

ОБЩИЙ ПОДХОД К ПОДВЕДЕНИЮ ИТОГОВ СОРЕВНОВАНИЙ ПО ПРОГРАММИРОВАНИЮ ПРИ ИСПОЛЬЗОВАНИИ РАЗЛИЧНЫХ СИСТЕМ ОЦЕНКИ

Аннотация

В настоящее время широкое распространение получили автоматические системы тестирования программных решений и проведения соревнований по программированию. В зависимости от назначения соревнований, могут использоваться различные системы оценки. Быстрое и простое добавление поддержки новых систем оценки является актуальной задачей.

В работе предложен метод подведения итогов соревнования, не зависящий от конкретной системы оценки. Реализация предложенного подхода произведена в системе автоматического проведения соревнований PCMS, разработанной при участии автора. Проведено сравнение системы с альтернативными системами.

Ключевые слова: тестирование программ, система оценки, автоматическая проверяющая система.

ВВЕДЕНИЕ

В последнее время широкое распространение получили различные соревнования по программированию [1]. Для их проведения используются автоматизированные системы управления соревнованиями [2, 3, 4], они применяются на олимпиадах, в процессе обучения школьников и студентов, тестирования абитуриентов и кандидатов на работу [1].

В зависимости от типа соревнований, используются различные подходы к формированию таблицы результатов. Универсальная система автоматического тестирования должна поддерживать различные системы оценки соревнований и построения таблицы результатов. При этом по мере развития соревнований и расширения сферы их использования, появляются все новые системы оценки, а также происходит модификация использованных ранее систем. Это влечет необходимость модификации системы проведения соревнований для поддержки новых систем или новых настроек старых систем.

В данной работе предлагается метод подведения итогов соревнований по программированию, не зависящий от используемой системы оценки и описывается реализация данного метода в системе автоматического тестирования PCMS. Проводится сравнение метода с использованными в других системах автоматического тестирования.

ОБЗОР РАЗЛИЧНЫХ СИСТЕМ ОЦЕНКИ

Рассмотрим наиболее широко распространенные системы оценки.

СИСТЕМА СТУДЕНЧЕСКОГО ЧЕМПИОНАТА МИРА ПО ПРОГРАММИРОВАНИЮ АСМ

Эта система получила краткое название «Система АСМ». Каждая из предложенных на соревновании задач оценивается одинаково. Оцениваются только полные решения задач. Учитывается количество неправильных попыток по каждой задаче до первой правильной попытки. Учитывается время отправки решения.

Система АСМ используется в финале студенческого чемпионата мира по программированию АСМ [5], на полуфинальных соревнованиях чемпионата мира, в том числе на Северо-Восточном Европейском полуфинале, который является чемпионатом России [6, 7], на Всероссийской командной олимпиаде школьников по программированию [8, 9], на большом количестве различных региональных соревнований, в том числе чемпионате Урала [10], Всесибирской олимпиаде [11], на Петрозаводских учебно-тренировочных сборах [12, 13], кубке Russian CodeCup, проводимом Mail.ru group совместно с СПбГУ ИТМО [14], и многих других. Система АСМ используется на многих интернет-сайтах, предлагающих онлайн-участие в соревнованиях [15, 16, 17], в том числе на интернет-олимпиадах по информатике [18, 19].

Описание системы

Основным результатом участника является количество решенных задач, выше в таблице классифицируется участник, решивший больше задач.

При равенстве количества решенных задач анализируется *штрафное время*. Штрафное время вычисляется следующим образом: для каждой задачи оно полагается равным количеству минут прошедшее от начала соревнования до момента решения задачи плюс 20 минут за каждую неверную попытку сдать задачу. За нерешенные задачи штрафное время полагается равным нулю. Штрафное время участника равно сумме значений штрафного времени по всем задачам. При равном количестве задач выше оказывается участник, имеющий меньшее штрафное время.

При равенстве как количества решенных задач, так и штрафного времени используется третий критерий: время от начала соревнования до последней успешной попытки – чем меньше эта величина, тем выше классифицируется участник.

Участникам во время соревнования сообщаются результаты проверки их решений на тестах жюри. Участники могут, основываясь на этой информации, исправлять свои

решения, в случае, если они не приняты как правильные, и отправлять их снова на проверку. Обычно количество попыток не ограничено.

