

# GB21802 - Programming Challenges

## Week 0 - Introduction

Claus Aranha

caranha@cs.tsukuba.ac.jp

Department of Computer Science

2018/4/13

(last updated: April 8, 2018)

# Before Anything else: Important Notices

## Manaba Page

All lecture notes and announcements for this course will be done through MANABA. Access the url below:

[https://manaba.tsukuba.ac.jp/ct/course\\_971789](https://manaba.tsukuba.ac.jp/ct/course_971789)

Registration Code: 1473177

## Language

- Lectures: Japanese
- Slides and materials: English
- Exercises: English
- Questions, Mails and Homework: Any language

# About the Lecturer



- **Name:** Claus Aranha;
- **Country:** Brazil;
- **Research:** Artificial Intelligence, Evolutionary Algorithms, Genetic Programming;
- **Language:** Python, R;
- **Hobbies:** Game Programming, Geocaching;
- **webpage:**  
<http://conclave.cs.tsukuba.ac.jp>

# What is this course about?

You have learned many programming techniques...  
...but can you use them?

## Course Philosophy: Learning by Practice

- Every week, you will be asked to solve some programming problems;
- You have to decide the best **data structure**, and **algorithm** to solve each problem;
- Each problem has a **max time**, and **max memory**;
- We will discuss algorithms, techniques and tricks;

## Course Goal:

Improve programming abilities, techniques and familiarity.

# Why you should do this class?

- You like to program, you think programming is fun
- You learned a lot of programming theory, but you need more programming practice;
- You have not written many programs yet;
- You want to think about program efficiency;
- You want a class where skill is more important than memorization;
- You want to practice your technical English;
- You want to participate in Programming Contests;

# Warnings about this class

## 1- Heavy Workload

- Challenges start easy, but end very hard;
- Expect to use a few hours per week on homework;
- Lots of debugging;
- Hint: Do your homework early!

## 2- Course Language

- The teacher's Japanese is very bad T\_T
- All the course materials are in English;
- Importantly: All the homework is in English;
- You can make your homework in Japanese;
- Practice some English in this course too! :-)

# What is a “Programming Challenge”?

A programming challenge is a puzzle that you solve by making a computer program.

- Description of the problem
- Standard input
- Standard output
- Examples

You must write a program that **reads the input** and **prints the correct output**.

# Example Challenge: “Relational Operator” (1)

The challenges for this course are listed at the page:

<http://conclave.cs.tsukuba.ac.jp/lecture/monitor.html>

## allenges 2016: Problem Monitor

### Week 0

Deadline: 88 days, 07:31 hours from now

#	Name	Solved	My Status
1	<a href="#">Relational Operator</a>	0/2	

[click to show/hide](#)

### Week 1

Deadline: 10 days, 07:31 hours from now

[click to show/hide](#)

Click on the title to go to the problem page.



# Example Challenge: “Relational Operator” (2)

Clicking on the title will take you to the problem page.

The screenshot shows the UVA Online Judge interface. At the top, there's a navigation bar with 'UVA' and 'Online Judge'. Below this, a search bar and a 'Search' button are visible. The left sidebar contains a 'Main Menu' with links like 'Home', 'My Account', 'Contact Us', 'ACM-ICPC Live Archive', and 'Logout'. Below that is an 'Online Judge' section with links like 'Quick Submit', 'Migrate submissions', 'My Submissions', 'My Statistics', 'My uhhunt with Virtual Contest Service', 'Browse Problems' (which is highlighted), 'Quick access, info and search', and 'Problemsetters' Credits'. The main content area displays the problem title '11172 - Relational Operator' and a 'Time limit: 3,000 seconds'. On the right, there are buttons for 'Submit', 'Statistics', 'Debug', and 'PDF'. The problem description states: 'Some operators check about the relationship between two values and these operators are called relational operators. Given two numerical values your job is just to find out the relationship between them that is (i) First one is greater than the second (ii) First one is less than the second or (iii) First and second one is equal.' It then defines the input: 'First line of the input file is an integer  $t$  ( $t < 15$ ) which denotes how many sets of inputs are there. Each of the next  $t$  lines contain two integers  $a$  and  $b$  ( $|a|, |b| < 1000000001$ ).' The output requirement is: 'For each line of input produce one line of output. This line contains any one of the relational operators '>', '<' or '=', which indicates the relation that is appropriate for the given two numbers.' A 'Sample Input' is provided: '3' followed by two lines of input pairs '10 99' and '99 10'.

