

Getting Started with Kattis

(by J. Tasse, May 2021)

We will be using Kattis for the NB High School Programming Contest this year. Before the contest starts, you must create an account there. You should also get familiar with this environment, and try submitting the solution to an easy problem, just to make sure you will not have technological issues during the contest. This document will guide you through this.

First, go to open.kattis.com.

CLICK "PROBLEM" TO TRY THE KATTIS ENVIRONMENT

USE THIS LINK TO CREATE AN ACCOUNT

open.kattis.com

Amazon eBay Facebook YouTube

Kattis PROBLEMS CONTESTS RANKLISTS JOBS HELP Search Kattis Log in

Welcome to the Kattis Problem Archive

Here you can find hundreds of programming problems to solve. If you're new here you're very much welcome! *just register and start solving.*

COMPANY	JOB AD
10Xrecruit	Recruiter giving you access to the Swedish start up scene

Suggested problems

DIFFICULTY [?]	PROBLEM	POINTS
TRIVIAL	Hello World!	1.2 pt
	Autori	1.2 pt
EASY	Symmetric Order	1.5 pt
	Soda Slurper	1.5 pt
MEDIUM	Polygon Area	2.8 pt
	WFF 'N PROOF	2.8 pt
HARD	Moogles	5.5 pt
	Escape Plan	5.5 pt

Ranklist

#	USER	SCORE [?]
1	Ivan Petrov	9747.2
2	Josenildo Silva	8225.6
3	Dmitry Lyubshin	8137.2
4	Nick Wu	7793.2
5	Doug Goodman	7707.6
6	Bjarki Ágúst Guðmundsson	7468.8
7	Johan Wind	7260.2
8	Oskar Haarklou Veileborg	6956.2

Universities

#	UNIVERSITY	SCORE [?]
1	National University of Singapore	4168.4
2	KTH Royal Institute of Technology	4026.5
3	Lund University	3593.7
4	Reykjavik University	3110.7

Countries

#	COUNTRY	SCORE [?]
1	Sweden	4886.7
2	United States	4830.6
3	Canada	4498.2
4	Iceland	3831.1

2:57 PM 2021-04-30

Once you created your account, click "problems" to try out this environment. All problems in the competition will use the same format and submission process. You do not need to be in a contest in order to submit solutions to problems and get them judged.

When clicking “problems”, you get all problems available in Kattis...the easy ones and the hard ones! Click “DIFFICULTY” to get those problems sorted by difficulty levels, so that you can try the easiest ones. I recommend that you try the problem “Quadrant Selection” to get familiar with the system. Note: you can also access a given problem by searching it (see the search area at the top).

The screenshot shows the Kattis website interface. At the top, there is a navigation bar with the Kattis logo and links for PROBLEMS, CONTESTS, RANKLISTS, JOBS, and HELP. A search bar labeled "Search Kattis" and a "Log in" button are also present. Below the navigation bar, the main content area is titled "Problems" and includes an RSS feed link for new problems. A table lists various problems with columns for NAME, SUBMISSIONS (TOTAL, ACC., RATIO, FASTEST), and USERS (TOTAL, ACC., RATIO, DIFFICULTY). The "DIFFICULTY" column is circled in blue. The table lists 20 problems, including "0-1 Sequences", "10 Kinds of People", "2048", "2, 4, 6, Greaaaat", "3D Printed Statues", "3D Printer", "3-Sided Dice", "4 thought", "A1 Paper", "Aaah!", "Abandoned Animal", "ABC", "Ab Initio", "Above Average", "A+B Problem", "A Brief Gerrymander", "Abstract Art", "Abstract Painting", "Absurdistan Roads II", and "Absurdistan Roads III". The Windows taskbar is visible at the bottom of the screen, showing the time as 3:09 PM on 2021-04-30.