Параметры и модификации системы

При измерении штрафного времени и времени от начала соревнования до последней успешной попытки возможны различные подходы с округлением до минут.

– округление вниз (используется с формулировкой: количество минут, прошедших от начала соревнования до момента сдачи задачи);

– отсутствие округления – используется вещественное число (обычно используется точность до миллисекунд);

– округление по правилам математики.

Константа в 20 минут за неверную попытку сложилась исторически и считается достаточно сбалансированной для соревнования продолжительностью в 5 часов. Для соревнований другой продолжительности возможно изменение этой константы, хотя обычно она не меняется.

Иногда некоторые неверные попытки не добавляют штрафа. Обычно не добавляют штрафа за неверные попытки, сделанные после верной попытки. Исторически иногда неверные попытки добавляли штраф, даже если они были сделаны после успешной. Это правило было изменено для исправления следующей ситуации: в конце соревнования создается большая очередь из попыток, ожидающих проверки. В этой ситуации результат проверки может не быть сообщен участнику до конца соревнования. Участник, не имея возможности узнать, правильное ли у него решение, может отправить несколько различных модификаций решения. Модификации, отправленные после первого правильного, сейчас обычно не учитываются.

При проведении соревнований через интернет, когда нет возможности обеспечить эквивалентность настроек компиляторов на компьютерах участников и на компьютерах проверяющей системы, решения, получающие вердикт «ошибка компиляции», обычно не считаются неверными попытками и не увеличивают штрафное время.

КЛАССИЧЕСКАЯ СИСТЕМА МЕЖДУНАРОДНОЙ ОЛИМПИАДЫ ШКОЛЬНИКОВ ПО ИНФОРМАТИКЕ IOI

Эта система получила краткое название «Система IOI». Каждая из предлагаемых задач оценивается некоторым количеством баллов. Оцениваются частичные решения, каждое решение запускается на наборе тестов, баллы за тесты суммируются. Время отправки решения не учитывается. Неправильные попытки по задачам не учитываются.

Система IOI использовалась до 2009 года на международной олимпиаде школьников по информатике [20], до 2010 года использовалась на Всероссийской олимпиаде школьников по информатике [21], используется на большинстве национальных олимпиад, включая олимпиаду США USACO [22], используется на многих интернет-соревнованиях, ориентированных на школьников, в том числе личных интернет-олимпиадах [18, 19] и многих других.

Описание системы

Все задачи оцениваются независимо. Обычно решение запускается на наборе тестов, тест засчитывается, если выполнены все ограничения из условия задачи и выдан правильный ответ. При этом различные тесты оцениваются независимо. Каждый тест оценивается некоторым количеством баллов. Баллы за задачу равны сумме баллов за тесты.

Результат участника представляет собой сумму баллов за задачи. Выше оказывается участник, набравший больше баллов. Участники, набравшие одинаковое количество баллов, считаются разделившими место.

По каждой задаче оценивается последнее принятое на проверку решение. При использовании автоматизированной системы проверки решений решение обычно принимается на проверку, если оно успешно проходит предварительную проверку. Предварительная проверка заключается в том, что решение запускается на наборе тестов, приведенных в условии задачи, и требуется, чтобы оно прошло каждый из них.

Участнику во время соревнования сообщается только результат предварительной

проверки. Результат окончательной проверки и баллы за задачи не сообщаются.

Параметры и модификации системы

Основным параметром системы является распределение баллов по задачам. Исторически задачи оценивались пропорционально своей сложности, однако в последнее время чаще используется система, когда все задачи оцениваются одинаково. Наиболее часто все задачи оцениваются в 100 баллов, такая система принята на международной олимпиаде по информатике и Всероссийской олимпиаде по информатике.

Основной модификацией системы оценки является введение групп тестов, когда тесты оцениваются не независимо, а по группам. Обычно баллы за группу назначаются в случае, когда все тесты из этой группы пройдены, но иногда используются и другие подходы, например экспоненциальное убывание баллов за группу, в зависимости от количества непройденных тестов в ней, и другие.

