

プログラム設計 前期中間試験

※ 全てのプログラムファイルの先頭行に、コメントとして自分の番号と名前を書くこと。

準備 試験で利用可能なコードが事前に用意されている。端末内で、以下のコマンドを実行してコピーしておくこと。

```
$ cp /usr/local/common/kogai/201806pd.java . (←ここにピリオド)
```

- 1** int 型の配列 num に格納された整数値に対して、偶数であればその値を 0 にし、奇数であればそのまま値を残して、結果を出力するプログラムを作成しなさい。プログラムは、以下に示すコードに不足部分を追加すると完成させることができる。(コピーしたファイルから入手可能)

```
class Pd1mid1 {
    public static void main(String[] args) {
        int[] num = {3, 2, 17, 5, 4, 10};

        //----- ここに配列に対する処理を入れる -----

        for(int i=0; i<num.length; i++) {
            System.out.printf("num[%d]: %d  ", i, num[i]);
        }
        System.out.print("\n");
    }
}
```

この実行結果は以下のようになる。

```
[実行結果]
num[0]: 3   num[1]: 0   num[2]: 17   num[3]: 5   num[4]: 0   num[5]: 0
```

- 2** 次のような「食品のカロリー」を表すクラスを定義しなさい。

- クラス名: Food
- フィールド:
 - String 型の name (食品名)
 - int 型の calorie (カロリー (kcal))
 - int 型の amount (量 (g))
- メソッド:

```
public Food()
    //クラス Food のコンストラクタ
    //フィールド name に"food"、calorie に 50、amount に 100 を代入する
```

```
public void setInfo(String n)
    //フィールド name に仮引数 n を代入する
```

```
public void show(String str)
    //まず、仮引数 str を「---str---」の形で出力し、
    //フィールド name, calorie, amount の値と単位がわかるように出力する
    //(出力の様子は以下に示す実行結果を参照)
```

main の処理とその実行結果は以下のようになる。(コピーしたファイルから入手可能)

```
[main を持ったクラス]
```

```
class Pd1mid2 {  
    public static void main(String[] args) {  
        Food f1 = new Food();  
        f1.show("f1");  
        f1.setInfo("オレンジ");  
        f1.show("setInfo 後");  
    }  
}
```

[実行結果]

---f1---

食品名: food

カロリー: 50kcal

量: 100g

---setInfo 後---

食品名: オレンジ

カロリー: 50kcal

量: 100g

- 3 クラス Food に、次のようなメソッド（とコンストラクタ）をオーバーロードしなさい。

```
public Food(String n)  
    //フィールド name に n、calorie を 200、amount に 100 を代入する
```

```
public void setInfo(String n, int c)  
    //フィールド name に n、calorie に c を代入する  
    //その他のフィールドは変更されない
```

```
public void setInfo(int a)  
    //フィールド calorie を、a グラムの場合に計算し直す（a と amount の比率を calorie に掛ければ求まる）  
    //上記の計算は int 型同士の計算で構わない（つまり double 型などにキャストする必要はない）  
    //フィールド amount に a を代入する  
    //その他のフィールドは変更されない
```

main の処理とその実行結果は以下のようになる。（コピーしたファイルから入手可能）

[main を持ったクラス]

```
class Pd1mid3 {  
    public static void main(String[] args) {  
        Food f2 = new Food("おにぎり");  
        f2.show("f2");  
        f2.setInfo("唐揚げ", 313);  
        f2.show("setInfo で name と calorie を変更");  
        f2.setInfo(300);  
        f2.show("setInfo で amount を変更");  
    }  
}
```

[実行結果]

---f2---

食品名: おにぎり

カロリー: 200kcal

量: 100g

---setInfo で name と calorie を変更---

食品名: 唐揚げ

カロリー: 313kcal

量: 100g

```

---setInfo で amount を変更---
食品名: 唐揚げ
カロリー: 939kcal
量: 300g

```

- 4 「仮引数で渡されたクラス Food のインスタンスの情報を、自身のフィールドに結合・加算する」メソッド add を追加しなさい。

```

public void add(Food f)
//仮引数 f が参照しているインスタンスを、自身のインスタンスに以下のように結合・加算する
//自身のフィールド name と、仮引数 f が参照しているインスタンスの name とを、カンマ (", ") を間に挟んで結合する
//自身のフィールド calorie に、仮引数 f が参照しているインスタンスの calorie を加算する
//自身のフィールド amount に、仮引数 f が参照しているインスタンスの amount を加算する

```

main の処理とその実行結果は以下のようになる。(コピーしたファイルから入手可能)

```

[main を持ったクラス]
class Pd1mid4 {
    public static void main(String[] args) {
        Food f3 = new Food();
        Food f4 = new Food();
        f3.setInfo("ハンバーグ", 223);
        f4.setInfo("えびフライ", 292);
        f3.add(f4);
        f3.show("add 後の f3");
        f4.show("add 後の f4");
    }
}

```

```

[実行結果]
---add 後の f3---
食品名: ハンバーグ, えびフライ
カロリー: 515kcal
量: 200g
---add 後の f4---
食品名: えびフライ
カロリー: 292kcal
量: 100g

```

- 5 クラス Food に「賞味期限」を表す Date 型のフィールド shomi を追加して、以下のメソッドを shomi に対応するように変更し、新規にメソッド setShomi を追加しなさい。(変更内容については、以下のコメントを参照)
なお、クラス Date の定義はコピーしたファイルから入手可能である。

```

public Food()
public Food(String n)
//フィールド shomi に対して、クラス Date のインスタンスを作成する (引数なしコンストラクタでよい)

```

```

public void show(String str)
//食品情報の出力の後に、shomi のメソッド show で賞味期限の情報を出力する

