Задания на заключительный этап олимпиады студентов "Я - Профессионал" по направлению "Искусственный интеллект" 2020/2021 учебный год (КИМ для уровня "бакалавриат")

1. Все задания заключительный этап Олимпиады делятся на два типа:
   (a) (первого типа) задание на математические основы искусственного интеллекта, где требуется отправить решение в виде короткой программы на платформе Яндекс.Контест (таких заданий 5, и на решение их всех выделено 2 часа)
   (b) (второго типа) задание, требующее решения конкретной практической задачи, связанной с применением методов искусственного интеллекта. (таких заданий 3, и на решение их всех выделено 4 часа)

2. Задания первого типа оцениваются исходя из 8 баллов. Баллы начисляются за успешное прохождение автоматических тестов программного кода, отправляемого участниками на платформу Яндекс.Контест (предварительные тесты не включаются).

3. Задания второго типа оцениваются исходя из 20 баллов. Каждая посылка решения участниками имеет какое-то количество очков. Очки линейно нормируются в баллы так, чтобы 20 баллов получила посылка решения участника с максимальным количеством очков, а 1 балл получила посылка с базовым решением. После отправки решения на Яндекс.Контест сразу автоматически будет выводиться количество баллов, вычисленное относительно идеального решения. По окончании времени выполнения заданий рейтинговые показатели будут пересчитаны в баллы относительно лучшего решения участников Олимпиады.

4. В случае подачи апелляции участником, она должна содержать номер/номера посылок, с оценкой которых участник не согласен.

5. Задания заключительного этапа олимпиады студентов "Я - Профессионал"по направлению "Искусственный интеллект"размещаются членами методической комиссии на платформе Яндекс.Контест https://contest.yandex.ru.

1 Оценка модели. 8 баллов.

Вам предстоит сравнить вашу модель с «идеальной» моделью в задаче регрессии. Как ваша модель, так и идеальная представляют из себя кусочно-линейные функции $f(x)$ и $g(x)$ соответственно, где $f, g : \mathbb{R} \to \mathbb{R}$.

Требуется найти метрику качества сравнения моделей как площадь между графиками $f(x)$ и $g(x)$ (см. пример на рисунке 1). Формально

$$\int_a^b |f(x) - g(x)|dx$$

Учтите, что как f, так и g могут быть разрывными.

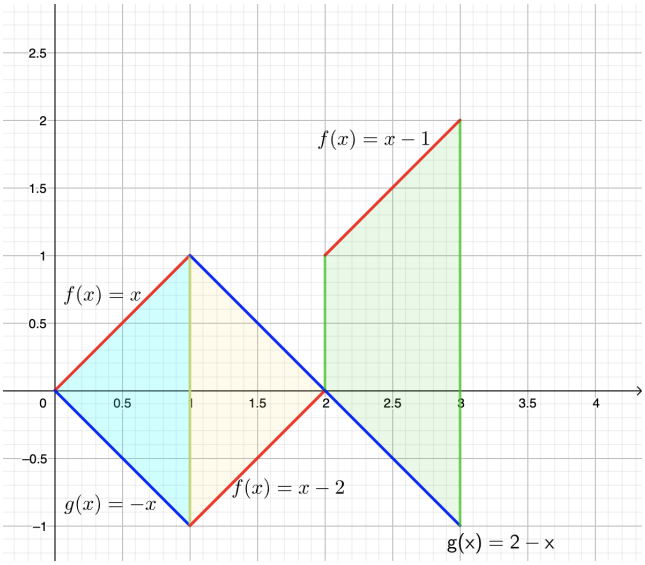

Рис. 1: Иллюстрация определения площади между графиками. Красным цветом выделен график функции f(x), синим - график g(x)

1.1 Формат ввода

На вход в первой строке через пробел подается два натуральных числа N и M ($1 \leq N, M \leq 10^5$) — число точек, внутри которых f и g линейны соответственно.

Далее идет на одной строке $N+1$ натуральных чисел $x_{f,i}$, таких, что $-10^6 \leq a = x_{f,0} < x_{f,1} < \ldots < x_{f,N} = b \leq 10^6$.

Затем идет N строк по два числа a_i b_i ($-10^2 \leq a_i, b_i \leq 10^2$) — коэффициенты прямой для $f(x) = a_i x + b_i$ на полуинтервале $[x_{f,i}, x_{f,i+1})$.