На некоторых олимпиадах, например Индивидуальной олимпиаде по информатике и программированию [23] или Открытой олимпиаде по программированию [24] во время соревнования участникам сообщается дополнительная информация о результатах проверки. При этом может использоваться ограничение на количество просмотров результатов проверки.

НОВАЯ СИСТЕМА МЕЖДУНАРОДНОЙ ОЛИМПИАДЫ ШКОЛЬНИКОВ ПО ИНФОРМАТИКЕ IOI

В 2010 году на международной олимпиаде школьников по информатике система оценки была радикально изменена. При этом основные изменения коснулись процесса проведения олимпиады и системы начисления баллов за задачу, но не схемы подведения итогов. По-прежнему задачи оцениваются баллами, баллы за задачи суммируются, результат участника равен сумме баллов за задачи, выше оказывается участник, набравший больше баллов.

Рассмотрим основные изменения по сравнению с классической системой IOI.

Изменения в процессе проведения

Участники имеют возможность посмотреть результаты проверки во время тура. Степень детализации результатов может варьироваться, вплоть до показа подробных результатов по каждому тесту.

Оценивается лучшее из двух решений: последнего отправленного на проверку и лучшего решения, по которому участнику известны баллы.

Изменения в методике оценивания решения

Задача разбивается на подзадачи. Каждая подзадача оценивается отдельно и представляет собой обычно исходную задачу меньшей размерности или с дополнительными ограничениями. Участник обычно получает баллы за подзадачу только в случае, если все тесты, соответствующие подзадаче, пройдены. Возможны и более сложные способы оценки подзадачи.

Параметры и модификации системы

Основными параметрами конфигурации системы оценки являются ограничения на просмотр своих результатов. Используются различные подходы.

На международной олимпиаде по информатике используется система «токенов». Каждой задаче соответствуют два токена, для просмотра результатов тестирования необходимо использовать один из токенов. Он восстанавливается через определенный промежуток времени (обычно, 30 минут).

На Всероссийской олимпиаде используются два ограничения: максимальное количество просмотров результатов по задаче и минимальный интервал между двумя просмотрами результатов по задаче.

СИСТЕМЫ С БАЛЛАМИ ЗА ЗАДАЧУ, ЗАВИСЯЩИМИ ОТ ВРЕМЕНИ (TOPCODER, CODEFORCES)

Некоторые интернет-соревнования, в частности соревнования серии Topcoder [25] и соревнования на сайте Codeforces [26], используют систему, в которой за каждую задачу начисляются баллы, но количество баллов, полученных за задачу, зависит от вре-

мени, прошедшего с начала соревнования до сдачи задачи.

Описание системы

Все задачи оцениваются независимо. Оцениваются только полностью правильные решения. Правильное решение получает баллы, в зависимости от времени, прошедшего с начала соревнования.

Результат участника представляет собой сумму баллов за задачи. Выше оказывается участник, набравший больше баллов. Участники, набравшие одинаковое количество баллов, считаются разделившими место.

По каждой задаче оценивается последнее принятое на проверку решение. За неправильные решения возможны штрафы.

Участнику во время соревнования может сообщаться как только результат предварительной проверки, так и результат окончательной проверки.

Параметры и модификации системы

Основными параметрами являются закон изменения баллов, штрафы за неверные попытки, дополнительные источники баллов.

Topcoder использует следующую формулу для определения баллов за задачу:

$$S = MP \times \left(0.3 + \frac{0.7 \times TT^2}{10PT^2 + TT^2} \right),$$

где MP – максимальные баллы за задачу, TT – продолжительность соревнования, а PT – время, потраченное на решение задачи. При этом время отсчитывается от момента «открытия» задачи, когда участник получает возможность читать условие. За неверные попытки начисляется штраф в 10% от стоимости задачи, но баллы за правильно решенную задачу никогда не опускаются ниже 30% максимального балла.

На Codeforces используется линейное убывание баллов за задачу от 100% до 30%, штрафы за неправильные попытки не начисляются.

И на Topcoder и на Codeforces, помимо решения задач можно набирать дополнительные баллы, подбирая тесты, на которых решения других участников работают неверно.

