

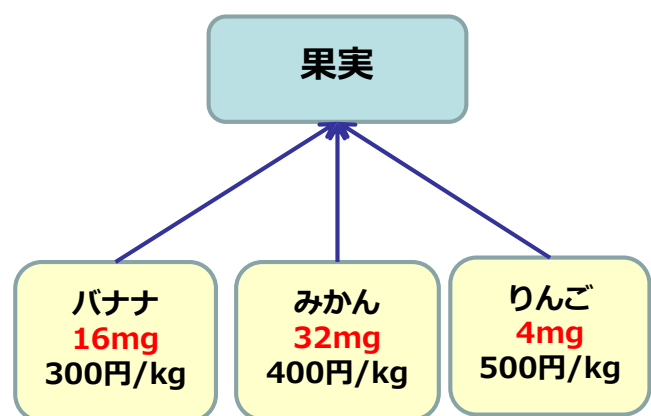
以下の内容を満たす Java プログラム群 (Fruit.java, Banana.java, Orange.java, Apple.java, Kadai3.java (ここに全体の main)), (その他、必要に応じて Ant 用の build.xml, input.txt) 等を含めてよい) を作成し、実行結果 (Kadai3.txt) とプログラム群を ZIP でまとめて、Moodle にアップせよ。メ切は 7 月 13 日 (火) の夜までとする。

以下の図に示すように、基底クラスを「果実クラス」(**Fruit クラス**)、派生クラスを「バナナクラス」(**Banana クラス**)、「みかんクラス」(**Orange クラス**)、「りんごクラス」(**Apple クラス**) とする クラスの継承例を授業資料の野菜クラスの例にならって作成せよ。この際、以下の条件を満たすこととする。

- (1) 基底の Fruit クラス (Fruit.java) には、double vitaminC, int price, int howMany を private 型で持たせる。これらをアクセスする get 関数は public 型とする。(public double getVitaminC(), public int getPrice(), public int getHowMany())
- (2) バナナ、みかん、りんご、をそれぞれ x (kg), y (kg), z (kg) (kg 単位で) 買えると仮定する。(ただし、 $x, y, z \geq 1$)
- (3) 派生クラスのオブジェクトを、基底クラスの変数 (Fruit クラスの変数) に代入し、個々の果実を代表させる Fruit クラスの変数を、線形リスト、配列、ArrayList 等で管理する。
- (4) プログラムに x, y, z (各フルーツ) の購入 kg 数 (整数値) をパラメータで与え、ビタミン C の総含有量(g)、総額をプリントせよ。なお、異なる 2 種類以上のパラメータで実行した結果を実行結果ファイルに書出すこと。

コメントとヒント:

Main プログラム (Kadai3.java とします) や、生成した果実のインスタンスを管理するプログラムは、授業資料のサンプルプログラムを適宜、参考にしてください。図の中の数値は、赤字がビタミン C の 100g あたりの量(mg)で、黒字は、1kg あたりの値段 (あくまで仮想的な値段で消費税はないとします) を表します。プログラムの実行では、たとえば、



```
$ javau Kadai3 x y z >> Kadai3.txt
```

のように x, y, z を与えて実行してください。>>>は、ファイル (ここでは Kadai3.txt) があれば、その末尾に追加するという意味です。なお、赤字のビタミン C の量は 100g

あたりの量ですが、たとえば **1kg** あたりのグラム数に変換した数値をクラス内のメンバー変数等に保持しておいて結構です。(4)のビタミンCの総含有量は **mg** でなく、**g** でプリントすることになりますので注意ください。

>> が使えない場合は、2回実行した結果を適宜連結したファイルを **Kadai3.txt** としてください。

実行例（雰囲気 1）

```
$ java Kadai3 x y z >> Kadai3.txt
```

```
*****
```

作成者：青野雅樹：01162069

日付：2021 年 7 月 2 日 11:04:50 JST

入力パラメータ：3 6 9

クラス継承

```
*****
```

<バナナ> 3 kg

<みかん> 6 kg

<りんご> 9 kg

ビタミンC総量 = xxx.xxxx (g)

値段 = xxxxx (円)

```
*****
```

作成者：青野雅樹：01162069

日付：2021 年 7 月 2 日 12:15:30 JST

入力パラメータ：6 12 2

クラス継承

```
*****
```

<バナナ> 6 kg

<みかん> 12 kg

<りんご> 2 kg

ビタミンC総量 = xxx.xxxx (g)

値段 = xxxxx (円)

実行例（雰囲気 2）『Ant の場合』

```
$ ant -e -Dargs="5 15 4" run >> Kadai3.txt
```

【出力は省略】