Here you can read the problem and submit a solution.  
(You will need an UVA account!)

## Example Challenge: “Relational Operator” (3)

### Problem Description

Some operator checks about the relationship between two values, these operators are called relational operators. Given **two numerical values**, your job is just to find out the relationship between them.

- First one is greater than the second
- First one is smaller than the second
- First one is equal to the second

### Input

First line is the number  $t$  of tests ( $t < 15$ ). Following  $t$  lines are two integers  $a$  and  $b$ .

### Output

For each line of input, print one line of output with '>', '<' or '=', according to the relationship of  $a$  and  $b$ .

# Solving “Relational Operator”

```
// UVA 11172 - Relational Operator
// Test if a is bigger, smaller or equal to b

#include <iostream>
using namespace std;

int main()
{
    int n; long a, b;

    cin >> n;
    for (; n > 0; n--)
    {
        cin >> a >> b;
        if (a > b) cout << ">\n";
        if (a < b) cout << "<\n";
        if (a == b) cout << "=\n";
    }
}
```

# A Python Solution (python 3)

```
n = int(input())

while (n > 0):
    line = input()
    tokens = line.split()
    a,b = int(tokens[0]),int(tokens[1])

    if a > b: print(">")
    if a < b: print("<")
    if a == b: print("=")

    n -= 1
```

# A Java Solution

```
import java.io.*;
class Main
{
    public static void main(String args[])
    {
        BufferedReader stdin = new BufferedReader(new InputStreamReader(System.in));
        BufferedWriter stdout = new BufferedWriter(new OutputStreamWriter(System.out));
        try {
            String line;
            line = stdin.readLine();
            int n = Integer.parseInt(line);

            for (int i = 0; i < n; i++)
            {
                line = stdin.readLine();
                String[] tokens = line.split("\\s+");
                long a = Integer.parseInt(tokens[0]);
                long b = Integer.parseInt(tokens[1]);

                if (a > b)
                    stdout.write(">\n");
                if (a < b)
                    stdout.write("<\n");
                if (a == b)
                    stdout.write("=\n");
                stdout.flush();
            }
            stdout.close();
        } catch (IOException ioe) { System.out.println("I/O Exception"); }
    }
}
```

# Java Solution – Keep in Mind

- All code must be in the same source file (can define many classes in this file)
- All programs must begin in a static main method in a [Main](#) class.
- Do not use public classes. Even Main must be non public.
- Use Buffered I/O to avoid time limit exceeded.

# How to submit a problem

Your weekly routine should have five steps:

- 1 Read the problem and think how to solve it;
- 2 Write the program, and submit it to UVA website;
- 3 If the program is correct go to 4, else go to 2;
- 4 Prepare your source zip file (all code + comment file);
- 5 Submit your zip file to MANABA;

# Submitting the problem to UVA

UVA is an [Automated Robotic Judge](#). It will test your program on a set of inputs, and check if the outputs are correct.

From the problem page, click on the [submit](#) button.

## 11172 - Relational Operator

Language

- ☐ ANSI C 5.3.0 - GNU C Compiler with options: -lm -lcrypt -O2 -pipe -ansi -DONLINE\_JUDGE
- ☐ JAVA 1.8.0 - OpenJDK Java
- ☐ C++ 5.3.0 - GNU C++ Compiler with options: -lm -lcrypt -O2 -pipe -DONLINE\_JUDGE
- ☐ PASCAL 3.0.0 - Free Pascal Compiler
- ☐ C++11 5.3.0 - GNU C++ Compiler with options: -lm -lcrypt -O2 -std=c++11 -pipe -DONLINE\_JUDGE
- ☐ PYTHON3 3.5.1 - Python 3

Paste your code...

