Ｊａｖａプログラミング Ⅰ

９回目　while、do while 文　課題

１．次の for 文を while 文で書き換えなさい。ここで i は int 型の変数とする。
ヒント：for 文の初期化の式と更新の式は、それらが実行されるタイミングから、while 文ではどこに書けばよいのか考えてみましょう

```
for(i=10; i>=0; i--)
{
    System.out.println("カウントダウン"+i);
}
```

２．次のようにある数値の 2 進数表現の各桁を入力して 10 進数に変換するコードを作成しなさい。入力の終わりは 0 / 1 以外を入力するものとする。

（実行例）
2 進数の 1 桁目を入力してください
[終了するには、0 / 1 以外を入力してください]
1 
2 進数の 2 桁目を入力してください
[終了するには、0 / 1 以外を入力してください]
0 
2 進数の 3 桁目を入力してください
[終了するには、0 / 1 以外を入力してください]
1 
2 進数の 4 桁目を入力してください
[終了するには、0 / 1 以外を入力してください]
2 
入力された 2 進数は 10 進数で 5 です。

ヒント：
2 進数を 10 進数に変換するには、2 進数の各 i (i>=1) 桁の値 (0 または 1) に乗数 $2^{(i-1)}$ を掛け算し、すべての桁についてこれらを合計すればよい。

実行例のように 2 進数の 1 桁目から順番に入力を行う場合、入力された 2 進数と乗数を掛け算して累積していけばよい。

乗数 $2^{(i-1)}$ を作るためには、ある変数を 1 で初期化して繰り返しの度に 2 倍してやればよい。

３．次のプログラムは九九の問題を順次に出力してユーザが解答する九九の計算ドリルプログラムです。ユーザが入力した解答が間違っていた場合に正解するまで繰り返すようにします。空欄を do while 文で適切に埋めてプログラムを完成させなさい。

（ソースプログラム）

```
import java.io.*;
public class Assignment9_3
{
    public static void main(String[] args) throws IOException
    {
        BufferedReader br;
        br = new BufferedReader(new InputStreamReader(System.in,"Shift-JIS"));

        int i, j, ans;
        for(i=1; i<=9; i++)
        {
            for(j=1; j<=9; j++)
            {
                ┌──────────────────────────────┐
                │                              │
                └──────────────────────────────┘
                    System.out.print(i+"×"+j+"は？");
                    ans = Integer.parseInt(br.readLine());
                ┌──────────────────────────────┐
                │                              │
                └──────────────────────────────┘
            }
        }
    }
}
```

４．（３．）のプログラムにおいて、ユーザが間違えた問題をその後 4 回連続して不正解した場合（計 5 回）は「次の問題へいきます。」と表示して次の問題へ移るように改良しなさい。
ヒント：新しい変数を宣言し間違えた回数を数えながら if 文と break 文を用いましょう

（実行例）

```
  :
2×2 は？4
2×3 は？5
2×3 は？1
2×3 は？4
2×3 は？7
2×3 は？8
次の問題へいきます。
2×4 は？
```

５．（４．）のプログラムにおいて同じ値同士の掛け算（1×1、2×2、・・・）はスキップするように改良しなさい。
ヒント：if 文と continue 文を上手に用いましょう

６．関数 f(x)=x²-x-1 の最小値（頂点）を最急降下法により求めなさい。

（最急降下法）
関数f(x)においてxの値を次の規則により更新し徐々にf(x)を最小値に近づける手法です。

規則　$x^{(k+1)} \leftarrow x^{(k)} - c * \frac{df(x^{(k)})}{dx}$

ここで、パラメータcはステップサイズであり 1 回の反復での値の変化量を決定します。本課題ではxの初期値$x^{(0)}$を 1.0、ステップサイズcを 0.25 とします。

ヒント：while 文を用いてxの変化量がある一定値（閾値）以下になるとき更新の繰り返しを終了します。そのときの座標(x, f(x))を最小値として出力します。本課題では閾値を 0.001 とします。

