

Lab 2 – DP Strategy

Designmönster med C++

Syfte: Tillämpning av designmönstret Strategy

Uppgift

Dags att baka!

Du ska skriva ett program som motsvarar nedanstående beskrivning.

Klassen Bakery är en abstraktion av ett bageri som bakar kakor, bröd och annat mums-mums av olika sorter, beroende på vilka recept och råvaror som finns tillgängliga. Varje recept motsvaras av en klass som är deriverad från den abstrakta basklassen BakingRecepy.

<i>BakingRecepy</i>
#iName: string #neededIng: vector<Ingredient>
+BakingRecepy(aName:string) +bakeIt(): void +isBakeable(availableIngred:vector<Ingredient>&): bool +getName(): string

Operationen `bakeIt()` utför själva bakningen genom att ge en kort beskrivning hur man bakar en viss produkt. Beskrivningen behöver inte följa kokboks-standard...

Operationen `isBakeable()` används för att testa om just det här receptet kan bakas med de ingredienser som finns (parametern `availableIngred`).
`neededIng` innehåller uppgifter om nödvändiga ingredienser.

Det är endast de konkreta recepten som har kunskap om

- vilka ingredienser som behövs, och
- hur bakningen går till.

Alla ingredienser är instanser av klassen `Ingredient`:

```
class Ingredient {
public:
    Ingredient(string aName)
        :iName(aName) {}

    string getName() const { return iName; }
    operator string() const { return iName; }
    bool operator==(const Ingredient &rhs) const {
        return iName == rhs.iName;
    }
private:
    string iName;
};
```

Bakery har till sin hjälp ett `BakingRecepyManager`-objekt som tillhandahåller de olika recepten:

BakingRecepyManager
<code>+BakingRecepyManager(fileName:string)</code> <code>+hasAnotherRecepy(): bool</code> <code>+getNextBakingRecepy(): BakingRecepy*</code>

Vilka råvaror som finns tillgängliga för dagen bestäms av innehållet i textfilen `skafferi.dat`. Filnamnet ges till konstruktorn för `BakingRecepyManager` som läser filen för att se vilka råvaror som finns och avgör vilka recept som kan bakas.

Metoden `getNextBakingRecepy()` returnerar en ny `BakingRecepy` för varje anrop. `hasAnotherRecepy()` används för att testa om det finns ytterligare ett recept. Om `getNextBakingRecepy()` anropas när det inte finns något fler recept ska ett exception av typen `NoBakingException` kastas.

Bakery utför bakningen genom att anropa `bakeIt()` för recepten men använder använder sitt `BakingRecepyManager`-objekt för att få tillgång till recepten. Du behöver inte bekymra dig om hur mycket som finns av en viss råvara. Finns den med i skafferiet så räcker den till allt som ska bakas!

Förslag på recept:

- Pizza
kräver ingredienserna: oil, yeast , wheat-flour, salt
- Scones
kräver ingredienserna: oil, baking-powder, wheat-flour, salt, sugar, milk
- Kladdkaka
kräver ingredienserna: egg, sugar, baking-powder, wheat-flour, salt, marge, cocoa-powder
- Sockerkaka
kräver ingredienserna: egg, sugar, baking-powder, wheat-flour, marge, lemon

Redovisning

Zippad fil med QT-projekt eller VisualStudio-projekt. Kommenterad källkod med testprogram + `skafferi.dat`.