NAME ^	SUBMISSIONS				USERS				DIFFICULTY		
	TOTAL	ACC.	RATIO	FASTEST	TOTAL	ACC.	RATIO				
0-1 Sequences	10870	1703	16%	0.00	2213	1174	53%	5.5			
10 Kinds of People	25186	5138	20%	0.01	4375	3162	72%	4.7			
2048	10297	4719	46%	0.00	4549	3896	86%	2.4			
2, 4, 6, Greaaaat	739	107	14%	0.15	84	37	44%	9.4			
3D Printed Statues	13522	6223	46%	0.00	5812	5215	90%	1.9			
3D Printer	948	157	17%	0.00	391	120	31%	8.8			
3-Sided Dice	2834	233	8%	0.00	454	162	36%	8.6			
4 thought	11773	4036	34%	0.00	3878	3276	84%	2.7			
A1 Paper	6433	1939	30%	0.00	1919	1532	80%	3.7			
Aaah!	34188	15945	47%	0.00	14319	13367	93%	1.6			
Abandoned Animal	2449	551	22%	0.03	630	462	73%	5.4			
ABC	18571	8434	45%	0.00	7983	7247	91%	1.8			
Ab Initio	1735	290	17%	0.09	254	157	62%	7.6			
Above Average	12236	5088	42%	0.00	4895	4399	90%	1.8			
A+B Problem	9351	921	10%	0.03	1339	444	33%	8.3			
A Brief Gerrymander	519	104	20%	0.11	96	46	48%	8.9			
Abstract Art	445	147	33%	0.00	117	89	76%	5.7			
Abstract Painting	985	408	41%	0.00	367	301	82%	3.8			
Absurdistan Roads II	691	235	34%	0.00	232	155	67%	6.9			
Absurdistan Roads III	1724	493	29%	0.02	468	356	76%	5.6			

Here is what the page will look like after selecting that problem ("Quadrant Selection").

PROBLEM DESCRIPTION

CLICK HERE TO SUBMIT YOUR SOLUTION

CLICK HERE TO GET THE INPUT FILES FOR TESTING YOUR CODE

Quadrant Selection

A common problem in mathematics is to determine which quadrant a given point lies in. There are four quadrants, numbered from 1 to 4, as shown in the diagram below:

Quadrant 2	Quadrant 1
B (-12, 5) •	• A (12, 5)
C (-12, -5) •	• D (12, -5)
Quadrant 3	Quadrant 4

For example, the point *A*, which is at coordinates (12,5) lies in quadrant 1 since both its *x* and *y* values are positive, and point *B* lies in quadrant 2 since its *x* value is negative and its *y* value is positive.

Your job is to take a point and determine the quadrant it is in. You can assume that neither of the two coordinates will be 0.

Input

The first line of input contains the integer x ($-1000 \leq x \leq 1000; x \neq 0$). The second line of input contains the integer y ($-1000 \leq y \leq 1000; y \neq 0$).

Output

Submit Stats

Problem ID: quadrant
CPU Time limit: 1 second
Memory limit: 1024 MB
Difficulty: 1.2
Download:
Sample data files

Source: Canadian Computing Competition 2017
License: For educational use only

Be very careful with the Input/Output specifications.

Make sure that your program runs properly with any input matching the specifications provided. You do not have to add code for checking that the input does indeed have that format (e.g., checking that the first line does contain a single number in the specified range, for the example above): it is guaranteed that the input will meet such specifications.

Be very careful with the output format. Even adding an extra space somewhere would be considered as a wrong answer! Make sure you do not have any spelling mistake, that you have a return character at the end of each line, and that you do not have extra blank lines at the end.

Develop your solution in your own IDE / programming environment, outside of Kattis. Use the sample data files provided to help you test your program.