Модель подведения итогов соревнований
Сравнительный анализ различных систем оценки распространенных соревнований позволяет выделить следующие общие моменты во всех системах:

- задачи оцениваются независимо;
- результат по каждой задаче однозначно задается набором подходов участника по этой задаче и их результатов проверки;
- результат участника однозначно задается набором его результатов по задачам;
- для ранжирования участников достаточно знать их результаты.

Это позволяет рассмотреть следующую последовательность подведения итогов соревнования:

- для каждого участника по каждой задаче формируется список подходов и их результатов проверки;

- для каждой задачи каждого участника на базе списка подходов получается результат по данной задаче;

- для каждого участника на основании результатов по всем задачам формируются результаты данного участника;

- участники ранжируются с использованием их результатов.

Процесс подведения итогов проиллюстрирован на рис. 1.

Для окончательной формализации необходимо сформулировать понятия «результат проверки подхода», «результат участника по задаче» и «результат участника». Каждое из этих понятий можно представить как набор полей, каждое из которых содержит некоторое значение.

Приведем наборы полей и алгоритмы их обработки для получения результатов для основных систем оценки.

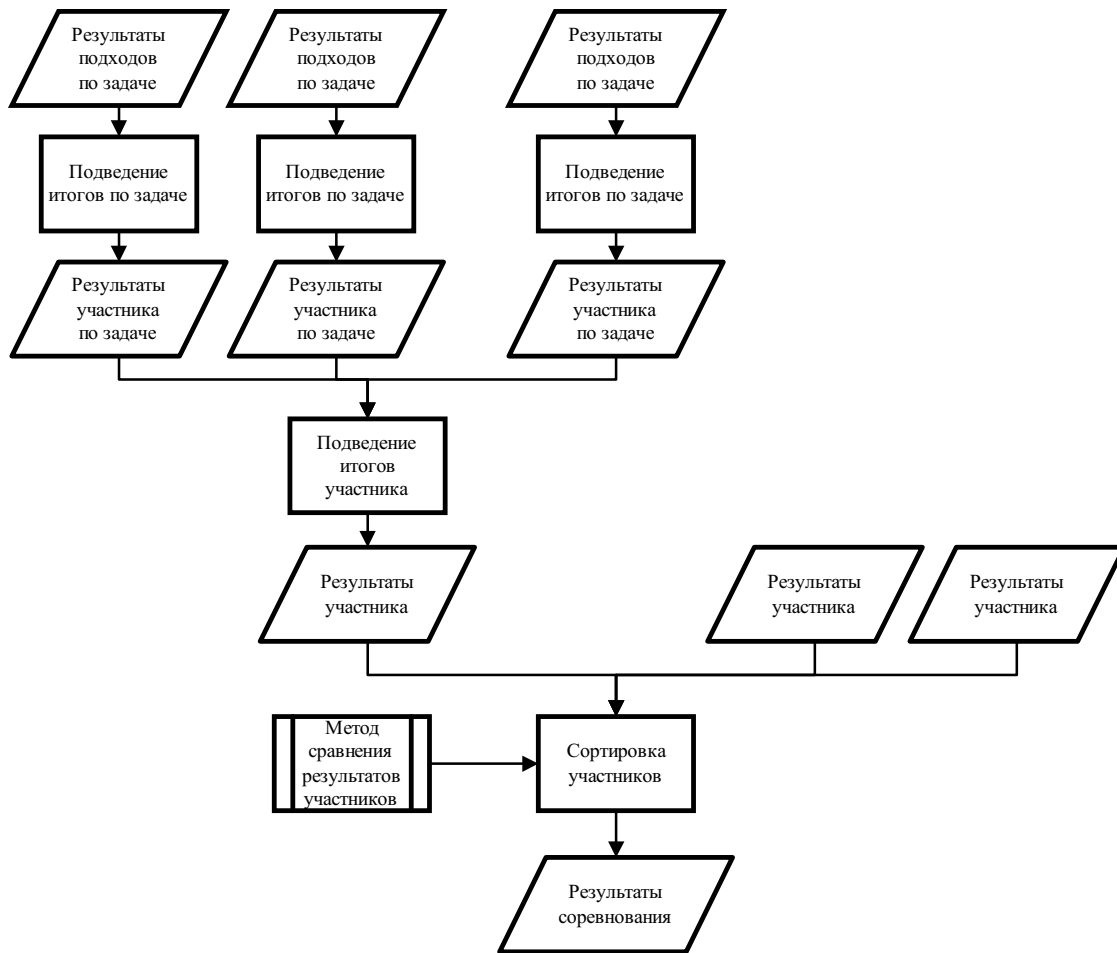


Рис. 1. Процесс подведения итогов соревнования

СИСТЕМА СТУДЕНЧЕСКОГО ЧЕМПИОНАТА МИРА ПО ПРОГРАММИРОВАНИЮ АСМ

Результаты проверки подхода

– *time*: время от начала соревнования, когда сделан подход;

– *outcome*: результат проверки подхода, с точки зрения подведения итогов интересует в основном, верно ли, что решение принято. В случае, если попытки с некоторыми результатами (например, «ошибка компиляции») не учитываются, может требоваться и более подробная информация.

Результаты по задаче

– *accepted*: верно ли, что задача решена;

– *time*: время от начала соревнования, до момента сдачи задачи, если она сдана;

– *attempts*: количество попыток, которое потребовалось для сдачи задачи;

– *penalty*: штрафное время за задачу.

Алгоритм получения по набору подходов участника по задаче результатов по задаче

Подходы по задаче сортируются по времени (параметру *time*), если подходы с некоторыми результатами (например, «ошибка компиляции») не учитываются, то они удаляются.

Подходы просматриваются по очереди, если очередной подход верный, то поле *accepted* устанавливается как «истина», поле *time* устанавливается равным полю *time* соответствующего подхода, поле *attempts* устанавливается равным номеру подхода в отсортированном списке, поле *penalty* устанавливается равным $time + 20 \times (attempts - 1)$.

