

プログラミング基礎

第2回

int型

- 整数を扱うint型では,
-2147483648~2147483647($-2^{31} \sim 2^{31}-1$)
までの値が記憶できる
- では, それらの範囲を超えた値を記憶しようとするとならうか？

オーバーフロー

- Javaでは, 2147483647に+1すると,
-2147483648になる
⇒オーバーフロー

-2147483648に-1すると
2147483647になる
⇒アンダーフロー

(あまり使わない言葉, 世間的にはひっくるめてオーバーフローと呼んでしまうことが多い)

オーバーフロー

- 数字を扱う型において、記憶できる値の範囲を越えること
⇒オーバーフロー

課題

- わざとプログラム内でオーバーフローもしくはアンダーフローを起こせ (Overflow.java)
- 最小, 最大値は直接入力しなくても
Integer.MIN_VALUE, Integer.MAX_VALUEで得られる

オーバーフロー&アンダーフロー

- Javaではどのような現象か分かっているが...
- 基本的にはバグ(起こしてはいけない)
(わざと利用する場合もあるが、
プログラムの読みやすさを下げる)
- 計算結果の値が予想と極端に異なる場合は、
疑うべき要因

関連：2038年問題

- 標準的なC言語では、時刻を
1970年1月1日0時0分0秒
からの経過秒数で時刻を表現する

その秒数を表す型、time_t型はJavaで言うint型なので2147483647秒までしかカウントできない

⇒ 2147483647秒経過する
2038年1月19日3時14分7秒
を越えた時点でオーバーフローしてしまい、
プログラムからすると
経過秒数が -2147483648秒、
⇒1901年12月13日20時45分52秒になる

同じような原因で「2036年問題」も存在

関連：2038年問題

- 実は,

経過時間を内部で2倍する処理があったために、一足先にオーバーフローし、2004年にATMが正常動作しなくなる事例が既に発生

IT Pro【スクープ】コンピュータの“西暦2038年問題”発生、早くも日本を揺るがす
<http://itpro.nikkeibp.co.jp/free/NC/NEWS/20040202/139212/>

double型

- 小数点を含む値を扱うための型
- double型 : メモリを64bit (8byte) 使用
- float型 : メモリを32bit (4byte) 使用

精度が悪くなるがメモリ使用量が少なくなるため、現在でもゲーム開発では3次元計算などにおいてfloat型を使うのが慣例

double型

- double型では2進数でどのように値を表現しているか

⇒IEEE 754という規格に沿って行われる

http://ja.wikipedia.org/wiki/IEEE_754

書式付出力

- 数字を桁数を指定して表示：
Web資料の通り

intとdoubleの併用

```
int x = 10;  
double y;  
double a = 10.5;  
int b;
```

```
y = x;
```

⇒OK: 自動的にdoubleへ変換される(データが失われないから)

```
b = a;
```

⇒NG: 自動的にintへは変換できない
(小数点以下が失われてしまうから)

```
b = (int)a;
```

OK⇒小数点を切り捨ててint型にすることを明示的に記述する

「ある型の変数を他の型に変換する」⇒「キャスト」と呼ばれる

intとdoubleの併用

- 異なる型同士で演算を行うと、結果は、大きい値を記憶できる方の型になる
(小さい方の型ではデータが失われてしまう可能性があるため)

```
int x = 10;
```

```
double y = 2.5;
```

```
double a;
```

```
int b;
```

```
a = x * y; //OK: int * doubleの結果はdouble
```

```
b = x * y; //NG
```

```
b = (int)(x * y) //OK
```

課題

- 先週のコマンドライン引数を使って, 値段を入力し, 入力された値段の消費税込みの値段を求めて表示せよ(Tax.java)

```
C:\¥java>java Tax 5000  
税込 : 5250円
```

誤差

- double(float)型は2進数で小数を表現するため、正確な値を表現できない場合がある(むしろ大抵できない)

2進数	10進数
0.1	0.5
0.01	0.25
0.001	0.125
0.0001	0.0625
...	...

誤差

- 「double(float)型の値がxである場合」は大抵正しく動かない

```
double x;
```

```
x = いろいろ計算;
```

```
if (x == 10.65) { ... }
```



誤差

- double(float)型における値の表現方法や誤差の話は非常に奥が深い話です(色々な話題がある)

興味を持った方は以下の書籍や, Webを

- 川上一郎 ”理工系の数学入門コース8 数値計算” 岩波書店
- 吉田武 ”オイラーの贈り物 新装版” 東海大学出版局
- 平山尚 ”ゲームプログラマになる前に覚えておきたい技術”
秀和システム
- 日経PC21 / 演算誤差の正体 - IEEE 754 浮動小数点数の仕組み
<http://pc.nikkeibp.co.jp/pc21/special/gosa/eg4.shtml>

補足

- Advancedの問題に出てくる「ニュートン法」

⇒「数値計算」と呼ばれる分野の手法

- ”なぜこれで求まるのか”知りたい場合は以下書籍などを参照

川上一郎 ”理工系の数学入門コース8
数値計算” 岩波書店