ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecturer: Prof. B. Gärtner

Informatik für Mathematiker und Physiker HS15

Exercise Sheet 5

Submission deadline: 15:15 - Tuesday 20th October, 2015
Course URL: http://lec.inf.ethz.ch/ifmp/2015/

Assignment 1 (4 points)

a) Compute the exact binary value of the decimal value 0.3

b) Compute the exact binary value of the decimal value 11.7

c) Assuming IEEE standard 754, what is the value of a, and why?

```
float x = 1.9f;
unsigned int a = x;
```

d) Assuming IEEE standard 754, what is the value of b, and why?

```
float y = 0.2f;
bool b = 5 * y == 1.0f;
```

e) Assuming IEEE standard 754, what is the problem with the following loop?

```
for (float i = 0.0f; i < 100000000.0f; ++i)
   std::cout << i << "\n";
```

Assignment 2 (4 points)

Write a C++ program point_on_line.cpp that determines if a point (x, y) is on a line g or not. The line g is defined as: $g(x) = 2.1 \cdot x + 0.5$. Use floating point variables of type double to do the calculation. Furthermore, since the computer might have to round the floating point numbers, you can not just compare them for equality using ==. Therefore, when you want to compare two floating point numbers, treat them as equal as soon as their difference[1] is less than 0.000001.

Judge Examples (Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)

```
x coordinate of point =? 0.25
y coordinate of point =? 3.91
Point is not on line
```

[1]Remark: This leads to an absolute tolerance. When the numbers were large, this approach would no longer be suitable. But for simplicity, this exercise can be solved with an absolute tolerance.

```
x coordinate of point =? -110
y coordinate of point =? -230.5
Point is on line
```

Submission: https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15052

Assignment 3 – Skript-Aufgabe 90 (4 points)

a) Implement the following function. You may assume that the type `double` complies with the IEEE standard 754. The function is only required to work correctly, if the nearest integer is in the value range of the type `int`.

```
// PRE: x is roundable to a number in the value range of type int
// POST: return value is the integer nearest to x, or the one closer
//       to 0 if x lies right in between two integers.
int round (double x);
```

b) Write a program `round.cpp` which inputs a number of type `double` from the user, then rounds this number using your function from a), and then outputs the rounded number.

Judge Examples (Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)

```
Input number to be rounded =? 2.7
The rounded number is 3
```

```
Input number to be rounded =? 5.5
The rounded number is 5
```

```
Input number to be rounded =? -3.6
The rounded number is -4
```

Submission: https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15053

Assignment 4 – Skript-Aufgabe 67 (4 points)

Write a program `dec2float.cpp` that outputs for a given decimal input number x, $0 < x < 2$, its normalized `float` value on your platform. The output should contain the (binary) digits of the significand, starting with 1, and the (decimal) exponent. You may assume that the floating point number system underlying the type `float` has base $\beta = 2$.

Hint: If you'd like to have more hints, you can find here a code skeleton outlining the structure of a possible solution: http://lec.inf.ethz.ch/ifmp/2015/u5/dec2float_skeleton.cpp

Judge Examples (Explanation: http://lec.inf.ethz.ch/ifmp/2015/judge_boxes.html)

```
Decimal number x (0 < x < 2) =? 0.125
Significand: 1
Exponent: -3
```

```
Decimal number x (0 < x < 2) =? 1.1
Significand: 100011001100110011001101
Exponent: 0
```

```
Decimal number x (0 < x < 2) =? 0.000000001372
Significand: 101111001001000011110101
Exponent: -30
```

Submission: https://challenge.inf.ethz.ch/team/websubmit.php?cid=5&problem=MP15054

Challenge - Skript-Aufgabe 77

This exercise asks you to draw an approximation to the famous *Mandelbrot set*. You can use the `libwindow` library to produce the drawing. This library is already installed on your system if you set it up according to the documentation from the webpage (or if you use VirtualBox).

Example code that shows how to draw points (and more complicated objects), can be found in the folder `progs/libraries/demo`. A small documentation can be found in the folder `progs/libraries/doc_html` (open the file `contents.html`).