Если верного подхода найдено не было, поле *accepted* устанавливается как «ложь», поле *attempts* устанавливается равным количеству подходов в списке, поля *time* и *penalty* устанавливаются равными 0.

Результаты участника

– *solved*: количество решенных задач;

– *penalty*: штрафное время;

– *last_time*: время от начала соревнования до последнего успешного результата по задаче.

Алгоритм получения по набору результатов участника по задачам полных результатов участника

Поле *solved* устанавливается равным количеству задач, у которых поле *accepted* установлено как «истина».

Поле *penalty* устанавливается как сумма значений поля *penalty* всех задач.

Поле *last_time* устанавливается как максимум значений поля *time* всех задач.

Алгоритм ранжирования участников на основании их результатов

Участники сравниваются по полю *solved*, участник с большим значением поля оказывается выше.

При равенстве поля *solved* участники сравниваются по полю *penalty*, участник с меньшим значением поля оказывается выше.

При равенстве полей *solved* и *penalty* участники сравниваются по полю *last_time*, участник с меньшим значением поля оказывается выше.

При равенстве всех трех полей участники считаются разделившими место.

КЛАССИЧЕСКАЯ СИСТЕМА МЕЖДУНАРОДНОЙ ОЛИМПИАДЫ ШКОЛЬНИКОВ ПО ИНФОРМАТИКЕ IOI

Результаты проверки подхода

– *time*: время от начала соревнования, когда сделан подход;

– *outcome*: результат предварительной проверки подхода, с точки зрения подведения итогов интересует в основном, верно ли, что решение принято на окончательную проверку;

– *score*: баллы, которые набирает данное решение на основном наборе тестов.

Результаты по задаче

– *score*: баллы за задачу.

Алгоритм получения по набору подходов участника по задаче результатов по задаче

Среди подходов по задаче, которые были приняты на проверку (что устанавливается по полю *outcome*), выбирается последний по времени (с максимальным полем *time*).

Поле score устанавливается равным полю score этого подхода, если ни одного подхода не было принято на проверку, поле score устанавливается равным 0.

Результаты участника

– *score*: баллы участника.

Алгоритм получения по набору результатов участника по задачам полных результатов участника

Поле score устанавливается равным сумме полей score всех задач.

Алгоритм ранжирования участников на основании их результатов

Участники сравниваются по полю score, участник с большим значением поля оказывается выше.

При равенстве поля score участники считаются разделившими место.