Далее идет на одной строке $M+1$ натуральных чисел $x_{g,i}$, таких, что $-10^6 \leq a = x_{g,0} < x_{g,1} < \ldots < x_{g,M} = b \leq 10^6$.

Затем идет M строк по два числа a_i b_i ($-10^2 \leq a_i, b_i \leq 10^2$) — коэффициенты прямой для $g(x) = a_i x + b_i$ на полуинтервале $[x_{g,i}, x_{g,i+1})$.

1.2 Формат вывода

Выведите метрику качества с точностью до четвертого знака.

2 Пароль. 8 баллов.

Это интерактивная задача

Хакер Андрей узнал, что пароль состоит из двух целых неотрицательных чисел a и b. Эти числа нельзя получить напрямую, но можно их угадать, так как у сервера имеется команда «? n», которая вернет количество решений уравнения $ax + by = n$ в целых неотрицательных числах.

Андрей бы и сам мог справиться с этой задачей, но есть всего лишь одна попытка ввести пароль. Поэтому он просит вас помочь ему найти ответ. Когда вы будете готовы, Андрею нужно будет ввести в терминал команду «! a b», которая означает попытку войти в систему.

2.1 Формат ввода

Вы в праве делать 2 типа запросов.

- «? n» — узнать, сколько имеется решений у уравнения $ax + by = n$ в целых неотрицательных числах.
- «! a b» — сообщить ответ на задачу.

Учтите, что запросов первого типа можно совершить не более тысячи. В запросе первого типа число n должно быть неотрицательным и целым, при этом не больше 10^3.

После запроса второго типа, ваша программа должна завершиться. Не забывайте сбрасывать буфер вывода после каждого запроса. Для сброса буфера вывода (то есть для операции 'flush') сразу после вывода запроса можно сделать:

- fflush(stdout) в языке C++;
- System.out.flush() в Java;
- stdout.flush() в Python;
- flush(output) в Pascal;
- смотрите документацию для других языков.

2.2 Формат вывода

В связи с тем, что взаимодействие интерактивное, приведем пример ввода и вывода вашей программы в ходе взаимодействия с интерактором:

Поток ввода	Поток вывода
	? 1
2	
	? 3
4	
	! 1 1

2.3 Комментарии

Вводить a и b можно в любом порядке во втором типе запроса.

Вы получите вердикт Wrong Answer, если:

- Было сделано больше 1000 запросов.
- Итоговый ответ не является правильным.

Вы получите вердикт Presentation Error, если:

- Запрос не соответствует описанному выше формату.

Вы получите вердикт Idleness Limit Exceeded, если не будете ничего выводить (а тестирующая программа будет ожидать ввода) или забудете сделать операцию 'flush' после какого-нибудь вывода (смотрите ниже).

Если ваша программа, написанная на интерпретируемом языке (например Python), получает какую-либо ошибку исполнения во время общения с интерактором, то возможно получение любого вердикта Presentation Error/Wrong Answer/Idleness Limit Exceeded, поскольку сообщение об ошибке передается интерактору и может быть воспринято поразному.

3 Метро. 8 баллов.

На планете Тетра есть четыре города A, B, C и D, расположенные в вершинах правильного тетраэдра. Также на этой планете есть метро, составы которого курсируют вдоль ребер тетраэдра. Машинист Василий начинает свой день с города A и заканчивает в том же самом городе.

За день Василию надо проехать вдоль K ребер. Но ездить постоянно по одному и тому же маршруту может наскучить, поэтому Василию интересно, а сколько существует различных необязательно простых маршрутов из города A в город A длины K. Можно посещать город A несколько раз.

3.1 Формат ввода

На вход подается число K ($1 \leq K \leq 10^{100000}$) — длина маршрута Василия в ребрах.

3.2 Формат вывода

Выведите ответ на задачу по модулю $(10^9 + 7)$.

4 Акинатор. 8 баллов.

Это интерактивная задача, предложенная партнером Олимпиады - Российской ассоциацией искусственного интеллекта

Есть база данных известных людей, всего $N < 1000$ записей. Каждый человек описан следующими признаками: имя, страна рождения, континент, пол, профессия, область деятельности.

Нужно угадать человека, задав не более $M = 40$ вопросов о значениях его признаков. Вопрос имеет вид«признак = значение», возможен ответ `YES` или `NO`.