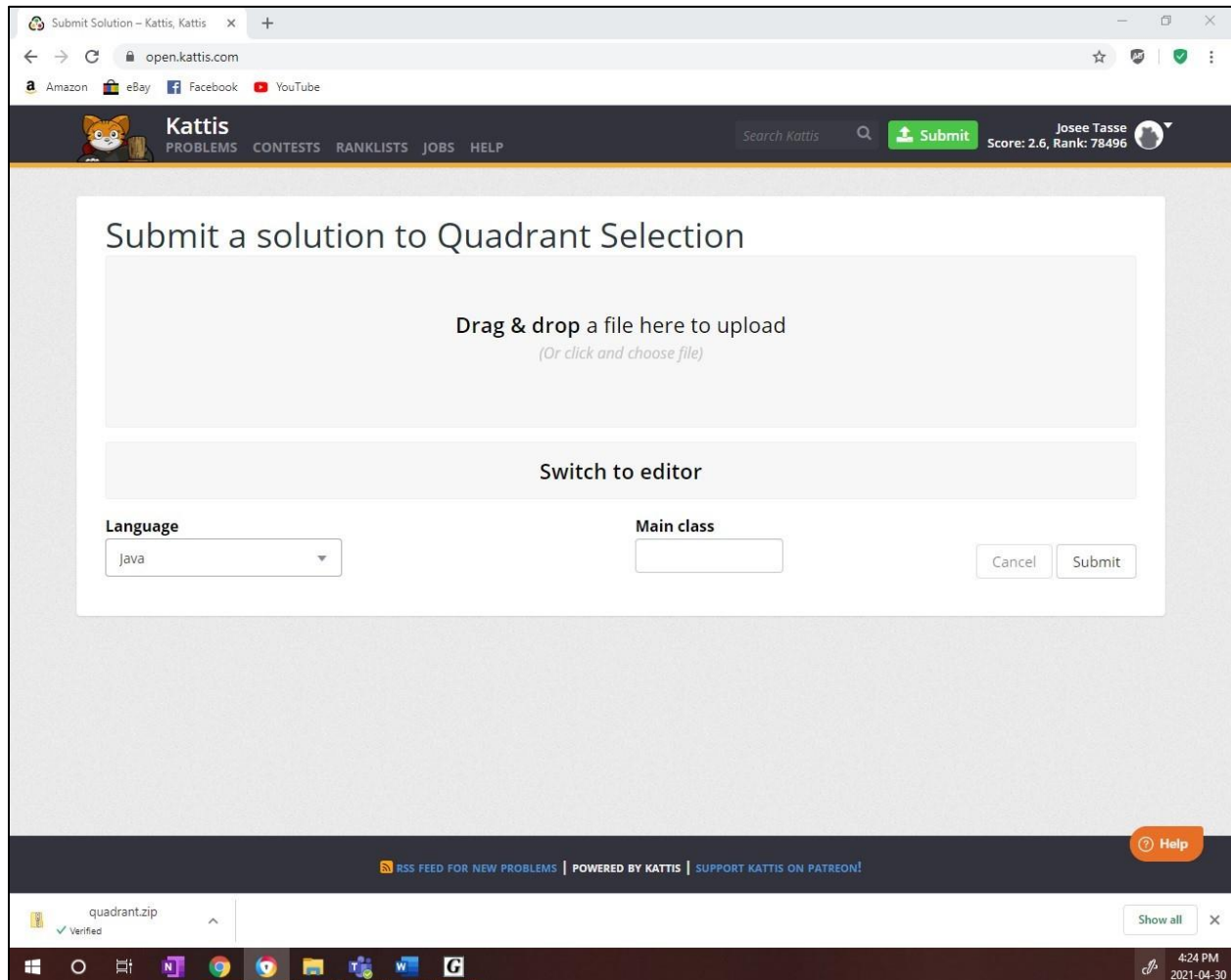
When you submit your program for judging, the verification process is completely automated: your program is first compiled, and then run multiple times using different inputs. Your output is compared to the expected output. So although you should read your input from standard input (i.e., keyboard) and print to standard output (i.e., monitor), you should not worry about telling the user what to enter (e.g., "Enter the value for x: "). This would actually get your program rejected.

There are 2 ways you can test your program. First, you can use the "command prompt" to run it by feeding it the input file. The following video does show you how this can be done for Java and for Python: Video "Kattis – Getting Started": <https://www.youtube.com/watch?v=7NHyzBN4GAM>

Note that the video contains also other information about Kattis. The part about running at the command prompt starts around the 3:00 mark.

You can also run your program directly inside your IDE (as you are used to), copy-pasting the whole content of the input file at once at the first occurrence of the programming waiting to get some input. Check your output after the last line of input.

Once you are ready to submit, click on the “submit” button, which will get you to the window below. Note: this will not work unless you are signed in. Please make sure that you upload your file, select the programming language used, and add any other requested information specific to your language (e.g., type the Main class in Java).



Please try this out before our contest, to get familiar with the Kattis environment. If you have any problems (before the contest), please do not hesitate to contact us: Josee Tasse (jtasse@unb.ca) or Owen Kaser (owen@unb.ca).