**НОВАЯ СИСТЕМА МЕЖДУНАРОДНОЙ
ОЛИМПИАДЫ ШКОЛЬНИКОВ
ПО ИНФОРМАТИКЕ IOI**

Результаты проверки подхода

– *time*: время от начала соревнования, когда сделан подход;

– *outcome*: результат предварительной проверки подхода, с точки зрения подведения итогов интересует в основном, верно ли, что решение принято на окончательную проверку;

– *score*: баллы, которые набирает данное решение на основном наборе тестов;

– *feedback_status*: информация о том, известны ли участнику баллы за этот подход.

Результаты по задаче

– *score*: баллы за задачу.

Алгоритм получения по набору подходов участника по задаче результатов по задаче

Среди подходов по задаче, которые были приняты на проверку (что устанавливается по полю outcome), выбирается последний по времени (с максимальным параметром time).

Среди подходов по задаче, которые были приняты на проверку (что устанавливается по полю outcome) и информация о результатах проверки которых известна участнику (что

устанавливается по полю feedback_status), выбирается лучший (с максимальным полем score). Если ни одного такого подхода нет, используется только последний принятый на проверку подход.

Поле score устанавливается равным максимуму значений поля score из двух выбранных подходов. Если по задаче ни одного подхода не было принято на проверку, поле score устанавливается равным 0.

Результаты участника

– *score*: баллы участника.

Алгоритм получения по набору результатов участника по задачам полных результатов участника

Поле score устанавливается равным сумме полей score всех задач.

Алгоритм ранжирования участников на основании их результатов

Участники сравниваются по полю score, участник с большим значением поля оказывается выше.

При равенстве поля score участники считаются разделившими место.

**РЕАЛИЗАЦИЯ РАЗЛИЧНЫХ СИСТЕМ
С ПОМОЩЬЮ ПЛАГИНОВ
В МНОГОКОМПОНЕНТНОЙ
АВТОМАТИЧЕСКОЙ СИСТЕМЕ
ТЕСТИРОВАНИЯ РСМС**

Как показано в предыдущем разделе, возможна простая формализация различных систем оценки с помощью введения сущностей «результат подхода», «результат по задаче», «результат участника» и «метод сравнения участников». В работах [1, 2, 3, 4] излагается метод построения автоматической системы тестирования на базе системы взаимодействующих компонентов. Реализацией этой идеи является проверяющая система РСМС, используемая для проведения Северо-восточного полуфинала чемпионата мира по программированию, Всероссийской командной олимпиады школьников по программированию, Всероссийской олимпиады школьников по программированию и многих других олимпиад.

Рассмотрим реализацию изложенного подхода к построению таблиц результатов в рамках такой системы. Для оценивания участников и формирования таблицы результатов используется одна компонента: *StandingsGenerator* (генератор результатов). Каждое соревнование в системе характеризуется параметром *scoringModel* (модель оценки). Участие в соревновании для каждого участника соответствует объект типа *Session* (сессия).

Для каждой модели оценки в рамках построения таблицы результатов необходимо реализовать следующие классы:

- Хранение данных:

RunResult для хранения результатов проверки подхода;

SessionProblemResult для хранения результатов участника по задаче;

SessionResult для хранения результатов участника;

Standings для хранения результатов соревнования с отсортированным списком участников.

- Обработка данных:

RunResultCalculator для получения по подходу его результатов проверки;

SessionProblemCalculator для получения по списку подходов результатов участника по задаче;

SessionCalculator для получения по результатам участника по задачам его результатов;

Соответствующие классы представляют собой плагин для компонента *StandingGenerator* и не требуют изменения логики его работы при добавлении новой системы оценки или изменении старой.

СРАВНЕНИЕ

С СУЩЕСТВУЮЩИМИ РЕШЕНИЯМИ

Сравнение с существующими решениями затруднено малым количеством документации и закрытостью кода большинства проверяющих систем. Тем не менее, часть проверяющих систем распространяется свободно и можно проанализировать реализацию подхода к построению таблицы результатов.

