

# Simple Programming Contests Online Judge System

Документация на електронна тренировъчна система за автоматично оценяване на задачи по програмиране

## Формат на задачите

---

Условието на задачата да се изготви на български и английски езици. Всяка задача се намира в поддиректорията на темата си, чието име съвпада с letter полето от базата, която съдържа:

### description.txt

---

Условието на задачата в един или няколко формата (txt, doc, pdf). Ако такова не присъства, се търси условие от директорията на темата. Важат същите ограничения като при условията на темите. Примерен [description файл](#)

### test.in

---

Входни данни за тестване на изпратените решения при еднотестова задача. Примерен файл може да видите от [тук](#).

### test.ans

---

Изходни данни за тестване на изпратените решения при еднотестова задача. Примерен файл може да видите от [тук](#).

### test.?.in

---

В случай на многотестови задачи, набор от входни данни. Трябва да се именуват като: text.01.in, text.02.in, text.03.in и т.н. Системата определя дали задачата е еднотестова или многотестова по присъствието на файл с име text.01.in.

### test.?.ans

---

В случай на многотестови задачи, набор от изходни данни. Трябва да се именуват като text.01.ans, text.02.ans, text.03.ans и т.н.

Внимание: въпреки че `spojo` поддържа многотестови задачи, към текущият момент системата не поддържа частично оценяване, т.е. задачите се приемат за верни единствено ако минават всички тестове. тестовите двойки трябва да са с

# Simple Programming Contests Online Judge System

Документация на електронна тренировъчна система за автоматично оценяване на задачи по програмиране

последователни номера, започващи от `00`, т.е. в тях не трябва да има пропуснати номера.

## solution\*

---

Файлове с имена от вида `solution\*.cpp` или `solution\*.java` които представляват авторските решения (незадължително). Примерен файл може да видите от [тук](#).

## checker

---

Изпълним файл, програма проверяваща коректността на даденото решение (не е задължителна). Ако такава отсъства, се приема, че изходните данни трябва да съвпадат буквално с дадените.

```
checker [входен_файл] изходен_файл [помощен_файл]
```

Чекерите са произволни изпълними файлове (могат да бъдат както компилирани, така и писани на интерпретиран език, когато това е удобно). За проверка на задачата те се изпълняват с три аргумента – входен файл, авторски изходен файл и изходен файл от изпълнение на решението. Те се извикват само когато решението е изпълнено успешно и е влязло в ограничението на времето. Възможно е, при задачи с чекерите в .ans файловете да не се намира реален резултатен файл, а подсказваща информация за чекера. Чекера трябва да върне код 0, когато решението е дало коректен резултат или код 1, когато резултатът е грешен. Когато чекерът върне друга стойност това се интерпретира като вътрешна грешка, което може да се види от треньора или администратора за да се анализира проблема. Всичко което се изведе от чекера на стандартен изход или стандартен изход за грешки се записва в лога на изпълнението. Всички материали за една задача да са в отделна папка, а папките на всички задачи да са архивирани.

# Simple Programming Contests Online Judge System

Документация на електронна тренировъчна система за автоматично оценяване на задачи по програмиране

## set-info.conf

Представява текстов файл състоящ се от редове от вида `**ключ**=<стойност>`. За него са дефинирани следните ключове:

Ключ	Описание
<b>name</b>	Име на темата / състезанието
<b>start_time</b>	Време в което темата / състезанието става активно. Трябва да е коректно написана в ISO 8601 формат *до секундите* включително
<b>duration</b>	продължителност на темата / състезанието в минути
<b>show_sources</b>	1 или 0. Определя дали след изтичане на времето на състезанието, изпратените решения ще бъдат видими за всички
<b>about</b>	Описание на темата/ състезанието. Може да съдържа HTML

Пример:

```
name=DAA 2009 - Practice problems for week 9
start_time=2009-05-19 08:00:00
duration=2880
show_sources=1
about=DAA 2009 - Practice problems for week 9 - Dynamic Programming
```

# Simple Programming Contests Online Judge System

Документация на електронна тренировъчна система за автоматично оценяване на задачи по програмиране

## problem-info.conf

Дефинира свойствата на дадена задача. Също представлява текстов файл от ключове и стойности. За него са дефинирани следните ключове:

Ключ	Описание
<b>name</b>	Име на проблема / задачата
<b>time_limit</b>	Цяло число указващо допустимото време за изпълнение в секунди
<b>about</b>	Допълнително описание на проблема / задачата. Може да съдържа HTML

Пример:

```
name=variables
time_limit=6
about=DAA 2009 – Week 10 – Task 1
```

В никакъв случай не поставяйте изпълними файлове (\*.exe или \*.bat) в архива.