Для каждого из 8 наборов тестовых данных программа запускается 10 раз. Тест засчитывается, если удалось угадать персонажа за 30 во всех случаях.

4.1 Формат ввода

Первая строка содержит натуральное число. N ($1 \leq N < 10^3$) — число записей.

Вторая строка содержит названия признаков, разделитель «;». Дальше идет N строк с данными, разделитель «;».

4.2 Формат вывода

На каждой итерации программа может вывести на печать вопрос в следующем формате: «? feature value». Здесь *feature* — название признака, например, «gender», и *value* — значение признака, например, «MALE».

На следующий строке будет получен ответ, `YES` или `NO`. В конце работы программа должна вывести окончательный ответ в формате: «! name».

После этого программа завершает работу.

В связи с тем, что взаимодействие интерактивное, приведем пример ввода и вывода вашей программы в ходе взаимодействия с интерактором:

Поток ввода	Поток вывода
	? continentName Europe
YES	
	? countryName MOLDOVA
NO	
	! Linda Maria Baros

4.3 Комментарии

Вы получите вердикт Presentation Error, если:

- Запрос не соответствует описанному выше формату.

Вы получите вердикт Idleness Limit Exceeded, если не будете ничего выводить (а тестирующая программа будет ожидать ввода) или забудете сделать операцию 'flush' после какого-нибудь вывода (смотрите ниже).

Если ваша программа, написанная на интерпретируемом языке (например Python), получает какую-либо ошибку исполнения во время общения с интерактором, то возможно получение любого вердикта Presentation Error/Wrong Answer/Idleness Limit Exceeded, поскольку сообщение об ошибке передается интерактору и может быть воспринято поразному.

5 Дуров, верни стену! 8 баллов.

Вы задались целью уговорить Павла Дурова вернуть стену. Но его личные сообщения открыты только для друзей, поэтому вам необходимо добавиться к нему в друзья. Изначально у вас нет друзей. Известно, что граф друзей имеет вид связного графа без петель и кратных ребер, где вершины — пользователи, а ребро означает дружбу (она всегда взаимная).

При этом каждый пользователь заботится о том, кто у него в друзьях, поэтому он добавит в друзья человека, только если у них как минимум $k_i \geq 0$ общих друзей. Вам нужно узнать, сможете ли вы донести свою просьбу до Павла Дурова.

Для удобства пользователи пронумерованы от 0 до $N-1$, где 0 — id Павла, а $(N-1)$ — ваш id.

5.1 Формат ввода

В первой строке через пробел задано число пользователей в соцсети на данный момент N $(1 \leq N \leq 10^3)$, число возможных пар друзей M $\left(N-1 \leq M \leq \frac{N(N-1)}{2}\right)$.

Далее идет строка из N натуральных чисел k_i $(0 \leq k_i \leq N-1)$ через пробел — минимальное число общих друзей, необходимое, чтобы пользователь с номером i добавил вас в друзья.

В следующих M строках заданы ребра графа дружбы в формате «u_i v_i», где $(0 \leq u_i < v_i < N)$ — пара друзей u_i и v_i.

5.2 Формат вывода

Вывести `YES`, если удастся добавиться в друзья к Павлу Дурову, и `NO` иначе.

5.3 Комментарии

Пример ввода:

```
4 5
2 1 0 0
0 1
0 2
1 2
1 3
2 3
```

Пояснение ко второму примеру. Сначала мы можем добавиться в друзья к пользователю с id 2, потом к пользователю с id 1 и потом уже добраться до Павла.

6 В поисках нефти. 20 баллов

Задача предоставлена партнером Олимпиады — ПАО "Газпром-Нефть".

В ходе разведки месторождений нефти специалисты производят пробные бурения скважин и осуществляют анализ получаемых в ходе этого технических, геологических и геофизических данных. Целью этого является обнаружение нефтенасыщенных пластов, то есть пластов, содержащих в себе нефть и способных ее отдавать.

Перед вами стоит задача разработать алгоритм интеллектуального анализа реальных данных, подготовленных ПАО "Газпром-Нефть" , позволяющий наиболее качественно определять наличие или отсутствие нефтяных пластов на тех или иных глубинах залегания скважин.

