabler får tillföras i klassen.

Testprogrammet **CandyTest.cpp** ska användas för att testa din implementation av Candy-klassen.

CandyStore

CandyStore
- storeName: string - currentNrOfCandies: int
+ getStoreName(): string + getCurrentNrOfCandies(): int + addCandy(name: string, price: float): bool + removeCandy(name: string, price: float): bool

Relationen mellan CandyStore och Candy ska implementeras genom användande av en **statiskt allokerad array innehållande pekare**

Vid tillägg (funktionen addCandy) placeras det nya objektet direkt efter redan befintliga om det finns plats. Vidare returneras true. Om det inte finns plats returneras i stället false.

Vid borttagning (funktionen removeCandy) ska borttaget objekt ersättas med det som är sist av alla. Om borttagning genomförts returneras true och annars returneras false (objekt motsvarande parametrarna fanns inte).

Vidare ska det vara möjligt att

- skapa ett CandyStore-objekt givet *storeName*
- skapa ett CandyStore-objekt utan att bifoga något

För testningen krävs dessutom medlemsfunktionen **Candy* candyAt(int index)** som ska returnera adressen/pekaren till det Candy-objekt som finns i arrayen på det index som parametern motsvarar.

Du behöver tillföra **medlemsvariabel/medlemsvariabler för att implementera relationen** till klassen Candy.

Alla medlemsvariabler ska vara **privata**.

Inga andra funktioner eller variabler får tillföras i klassen.

Din implementation får **inte generera några minnesläckor**.

Testprogrammet **CandyStoreTest.cpp** ska användas för att testa din implementation av CandyStore-klassen.