```

```

public void setShomi(Date d)
//フィールド shomi が仮引数 d で渡されたインスタンスを参照する

```

main の処理とその実行結果は以下のようになる。(コピーしたファイルから入手可能)

```

[main を持ったクラス]

```

```
class Pd1mid5 {  
    public static void main(String[] args) {  
        Date d1; Food f5, f6;  
        f5 = new Food();  
        f5.show("f5");  
        f6 = new Food("食パン");  
        d1 = new Date();  
        f6.show("f6");  
        d1.setYMD(2018, 6, 14);  
        f6.setShomi(d1);  
        f6.show("setShomi 後");  
    }  
}
```

[実行結果]

---f5---

食品名: food
カロリー: 50kcal
量: 100g
賞味期限: 1970/ 1/ 1

---f6---

食品名: 食パン
カロリー: 200kcal
量: 100g
賞味期限: 1970/ 1/ 1

---setShomi 後---

食品名: 食パン
カロリー: 200kcal
量: 100g
賞味期限: 2018/ 6/14

問題はここまで

1～5 各 20 点

定期試験の実施について

試験中に使用できるもの

- 筆記用具（メモ用紙は必要な人に配布）
- 演習室のコンピューター一台（一つの机に一人の配置）

試験中に参照できるもの

- 自分のホームディレクトリ（ホームフォルダ）以下に保存されているファイル
- * 上記以外の情報を参照することは不正行為とする
(例：USB で接続された機器に保存されているファイルの参照など)
- * 試験中は、演習室外へのネットワークアクセスは遮断される

答案の提出

- 提出する全てのプログラムファイルの先頭行に、自分の学科の出席番号と氏名をコメントとして書く
- 保存したファイルは次のように「report」コマンドで提出する
(ちゃんと提出できた場合は、「Succeed.」と画面に表示される)
\$ ~kogai/report pd1mid 「プログラムファイル」
- 複数のファイルを提出する場合は、report コマンドを分けて提出する
例えば、test1.java と test2.java のファイルを提出したい場合は、次のように 2 回に分けて提出する
\$ ~kogai/report pd1mid test1.java
\$ ~kogai/report pd1mid test2.java
- 同じ問題に対して、複数の提出ファイルが存在した場合は、更新日時が新しい方を提出ファイルとする

前期中間試験 模範解答 (平均 86.8 点)

採点について コンパイル時にエラーとなる箇所は -4 点, 実行可能だが処理内容が問題の意図と違う箇所は -2 点を基本とする。

配点: 1 ~ 5 各 20 点

```
class Pd1mid1 {
    public static void main(String[] args) {
        int[] num = {3, 2, 17, 5, 4, 10};

        for(int i=0; i<num.length; i++) {
            if(num[i] % 2 == 0) {
                num[i] = 0;
            }
        }
        for(int i=0; i<num.length; i++) {
            System.out.printf("num[%d]: %d ",
                               i, num[i]);
        }
        System.out.print("\n");
    }
}

class Food {
    private String name;
    private int calorie;
    private int amount;
    private Date shomi;
    public Food() {
        name = "food";
        calorie = 50;
        amount = 100;
        shomi = new Date();
    }
    public Food(String n) {
        name = n;
        calorie = 200;
        amount = 100;
        shomi = new Date();
    }
    public void show(String str) {
        System.out.println("----" + str + "----");
        System.out.println("食品名: " + name);
        System.out.println("カロリー: "
                           + calorie + "kcal");
        System.out.println("量: " + amount + "g");
        System.out.print("賞味期限: "); shomi.show();
    }
    public void setInfo(String n) {
        name = n;
    }
    public void setInfo(String n, int c) {
        name = n;
        calorie = c;
    }
    public void setInfo(int a) {
        calorie = a / amount * calorie;
        amount = a;
    }
    public void add(Food f) {
        this.name += ", " + f.name;
        this.calorie += f.calorie;
        this.amount += f.amount;
    }
    public void setShomi(Date d) {
        shomi = d;
    }
}

class Date {
    private int year;
```

```
        private int month;
        private int day;
        public Date() {
            year = 1970; month = 1; day = 1;
        }
        public Date(int y, int m, int d) {
            year = y; month = m; day = d;
        }
        public Date(Date d) {
            this(d.year, d.month, d.day);
        }
        public void setYMD(int y, int m, int d) {
            year = y; month = m; day = d;
        }
        public void show() {
            System.out.printf("%4d/%2d/%2d\n",
                               year, month, day);
        }
    }

    class Pd1mid2 {
        public static void main(String[] args) {
            Food f1 = new Food();
            f1.show("f1");
            f1.setInfo("オレンジ");
            f1.show("setInfo 後");
        }
    }

    class Pd1mid3 {
        public static void main(String[] args) {
            Food f2 = new Food("おにぎり");
            f2.show("f2");
            f2.setInfo("唐揚げ", 313);
            f2.show("setInfo で name と calorie を変更");
            f2.setInfo(300);
            f2.show("setInfo で amount を変更");
        }
    }

    class Pd1mid4 {
        public static void main(String[] args) {
            Food f3 = new Food();
            Food f4 = new Food();
            f3.setInfo("ハンバーグ", 223);
            f4.setInfo("えびフライ", 292);
            f3.add(f4);
            f3.show("add 後の f3");
            f4.show("add 後の f4");
        }
    }

    class Pd1mid5 {
        public static void main(String[] args) {
            Date d1; Food f5, f6;
            f5 = new Food();
            f5.show("f5");
            f6 = new Food("食パン");
            d1 = new Date();
            f6.show("f6");

            d1.setYMD(2018, 6, 14);
            f6.setShomi(d1);
            f6.show("setShomi 後");
        }
    }
}
```