Метрикой качества выступает точность нахождения нефтенасыщенного пласта

$$Accuracy = \frac{samples\_true}{samples\_all},$$

где $samples\_true$ - количество правильных предсказаний наличия/отсутствия нефтяного пласта, $samples\_all$ - общее количество записей в таблице.

6.1 Формат ввода

Ссылка на данные

$train\_wells.csv$ - файл с обучающими табличными данными, $test\_wells.csv$ - файл с тестовой выборкой. Файл с тренировочными табличными данными содержит информацию по 1000 скважинам, для каждой из которых имеется различная техническая, геологическая и геофизическая информация в виде следующих полей:

- **MD** — относительная глубина скважины (относительно поверхности бурения), всегда является положительной величиной, используется для привязки глубин внутри скважины, но не может выступать в роли какого-то признака при прогнозе (по крайней мере с физической точки зрения).
- **TVDSS** — глубина скважины относительно уровня моря, всегда является положительной величиной, может отражать поверхность геологического пласта или уровень водонефтяного контакта.
- **Layer** — название пласта, геологическая принадлежность интервала, качественная характеристика, выдаваемая геологом на основе его понимания геометрических характеристик целевого пласта, служащая для сопоставления пластов из различных скважин между собой.
- **GK** — гамма-каротаж, измеряет естественную радиоактивность пород, различные минералы имеют разное содержание радиоактивных материалов, как правило, чем выше — тем больше глинистая составляющая и меньше песчанистая, может измеряться в единицах API или мкр/ч.
- **NNKT_big** — нейтронный каротаж, регистрирует относительное водородосодержание, что может говорить о количестве пор в горных породах (они не могут быть пустыми и всегда содержат какой-то флюид, который в значительном объеме содержит в себе водород). Меньшие значения отвечают за более высокое флюидосодержание.
- **PS** — каротаж естественной поляризации, последняя возникает при фильтрации флюида через породу, уменьшение значений говори о наличии проницаемого интервала. Единица измерения — милливольты, может иметь совершенно разный масштаб в разных скважинах.

- **IK** — индукционный каротаж, отражает электрическую проводимость горных пород, величину обратную сопротивлению. Поскольку нефть является диэлектриком, а вода проводником, высокие показания отражают водонасыщенные пласты, а низкие — интервалы, вмещающие нефть. С другой стороны, плотные породы, не содержащие в себе пор, также имеют высокое сопротивление, поскольку не имеют в себе флюида, который способен проводить ток.

- **BK** — боковой зонд, отражает сопротивление горной породы, интерпретируется схожим образом с кривой индукционного каротажа, но уже наоборот, повышенные значения — нефть или плотные породы, пониженные — вода или глина.

- **PZ** — потенциал-зонд, отражает сопротивление горной породы, интерпретируется схожим образом с кривой индукционного каротажа, но уже наоборот, повышенные значения — нефть или плотные породы, пониженные — вода или глина. Схож с боковым зондом (BK), но имеет другую глубинность исследования.

- **Grad_zond** — другая группа зондов, отвечающих за сопротивление горных пород, в зависимости от числа в названии определяется глубинность метода. При бурении буровой раствор попадает в пласт и может изменить содержание того или иного флюида, поэтому, в теории, пониженые сопротивления в затронутой части пласта и повышенные в глубинной могут быть признаком наличия углеводородов.

- **target_collector** — бинарная характеристика выдаваемая специалистом по интерпретации каротажных данных, отвечающая за то, является ли тот или иной интервал коллекторским пластом, то есть пластом способным принимать и отдавать флюид.

- **target_oil** — бинарная характеристика выдаваемая специалистом по интерпретации каротажных данных, отвечающая за то, является ли тот или иной интервал коллекторским нефтенасыщенным пластом.

- **Well** — номер скважины.

В качестве целевой переменной выступает **target_oil**, которая при значении 1 говорит о наличии нефтенасыщенного пласта, а при значении 0 — о его отсутствии.

6.2 Формат вывода

Файл submission.csv, из одной колонки, в которой для каждой скважины из тестовой выборки стоит классифицирующая ее метка.

6.3 Комментарии

Баллы за посылку с $submission\_score$ очками начисляются по следующей формуле:

$$\left[19 \cdot \frac{submission\_score - baseline\_score}{max\_score - baseline\_score} + 1\right],$$

где $baseline\_score$ - очки за базовое решение, max_score — максимальные очки по всем посылкам всех участников. В течение контеста эта величина равна 1.0, а по окончании контеста вычислится ее реальное значение, и баллы всех посылок будут пересчитаны.

