

Problem Set 4

*Prof. Nodari Sitchinava**Due: Friday, November 15, 2024 at 1:30pm*

You may discuss the problems with your classmates, however **you must write up the solutions on your own** and **list the names** of every person with whom you discussed each problem.

1 Skyline (100 pts)

This homework consists of one programming assignments for you to practice implementing a plane sweep algorithm. You will submit **the source code** on the submission server <https://judge.ics.hawaii.edu>. To login to the server, use the same username as your UH username, and your password is your student ID. Because the system is used for programming contests, it shows a scoreboard of who solved which problem, when and with how many attempts. To preserve anonymity, you are assigned a random letter as the team name, which will be used to identify your submissions on the scoreboard. This letter will not change throughout the semester.

Input/Output Formats

Your program should read the input from the standard input and print the answer to the standard output. You may submit source code written in Java, C, C++, Python2, or Python 3 (make sure to choose the correct language during the submission). The server allows you to download simple examples of the input and expected output. You can use these to check that your program correctly reads the input and outputs the result in the correct format as follows (`test.in` and `test.out` are the files containing the sample inputs and outputs).

```
% ./myprogram < test.in | diff test.out -
```

Remember, the sample input/output files are provided for you to test your understanding of the input/output formats. You should create your own testcases for testing and debugging your code.

Time limits

Note the time limit for the problem. Your program must finish within the allocated time limit using the server's hardware. Therefore, you should design the most efficient algorithm you can, before submitting it.

Technical details

The submission server will compile your code with the following switches, depending on the language:

- C: `gcc -x c -O2 -std=gnu11 -static -pipe file.c -lm`
- C++: `g++ -x c++ -O2 -std=gnu++14 -static -pipe file.cpp`
- Java: `javac -encoding UTF-8 -sourcepath . -d . file.java`

The submission server runs Debian 10 Operating System with the following compilers:

- C: `gcc v8.3.0`
- C++: `g++ v8.3.0`
- Java: `javac v11.0.12`

- Python 2: `python2 v2.7.16`
- Python 3: `python3 v3.7.3`

For each language, if the above compilation step is successful, the submission will be executed as follows:

- For C/C++: the executable file generated by the compiler will be executed to generate the output of the submission.
- For Python 2: the main source file will be executed by the `python` interpreter with JIT compiler to generate the output of the submission.
- For Python 3: the main source file will be executed by the `python3` interpreter to generate the output of the submission.
- For Java: the compiled main class will be executed using the following command:

```
java -Dfile.encoding=UTF-8 -XX:+UseSerialGC -Xss64m -Xms1920m -Xmx1920m
```

The system will use OpenJDK 11.0.12 Runtime Environment.

Submission Server Feedback

The submission server will run your submissions through a number of testcases. Upon running the testcases, the submission server will return one of the following:

- **CORRECT**: All tests have passed
- **WRONG ANSWER**: Your program returned the wrong answer on at least one of the testcases
- **RUN ERROR**: Your program exited with an error on at least one of the testcases (typically is caused by a crash)
- **TIMELIMIT**: Your program ran longer than the allocated time limit
- **COMPILER-ERROR**: There was an error during the compilation of your program

The server will stop running additional testcases as soon as it encounters (and reports) one of the errors above.

Grading

The problem is worth 100 points. To receive any credit, your program must pass **all** the testcases. You may submit your code as many times as you like. However, for each submission that doesn't pass all the tests, you will be deducted 1 point. The point of these deductions is to discourage you from using the system as a debugger: You should design the most efficient algorithm you can, analyze it (to make sure it will perform within the time limit), prove that it is correct on all inputs, implement it, test it on your own testcases, and submit only after you are confident that it works. Once the program passes all the tests, all consequent submissions will still be run/tested, but the system will not penalize you for additional submissions (so you can try other solutions if you like without worrying that it'll affect your grade).

The submission server reports two statistics for each problem: the first one is the number of submissions, the second one is the number of minutes since the submission server was started. **Remember**: there is no advantage for submitting the code early or being the first one to get the correct answer. You can totally ignore the second number and the "rankings" of the users.

2 OPTIONAL: Union of triangles (0 pts)

Let S be a set of n triangles in the plane. The boundaries of the triangles are disjoint, but it is possible that a triangle lies completely inside another triangle. Let P be a set of n points in the plane. Give an $O(n \log n)$ algorithm that reports each point in P lying outside all triangles.

3 OPTIONAL: Circle intersection reporting (0 pts)

Let S be a set of n circles in the plane. Describe a plane sweep algorithm to compute all intersection points between the circles. (Because we deal with circles, not discs, two circles do not intersect if one lies entirely inside the other.) Your algorithm should run in $O((n + k) \log n)$ time, where k is the number of intersection points.