**Efrog**

tests/input/1.txt  
 2.txt

/output/1.txt  
 2.txt

statement/statement.tex  
 input\_statement.tex

output\_statement.tex

note.tex

general\_info/name.txt  
 time\_limit.txt

memory\_limit.txt

examples/input/1.txt  
 2.txt

/output/1.txt  
 2.txt

**Polygon**

tests/01  
 01.a

statement-section/language/name.tex

input.tex

output.tex

legend.tex

example.01.a

example.01

statements/

solutions/

scripts/

files/

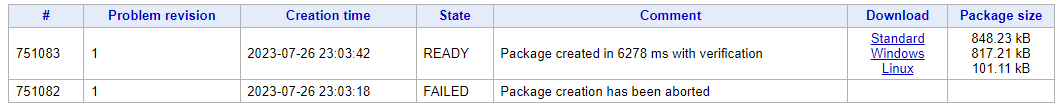
wipe.bat

problem.xml

doall.bat

check.exe

check.cpp

При завантаженні файлу файл можна завантажити до будь якої системи.  


Файл, який завантажується призначений для публікації задачі(наприклад на гітхаб), сам процес заватаження файлів до задачі відбувається по одному та відображається на вкладці умови. Файл problem.xml содержит не только базовую мета-информацию вроде ограничений на время и память, но и в деталях всё что понадобится для последующей автоматизированной работы над задачей.

Вот несколько примеров:

* для TL указан тип процессора, для которого он был выбран;
* явно указан способ ввода-вывода и имена файлов, если таковые используются;
* название задачи с поддержкой многоязычности;
* точный способ генерации каждого из генерируемых тестов;
* теги решений (например, заведомо медленное решение, может быть помечено тегом time-limit-exceeded);
* точные пути до тестов и прочих ресурсов.

Що до дублювання файлів за відсутності документації, усі відповіді брав з статей з codeforses або habr  
посилання: <https://habr.com/ru/articles/748298/>. За відсутності однозначної відповіді маю здогадку, що це використовують для зручності перегляду завантаженої задачі. Імпорт у Efrog планую робити через json файл який знаходиться в папці statement та містить одразу усю інформацію до задачі окрім тестів.

**Ejudge**

**Посилання на документацію:**

[**https://ejudge.ru/wiki/index.php/%D0%90%D0%BB%D1%8C%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F\_%D1%80%D0%B0%D1%81%D0%BA%D0%BB%D0%B0%D0%B4%D0%BA%D0%B0\_%D1%84%D0%B0%D0%B9%D0%BB%D0%BE%D0%B2**](https://ejudge.ru/wiki/index.php/%D0%90%D0%BB%D1%8C%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B0%D1%8F_%D1%80%D0%B0%D1%81%D0%BA%D0%BB%D0%B0%D0%B4%D0%BA%D0%B0_%D1%84%D0%B0%D0%B9%D0%BB%D0%BE%D0%B2)

Звичайна

statements/A.xml

B.xml

checkers/check\_A

check\_B

tests/A/1.dat

1.answ

B/1.dat

1.answ

Альтернативна

problems/A/statement.xml

check

tests/1.dat

1.answ

B/statement.xml

check

tests/1.dat

1.answ

Що до лімітів у ejudge, вони зберігаються у файлі serve.cfg та однакові для всіх задач, томи при імпорті задач з ejudge у efrog будуть задаватись дефолтні значення 5с та 256мб для всіх задач, за бажанням можна змінити.  
В ejudge є можливість імпорту задач з Polygon посилання:

<https://ejudge.ru/wiki/index.php/%D0%A0%D1%83%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE_%D0%BF%D0%BE_%D0%B8%D0%BC%D0%BF%D0%BE%D1%80%D1%82%D1%83_%D0%B7%D0%B0%D0%B4%D0%B0%D1%87_%D0%B8%D0%B7_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B_Polygon_%D0%B2_%D1%82%D1%83%D1%80%D0%BD%D0%B8%D1%80%D1%8B_ejudge>

Для включения альтернативной раскладки файлов необходимо установить глобальную конфигурационную переменную advanced\_layout в файле serve.cfg турнира.