7 Штрихкоды. 20 баллов

Задача предоставлена партнером Олимпиады - компанией ABBYY.

Даны изображения и результат работы детектора штрихкодов на каждом из них в виде черно-белой карты сегментации. К сожалению, детектор часто ошибается и может в качестве гипотезы выделить несколько штрихкодов в одну гипотезу, выделить мусор или обрезать штрихкод. Насколько хорошо детектируем штрихкод определяем с помощью метрики IoU (Intersection over union), которая считается по пикселям: отношение площади области пересечения гипотезы с ground truth к площади области объединения гипотезы с ground truth. Общий результат по всем изображениям считается как среднее значение IoU по всем ground truth объектам (см. рисунок 2).

Требуется выделить штрихкоды на карте сегментации (найти координаты углов прямоугольника вокруг штрихкода) таким образом, чтобы IoU между результатом и эталоном был наибольшим.

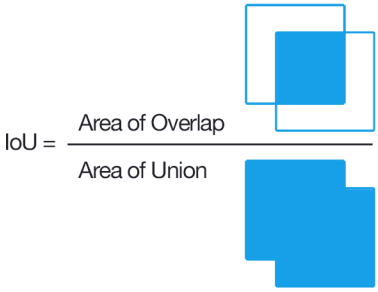

Рис. 2: Иллюстрация вычисления метрики IoU

7.1 Формат ввода

Дано: набор серых, черно-белых изображений. Для каждого изображения есть результат работы детектора: черно-белое изображение карты сегментации, черные пиксели у штрихкодов, белые – все остальное (см. пример на рисунке 3).

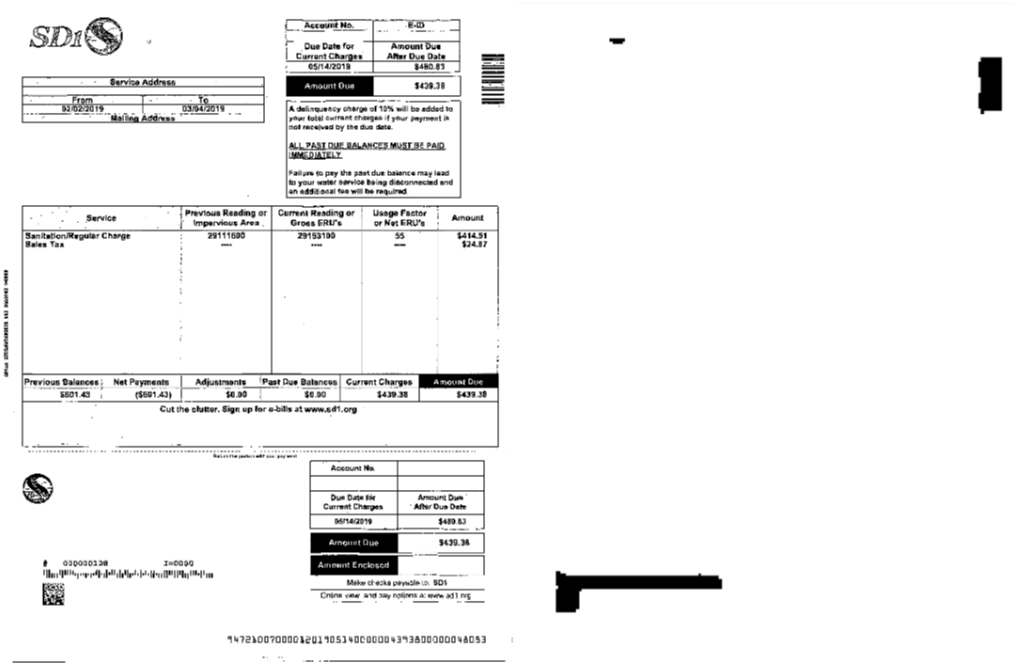

Рис. 3: Пример результата работы детектора, который выступает как входные данные (слева - исходное изображение, справа - карта сегментации).

7.2 Формат вывода

Один TSV файл следующего формата: имя файла с изображением, координаты одного объекта в виде 4х точек – углов гипотезы.