...or upload it

 No file chosen

Select your language, choose the file, and press submit.  
(You can use C, C++, Java, Python and Pascal)



# Submitting the problem to UVA

After you submit the program, the judge gives you the result (Verdict) and the Run time.

## My Submissions

#	Problem	Verdict	Language	Run Time	Submission Date
17182419	1124 Celebrity Jeopardy	Accepted	C++	0.000	2016-04-11 06:42:30
17181459	10141 Request for Proposal	Compilation error	C++11	0.000	2016-04-11 01:36:46
17181444	11498 Division of Nlogonia	Accepted	C++11	0.000	2016-04-11 01:30:43
17071417	102 Ecological Bin Packing	Compilation error	C++11	0.000	2016-03-23 09:21:55
17070667	161 Traffic Lights	Accepted	C++	0.000	2016-03-23 07:24:56
16607686	489 Hangman Judge	Accepted	C++	0.349	2015-12-20 03:52:45
16607670	489 Hangman Judge	Wrong answer	C++	0.335	2015-12-20 03:47:01
16607649	489 Hangman Judge	Runtime error	C++	0.000	2015-12-20 03:40:51

You can see this information on the “my submissions” page.

# Submission Statues:

- **Accepted:** Your program is correct! Congratulations!
- **Wrong Answer:** Your program is incorrect. Debugging time.
- **Time/Memory limit exceeded:** Your program is inefficient. You need a better algorithm.
- **Runtime Error:** Your program is crashing. To the debugger!

# Back to the problem Monitor

In the problem monitor page, you can check how many people solved each problem, which problems you still have to solve, and the deadlines.

## g Challenges 2016: Problem Monitor

Week 0			
Deadline: 11 days, 00:18 hours from now			
#	Name	Solved	My Status
1	<a href="#">Division of Nlogonia</a>	1/2	Accepted
2	<a href="#">Cancer or Scorpio</a>	0/2	Not submitted
3	<a href="#">The <math>3n + 1</math> problem</a>	0/2	Not submitted
4	<a href="#">Request for Proposal</a>	0/2	Not accepted

click to show/hide

# Submitting the problem to MANABA

After you finish the problems listed in the monitor, you need to submit your source code and a comment file as a zip package to MANABA.

s2015XXXXXX-weekYY.zip

- problem1.cpp
- problem2.cpp
- problem5.cpp
- kaisetsu.txt

## Attention

Submission to the UVA judge without a submission to MANABA will not be accepted!

# Outline

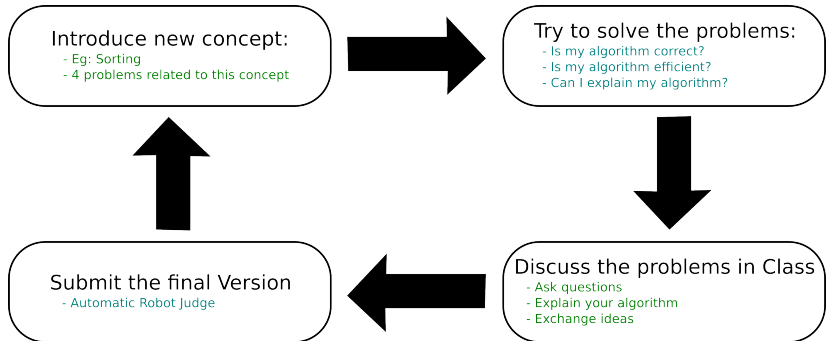
## Two classes per week

- Each week has a theme
- Friday Class: Introduction
- Monday Class: Problem Solving and Q&A

## Solving Problems

- Every week there are 6-10 programming assignments;
- Assignments follow the weekly theme;
- Automatic Submission and Evaluation System;
- Program Deadline is Thursday 23:59

# Outline



# Evaluation and Grading (1)

Evaluation Criteria: Problems solved, Code and Participation

Evaluation Process: Base Grade +Bonus -Penalty

## Evaluation and Grading (2) – Base Grade

The **Base Grade** is based on homework submissions to UVA.