Чекери використовуються для більш варіативного перевірення відповіді

* [cmp\_file](https://ejudge.ru/wiki/index.php/Cmp_file) — сравнение двух файлов.
* [cmp\_file\_nospace](https://ejudge.ru/wiki/index.php/Cmp_file_nospace) — сравнение двух файлов с игнорированием повторяющихся пробелов.
* [cmp\_bytes](https://ejudge.ru/wiki/index.php/Cmp_bytes) — сравнение двух файлов байт в байт.
* [cmp\_int](https://ejudge.ru/wiki/index.php/Cmp_int) — сравнение двух знаковых 32-битных целых чисел.
* [cmp\_int\_seq](https://ejudge.ru/wiki/index.php/Cmp_int_seq) — сравнение двух последовательностей знаковых 32-битных целых чисел.
* [cmp\_long\_long](https://ejudge.ru/wiki/index.php/Cmp_long_long) — сравнение двух знаковых 64-битных целых чисел.
* [cmp\_long\_long\_seq](https://ejudge.ru/wiki/index.php/Cmp_long_long_seq) — сравнение двух последовательностей знаковых 64-битных целых чисел.
* [cmp\_unsigned\_int](https://ejudge.ru/wiki/index.php/Cmp_unsigned_int) — сравнение двух беззнаковых 32-битных целых чисел.
* [cmp\_unsigned\_int\_seq](https://ejudge.ru/wiki/index.php/Cmp_unsigned_int_seq) — сравнение двух последовательностей беззнаковых 32-битных целых чисел.
* [cmp\_unsigned\_long\_long](https://ejudge.ru/wiki/index.php/Cmp_unsigned_long_long) — сравнение двух беззнаковых 64-битных целых чисел.
* [cmp\_unsigned\_long\_long\_seq](https://ejudge.ru/wiki/index.php/Cmp_unsigned_long_long_seq) — сравнение двух последовательностей беззнаковых 64-битных целых чисел.
* [cmp\_huge\_int](https://ejudge.ru/wiki/index.php/Cmp_huge_int) — сравнение двух целых чисел произвольного размера.
* [cmp\_double](https://ejudge.ru/wiki/index.php/Cmp_double) — сравнение двух вещественных чисел двойной точности с заданной максимальной ошибкой.
* [cmp\_double\_seq](https://ejudge.ru/wiki/index.php/Cmp_double_seq) — сравнение двух последовательностей вещественных чисел двойной точности с заданной максимальной ошибкой.
* [cmp\_long\_double](https://ejudge.ru/wiki/index.php/Cmp_long_double) — сравнение двух вещественных чисел расширенной точности с заданной максимальной ошибкой.
* [cmp\_long\_double\_seq](https://ejudge.ru/wiki/index.php/Cmp_long_double_seq) — сравнение двух последовательностей вещественных чисел расширенной точности с заданной максимальной ошибкой.
* [cmp\_sexpr](https://ejudge.ru/wiki/index.php?title=Cmp_sexpr&action=edit&redlink=1) — сравнение двух S-выражений (деревьев в синтаксисе языка Lisp).
* [cmp\_yesno](https://ejudge.ru/wiki/index.php/Cmp_yesno) — сравнение двух ответов YES или NO.

<https://ejudge.ru/wiki/index.php/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D1%8B%D0%B5_%D0%BF%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D1%8F%D1%8E%D1%89%D0%B8%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D1%8B>

**Valladolid**

Це місто у Іспанії, саме такої системи не знайшов, знайшов сайт який належить університету цього міста – це UVa Online Judge  
Сайт

<https://onlinejudge.org/index.php>

Посилання на статтю

<https://en.wikipedia.org/wiki/UVa_Online_Judge>

Для рядового користувача не знайшов функції на створення задач. Документація з даного приводу також вістуна.

Наразі більш популярним рішенням є Sphere Online Judge на якому більше задач, більше підтримуємих мов програмування та більше користувачів.

**SPOJ**

Для створення задач потрібно вирішити шо найменш 20 задач з сайту, відправити лист до адміністрації сайту, після чого вам можуть дозволити створення задач. Задача створюється у форму у вікні. Можливості завантажити файлом чи скачати файл після створення відсутня. Не маю ідей як реалізувати імпорт зачачі.

Посилання на документацію:

[**https://www.spoj.com/tutorials/PS/**](https://www.spoj.com/tutorials/PS/)

**USACO**

USACO, або Олімпіада з інформатики США, - це національне змагання, яке проводиться чотири рази на рік1. Однак USACO не надає функціональності для проведення власних олімпіад з програмування. Це змагання призначене для участі студентів, а не для власних олімпіад.

Посилання на сайт:   
<http://www.usaco.org/index.php?page=contests>

У той самий час можливість задавати свої задачі є на USACO guide у розділі групи. Але для того щоб задати задачу учасникам групи її власник може задати задачу з самого USACO guide або будь якого іншого сайту(наприклад Codeforses). Завантажити задачу файлом неможливо.  
Посиланняна группи: <https://usaco.guide/groups>

Посилання на задчу приклад: [Problem: Untitled Problem · New Group · USACO Guide](https://usaco.guide/groups/xfKpyFUpReWIdwMC9DPk/post/EOqNCo9EvfKN9ycQYyL9/problems/zsK0KzDjICwuu5uVjoWA)

**Geeksforgeeks**

Geeksforgeeks не надає можливості проводити власні олімпіади та створювати власні задачі, тому імпорт задач з Geeksforgeeks не можливий

CMS

contest.yaml  
[task\_name]/gen/  
 input/input0.txt

output/output0.txt

sol/solution.cpp

statement/statement.tex

task.yaml

<https://github.com/cms-dev/con_test/tree/master/batch/input>