Пример для изображения выше:

0001.tif [3217, 689];[3217, 337];[3369, 337];[3369, 689]
0001.tif [289, 4001];[289, 3857];[441, 3857];[441, 4001]
0001.tif [285, 3825];[285, 3769];[1421, 3769];[1421, 3825]

Эти координаты описывают штрихкоды, показанные на рисунке 4

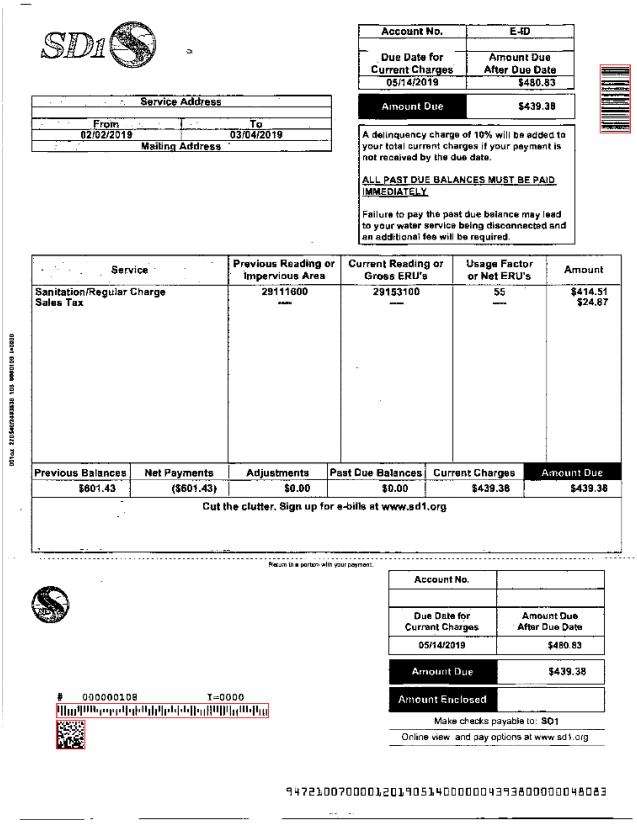

Рис. 4: Визуализация результата - прямоугольников, содержащих штрих-коды.

7.3 Комментарий

Баллы за посылку с *submission_score* очками начисляются по следующей формуле:

$$\left[19 \cdot \frac{submission\_score - baseline\_score}{max\_score - baseline\_score} + 1\right],$$

где *baseline_score* - очки за базовое решение, *max_score* — максимальные очки по всем посылкам всех участников. В течение контеста эта величина равна 1.0, а по окончании контеста вычислится ее реальное значение, и баллы всех посылок будут пересчитаны.

8 Распознай фейк. 20 баллов

Задача предоставлена партнером олимпиады — ПАО "Сбербанк".

Предлагается решить задачу бинарной классификации изображений для того, чтобы определить, на каких картинках изображены реальные люди, а какие сгенерированы нейронной сетью.

Для решения данной задачи предлагаются два набора данных: обучающий (train, размером 8 тысяч изображений) и тестовый (test, 12 тысяч изображений). Для каждого изображения из тестового набора нужно определить вероятность $pred$ того, что это изображение - фейковое.

Основной метрикой является метрика accuracy (доля верно классифицированных изображений). Предсказанный для изображения класс в случае $pred \geq 0.5$, это класс 1 (фейковое), в случае $pred < 0.5$ — это класс 0 (реальное изображение).

8.1 Формат ввода

Ссылка на данные

По ссылке выше лежат архив с изображениями, а также файл train.csv, в котором для изображений из обучающего набора train указаны истинные метки (где 1 — фейковое изображение, а 0 — настоящее).

8.2 Формат вывода

Результатом решения данной задачи должен быть файл submit.csv с двумя колонками — name и pred, где в первой будет название изображения из тестового набора test, а во второй — вероятность того, что это изображение является ненастоящим. Файл нужно отсортировать по колонке name.

8.3 Комментарии

Баллы за посылку с $submission\_score$ очками начисляются по следующей формуле:

$$\left[19 \cdot \frac{submission\_score - baseline\_score}{max\_score - baseline\_score} + 1\right],$$

где $baseline\_score$ - очки за базовое решение, max_score — максимальные очки по всем посылкам всех участников. В течение контеста эта величина равна 1.0, а по окончании контеста вычислится ее реальное значение, и баллы всех посылок будут пересчитаны.