

# 第 50 回検定（平成 26 年 1 月 19 日）全商情処プロ部門 Java 解説

## 2 級【7】

### 解答

(1)	<code>kiroku &lt;= 430</code>
(2)	<code>maisu * kinsyu[n]</code>
(3)	<code>n = n + 1</code> (別解 1 <code>n++</code> 別解 2 <code>n += 1</code> )
(4)	<code>tcode[p] != acode</code>
(5)	<code>(ryo[p] - wari) * nin</code>

問 1 は、最小値を求めるアルゴリズムである。基本的な内容であり比較的容易である。

問 2 は、**金種計算のアルゴリズム**である。事前にアルゴリズムを学んで知っていなければ難問に感じるであろう。

問 3 は、線形探索や一次元配列の問題である。基本的な内容だが問いになっている個所が難しい。2 級全般を通して前回検定より難しくなっている。5 問中配列を答えさせるものが 3 つあり、出題に対してのある一定の偏りを感じざるをえない。検定対策における配列の学習が重要となる。

問 1 は、最小値を求めるアルゴリズムのプログラムであるが、(1) はそれとは無関係な個所であり容易に解くことができる。(1) の条件を満たす場合は「[判定]予選通過」と表示されることから、処理条件 2 が答えだということが解る。「<=」を「≦」と書かないこと。

問 2 は、金種計算のアルゴリズムであり、計算には剰余を用いる。その個所が (2) であり、金種計算のアルゴリズムでは剰余を用いる、ということを知らなければ解くのはかなり困難だろう。対して (3) は変数 `n` をインクリメント (=1 増やす) するだけである。これは、この外の `while` で `n` を使用していることに気づけば容易に回答できる。

### 金種計算のアルゴリズム

`kingaku` が 168530 の場合を考える。168,530 なので、普通に自分の頭の中でお金を計算すると、1 万円札が 16 枚、5 千円札 1 枚、1 千円札 3 枚、5 百円玉 1 枚、10 円玉 3 枚、ということがわかる。これをコンピュータに計算させるには、まず 168530 を 1 万円札 (10000) で割る。(プログラムの `maisu = kingaku / kinsyu[n]` の個所) その答えが 1 万円札の必要枚数の 16 である。残りの金額を計算するためには 160000 をマイナスして 8530 を求める必要がある。その個所が (2) である。これは言い換えれば、168530 を 10000 で割った余りを計算することになる。この、割り算のあまりの事を剰余という。剰余の 8530 を求めると、今度は 5 千円札 (5000) で割って、5 千円札の枚数を計算する…という手順を繰り返す。

## **public KinsyuKeisan()**

引数がなくクラス名と同じ、ということでコンストラクタだとわかる。コンストラクタはインスタンス生成時に必ず呼び出されるものである。つまり、KinsyuKeisan インスタンス生成時には、必ずキーボードから何か入力しなければならない、ということである。

## **KinsyuKeisan keisan = new KinsyuKeisan();**

KinsyuKeisan クラスの main メソッドの中で、自分自身をインスタンス化している、というものである。こうすることにより、インスタンスメソッドである、input() や output() を呼び出すことができるようになる。

(4) は、線形探索である。流れ図でも同様だが、線形探索の場合はループの中は変数をインクリメントする命令文 1 つとなる。acode を配列 tcode の先頭 (0 番目) から順々に探していく。0 番目の次は 1 番目、その次は 2 番目、3 番目…と増加させていくので、この添字に p を使用することが分かる。また、while は継続条件を書く必要がある。「(4) になるまで繰り返す」ではなく「(4) の間は繰り返す」である。つまり、「acode が見つかるまで繰り返す」と考えるのではなく「acode が見つからない間は繰り返す」と考える必要がある。

(5) は、処理条件 4 から比較的容易に回答できる。他の難しい個所に惑わされることなく、このような回答し易い問いで確実に得点を取ることが合格へのカギとなる。

(4) では、配列 tcode に存在しないデータが acode として入力された場合 tcode とデータが一致せず、p の値は配列の最後の 9 番目を通り越して 10 番目となり、ArrayIndexOutOfBoundsException というエラーになる。これを回避するには番兵を用いるか、p が 9 を超えた場合にループを抜けるような処理が必要となる。もしくは問題文に「入力データに誤りはないものとする」という一文が必要になるであろう。