７．while 文を用いて、2×2×…を次々と計算し、その累乗が 1000 を超えるまで画面に出力するコードを書きなさい。

（実行例）
1 回目：2 を掛けます
現在の累乗は 2 です
2 回目：2 を掛けます
現在の累乗は 4 です
3 回目：2 を掛けます
現在の累乗は 8 です
　　⋮
△回目：2 を掛けます
現在の累乗はooo です
累乗が 1000 を超えました

８．ある計算問題を出力しなさい。その後、キーボードから答えを入力させ、正解するまで入力を促すコードを作成しなさい。
ヒント：do while 文をうまく用いましょう。

（実行例）
偶数でかつ 3 で割り切れる整数を入力してください
4
偶数でかつ 3 で割り切れる整数を入力してください
6
正解です♪

９．素数を 2,3,5,…と 100 個まで順番に表示するコードを作成しなさい。素数とは 2 以上の自然数であり、1 と自分以外では割り切れない数値です。たとえば、5 は 1 と 5 以外では割り切れないので素数です。

ヒント：

・繰返し文を 2 重に使います。

・外側の繰返し文で、素数かどうかをチェックしたい数 n を 2 から順番に増やしていきます

・内側の繰返し文で、数 n が素数かどうか判断するために、除数 m を 2 から n まで順番に増やしていきながら、n が m で割り切れるかをチェックします。もし、m が n と等しくなるまでに割り切れてしまえば、n は素数ではありません。

（実行例）

```
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103
107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 211
223 227 229 233 239 241 251 257 263 269 271 277 281 283 293 307 311 313 317 331
337 347 349 353 359 367 373 379 383 389 397 401 409 419 421 431 433 439 443 449
457 461 463 467 479 487 491 499 503 509 521 523 541
```

１０．数当てゲームを作成しなさい。まず、コンピュータが 1～99 までの整数をランダムに決めます。次に、予想をキーボードから入力し、当たるまで繰返します。正解と異なる場合は、「小さいです」または「大きいです」のメッセージを出力します。

ヒント：

1～99 の乱数は次のようにして得ることができます。

```
int num;
num=(int)(1+99*Math.random());
```

これで、変数 num に 1～99 までのいずれかの整数が代入されます。

※ Math.random()の部分は Java プログラミング 2 の講義で扱います

（実行例）

```
★数当てゲーム★
1 回目の予想(1～99)を入力してください
50
大きいです
2 回目の予想(1～99)を入力してください
25
小さいです
3 回目の予想(1～99)を入力してください
37
小さいです
4 回目の予想(1～99)を入力してください
44
4 回目で正解です♪
```

１１．サイコロを振り 1 の目がでる確率（0.1666…）を実験的に求めなさい。まず、コンピュータが 1～6 までの整数をランダムに決めます。次に、1 の目が出たらカウントします。これを繰り返します。繰り返しの毎に、1 の目が出た回数を繰り返し回数で割り、確率を求めます。この求めた確率と 1 つ前の繰り返しで求めた確率との差の絶対値が 0.00000001 未満になるまで繰り返しを続けます。ただし、最初の 20 回は必ず繰り返します。

ヒント１：
1～6 の乱数は次のようにして得ることができます。

```
int num;
num=(int)(1+6*Math.random());
```

これで、変数 num に 1～6 までのいずれかの整数が代入されます。
※ Math.random()の部分は Java プログラミング 2 の講義で扱います

ヒント２：
変数 a の絶対値は次のようにして求めます。

```
if(a<0)a*=-1;
```

ヒント３：
おおよそ以下のアルゴリズムになります。

1. 1~6 の乱数 number〔int 型〕を取得
2. 繰り返し回数 loop〔int 型〕をカウントアップ
3. 1 が出たら、1 が出た回数 count〔int 型〕をカウントアップ
4. loop と count を用いて 1 が出た確率 p〔double 型〕を計算
5. 1 つ前との確率 p_bak〔double 型〕の差 p_bak-p が 0.00000001 以上または繰り返し回数 loop が 20 以下であれば 1.に戻る

（実行例）
16677156 回サイコロを振りました
1 の目が出る確率は約 0.16677154066316824 です
※結果が 0.1666…と近ければ実験成功です