- **C**: You Solved One problem every week;
- **B**: You Solved Two problems every week;
- **A**: You Solved Three problems every week;



# Evaluation and Grading (3) – Bonus and Penalty

A **Bonus** or **Penalty** will be added to the base grade.

- Bonus: grade one step up (C→B, B→A, A→A+)
- Penalty: grade one step down (A+→A, B→C, **C→C**)

## Bonus: Grade Up

- Students with the largest number of solutions in each category (A,B,C)
- Students with exceptional comment files
- Great participation in Class

## Penalty: Grade Down

- More than 25% problems submitted after the deadline

Parameter *N* will be decided at a later date.

## Evaluation and Grading (4) – comment/kaisetsu file

When you submit your package every week, include a text file (no Word!) with comments on each problem you tried to solve.

### Example

```
Name: Claus, ID: 98884735
```

```
# Problem 1:
```

```
To solve this problem, I sorted the input data, and  
printed the input with the highest number of repeated  
letters.
```

```
# Problem 2:
```

```
I tried to solve this problem with brute force, but  
the time limit was exceeded. I had to use DP on the  
number of people instead.
```

Comments may be in Japanese. (FILENAMES must be in romaji)

## Evaluation and Grading (5) – about plagiarism

The assignments are **individual**. Use your **own strength** to solve the programs.

### GOOD

- Ask for ideas to your friends;
- Ask for ideas in the MANABA forum;
- Ask for help with a bug;

### BAD

- Copy a solution from the internet;
- Copy a solution from your friends;
- Give your code to a friend;

Plagiarism will result in course failure, and possibly worse.

# Useful Links

- [Manaba Page](#): All the class material will be here. Access Code is: 1473177
- [UVA Online Judge](#): Use this page to submit your problems.  
**Make an account and list the username on MANABA**
- [Problem Monitor](#): Use this page to check deadlines and weekly problems.
- [Github Repository](#): Working directory for lecture notes. Send me PR, issues!
- [uDebug](#): Web service that generates test inputs and test outputs for UVA problems. Useful tool for this course.

# Course Book

- Competitive Programming, 3rd Edition (<http://cpbook.net>)
- For suggestions of books in Japanese, please check the Manaba materials!

# uDebug Tool

If you are having problems, the uDebug site offers, for many problems in UVA, the correct set of outputs for any input you give.

<https://www.udebug.com/>



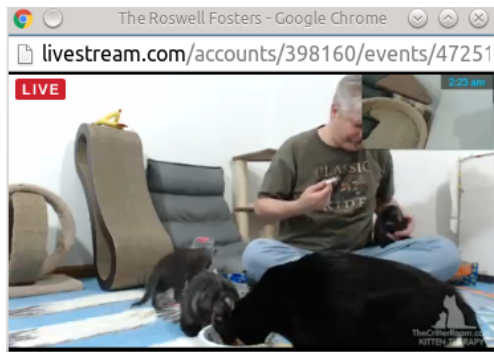
Search for a problem you've solved, provide input, and get accepted output!

[8299 problems and counting!](#)



# If you are still having problems...

Watch a cat stream to relax!



<http://livestream.com/FosterKittenCam/>

# Contact the professor

- **e-mail:** `caranha@cs.tsukuba.ac.jp`
- **website:** `http://conclave.cs.tsukuba.ac.jp`
- **Room:** SB1012 – Send an e-mail and we can talk!

Both English and Japanese are okay!



## Do we still have some time?

- Create an account on UVA (if you already have an account, you can use that)
- Submit your account name to the MANABA
- Ask any other questions you want to know!

Thank you for today!