

Aufgabe 1 Java PRP2

Teil 1 Von Ruby zu Java

Übersetzen Sie das Ruby Programm für Figuren (liegt der Aufgabe bei) in ein lauffähiges Java-Programm. Es dient als einfacher Einstieg in die Programmiersprache Java. Bringen Sie sich nicht um die einmalige Chance, diesen Einstieg zu verpassen, will heißen, kopieren Sie die Lösung weder aus dem Script noch aus anderen Quellen.

Teil 2 Basisdatentypen und Operatoren und Kontrollstrukturen

A_ Wertebereiche von int

Welche Ergebnisse liefern die nachfolgenden Operationen, wenn **max** wie folgt definiert ist?

Erklären Sie die Ergebnisse.

Schreiben Sie für **min** arithmetische Ausdrücke, die ein ähnliches Phänomen wie für max zeigen.

```
int max = Integer.MAX_VALUE;
int min = Integer.MIN_VALUE;

System.out.println("Binaerdarstellung Integer MAX_VALUE:"
    + Integer.toBinaryString(max));
System.out.println("Integer MIN_VALUE:" + min);
System.out.println("Binaerdarstellung Integer MIN_VALUE:"
    + Integer.toBinaryString(min));

System.out.println("Integer MAX_VALUE+1:" + (max + 1));
System.out.println("2-Integer MAX_VALUE" + (2 - max));
System.out.println("-1-Integer MAX_VALUE:" + (-1 - max));
System.out.println("-2-Integer MAX_VALUE:" + (-1 - max));
System.out.println("2* Integer MAX_VALUE:" + (2 * max));
System.out.println("4* Integer MAX_VALUE:" + (4 * max));
```

B_ Operationen mit double Überlauf und Unterlauf

Welche Ergebnisse liefern die nachfolgenden Operationen?

Erklären Sie die Ergebnisse.

```
double d = 3.14159;
System.out.println("d:" + d);
System.out.println("d+1:" + d+1);
System.out.println("8/(int)d:" + 8/(int)d);
System.out.println("8/d:" + 8/d);
System.out.println("(int)(8/d) : " + (int)(8/d));
System.out.println("8.0/0.0:" + 8.0/0.0);
System.out.println("0.0/0.0:" + 0.0/0.0);
d = 1e308;
System.out.println(d + "*10==" + d * 10);
```

```

d = 1e-305 * Math.PI;
System.out.print("d:" + d + "\n ");
for (int i = 0; i < 4; i++)
    System.out.print(" " + (d /= 100000));

System.out.println();
for (int i = 0; i < 100; i++) {
    double z = 1.0 / i;
    if (z * i != 1.0)
        System.out.print(" " + i);
}

System.out.println();
System.out.println(sqrt(2)*sqrt(2)==2);

```

C_Zufallszahlen

Schreiben Sie eine Programm, das 5 gleichverteilte Zufallszahlen zwischen 0 und 1 berechnet und die maximale, minimale und den Mittelwert der Zufallszahlen ausgibt

D_Schleifenvariablen

Gegeben folgende Deklarationen und Schleifen.

Welchen Wert haben i, j nach dem jeweiligen Schleifendurchlauf?

Erklären Sie die Ergebnisse.

```

int i, j;
for (i = 0, j = 0; i < 10; i++) j += i;
for (i = 0, j = 1; i < 10; i++) j += j;
for (j = 0; j < 10; j++) j += j;
for (i = 0, j = 0; i < 10; i++) {
    j += j++;
}

```

E_FunktionenWachstum

Schreiben Sie ein Programm FunktionenWachstum, das für $n = 32, 64, \dots, 2048$ die Werte $\log(n)$, n , $n \cdot \log(n)$, n^2 , n^3 , und 2^n tabellarisch ausgibt. Verwenden Sie für die Ausgabe das Tabulatorzeichen ($\backslash t$) und formatieren Sie die Spalten mit den Gleitkommazahlen angemessen.

Ergebnisbeispiel:

2,7725887222e+00	16	256	4096	65536.0
3,4657359028e+00	32	1024	32768	4.294967296E9
4,1588830834e+00	64	4096	262144	1.8446744073709552E19
4,8520302639e+00	128	16384	2097152	3.4028236692093846E38
5,5451774445e+00	256	65536	16777216	1.157920892373162E77
6,2383246250e+00	512	262144	134217728	1.3407807929942597E154
6,9314718056e+00	1024	1048576	1073741824	Infinity
7,6246189862e+00	2048	4194304	8589934592	Infinity

F_IntervallZerlegen

Schreiben Sie ein Programm, das die ganzen Zahlen von 1000-2000 mit 5 Zahlen pro Zeile ausgibt. Das Programm darf nur eine **for**-Schleife und eine **if**-Anweisung enthalten. Verwenden Sie den % Operator.

Ergebnisbeispiel

```
1000 1001 1002 1003 1004
1005 1006 1007 1008 1009
1010 1011 1012 1013 1014
... // hier stehen selbstredend die nachfolgenden Zahlen bis 1985

1985 1986 1987 1988 1989
1990 1991 1992 1993 1994
1995 1996 1997 1998 1999
```

G_Potenzen

Schreiben Sie ein Programm Potenzen, das für einen Integer *n* alle positiven Potenzen von *n* im Java-Typ **long** ausgibt. Hinweis: **Long.MAX_VALUE** definiert den maximal möglichen positiven **long** Wert. Achten Sie darauf, dass Sie diese Grenze nicht verletzen und vermeiden Sie die Effekte der Wertebereichsüberschreitung, die Sie in Aufgaben-Teil 2 a kennengelernt haben.

Beispielausgabe für n=37

```
37^0=1
37^1=37
37^2=1369
37^3=50653
37^4=1874161
37^5=69343957
37^6=2565726409
37^7=94931877133
37^8=3512479453921
37^9=129961739795077
37^10=4808584372417849
37^11=177917621779460413
37^12=6582952005840035281
Überlauf bei 37^13 (= 3761551257857134389)

Long.MAX_VALUE 9223372036854775807
```

H_Zahlenumwandlung

Schreiben Sie ein Programm, das einen Integer *n* in eine Zeichenkette zur Basis *k* umwandelt. Verwenden Sie zur Darstellung der Ziffern im Zahlensystem der Basis *k* >10 Buchstaben beginnend mit A.

Hinweis: Das Verfahren zur Umwandlung zur Basis 2 wird in der Vorlesung gezeigt.

Hinweis: Für die gängigen Zahlendarstellungen können Sie Ihre Lösung mit den Methoden von **Integer** (**toBinaryString**, **toOctalString**, **toHexString**) überprüfen.

G_Reihen trigonometrischer Funktionen

Schreiben Sie Programme **SinReihe** und **CosReihe**, die für eine Zahl x (double) den **sin / cos** Wert solange nähert, bis sich der Näherungswert in der Berechnung nicht mehr ändert.

Geben Sie den Näherungswert und die Anzahl der Iterationen aus, die für die Berechnung verwendet wurden.

Entwickeln Sie für beide Näherungen effiziente Lösungen, die mit einer Schleife arbeiten und die Zwischenergebnisse in geeigneten Akkumulatoren vorhalten.

Hinweis: für die Exponentialfunktion e^x wird das Vorgehen in der Vorlesung erläutert.

Reihendarstellung für sin:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$$

Reihendarstellung für cos:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots$$