К сожалению, объективные показатели, позволяющие сравнивать между собой различные подходы к построению таблицы результатов, выделить трудно. В качестве критериев для сравнения были выделены возможность добавления систем оценки без изменения исходного кода, ответственного за подведение итогов соревнования и размер исходного кода, осуществляющего подведение итогов. Данные критерии не являются универсальными, особенно с учетом использования различных языков программирования для написания проверяющих систем, но позволяют приблизительно оценить трудозатраты разработчика по внедрению новой системы оценки.

Приведем сначала статистику по системе PCMS, реализующей описанную в данной работе идею.

Система PCMS

Система PCMS разработана при участии автора работы в СПбГУ ИТМО. Система написана на языке Java. Система используется для проведения полуфинала чемпионата мира по программированию, Всероссийской олимпиады школьников по программированию, Всероссийской командной олимпиады школьников по программированию, интернет-олимпиад школьников по информатике и программированию и многих других олимпиад.

Реализация построения таблицы результатов происходит с использованием предложенного в данной работе метода. Построение таблицы результатов и ее форматирование для вывода разделены, используется специальное промежуточное представление. Компонента *StandingsGenerator* осуществляет обобщенное формирование результатов, не зависящее от используемой системы оценки. Общий размер файла: 17832 байта, 555 строк. Количество поддерживаемых систем оценки не ограничено, реализована поддержка двух систем оценки (АСМ и IOI, в том числе современной версии). Суммарный размер файлов для каждой системы оценки около 5–10 килобайт, около 200 строк. Возможно добавление новых систем оценки без модификации исходного кода.

Рассмотрим теперь известные системы с открытым исходным кодом.

Система eJudge

Система eJudge [27] разработана Александром Черновым, МГУ. Система написана на Си. Система активно используется на различных олимпиадах, проводимых в Москве [28], а также в летней компьютерной школе [29].

Анализ производится для последней на момент написания работы версии 2.3.18.

Реализация построения таблицы результатов в каждой из систем оценки происходит независимо, переиспользования кода не происходит. Файл `html.c`, ответственный за вывод таблицы результатов, осуществляет все действия: получение по подходам результатов участника, сортировку участников, форматирование результатов в форме HTML-таблицы, вывод таблицы. Общий размер файла – 5518 строк, 189879 байт, количество поддерживаемых систем оценки: 4 (АСМ, IOI и две специальные системы: Kirov – в настоящее время используемая редко система оценки, в которой участники получают баллы за частичные решения пропорциональные количеству пройденных тестов, но учитывается и штрафное время в качестве вторичного критерия, и Moscow – модификация системы IOI с группами тестов, где часть баллов по задаче (обычно 40% и 70%) можно получить, лишь решив полностью некоторую базовую подзадачу, а остальные баллы, в случае решения базовой подзадачи, можно получить за прохождение отдельных тестов).

Есть общий метод построения таблицы результатов, который использует информацию из объекта, содержащего соревнование, для принятия решения о том, какой метод построения таблицы результатов запустить. Список систем оценки и их соответствие способам вывода таблицы результатов находится в коде и без модификации кода добавление новых способов оценки невозможно.

Система DOMjudge

Система DOMjudge [30, 31] разработана в Голландии, в университете Утрехта. Сис-

тема написана на PHP. Система используется в ряде полуфиналов чемпионата мира по программированию.

Анализ производится для последней на момент написания работы версии 3.1.3. Система поддерживает только систему оценки АСМ.

Для построения таблицы результатов используются общение напрямую с базой данных, осуществляющей хранение результатов подходов. Файл `scoreboard.php`, ответственный за построение таблицы результатов, содержит функции, ответственные за два этапа: выборка из базы данных и формирование данных с результатами и построение таблицы с ее выводом. Однако метод вывода таблицы и промежуточный формат хранения результатов заточены на систему оценки АСМ, поэтому простая адаптация для других систем оценки, по-видимому невозможна. Размер файла: 23210 байт, 638 строк, количество поддерживаемых систем оценки: 1 (АСМ).

Возможность добавления системы оценки без модификации кода отсутствует.

Система dudge

Система dudge [32] разработана в Самарском государственном техническом университете. Система написана на Java с использованием технологии JavaEE.

Анализ производится для последней на момент написания работы версии, взятой из репозитория на сайте [32].

В системе абстрагировано понятие MonitorRecord для описания результата пользователя в соревновании. Реализованы интерфейсы для систем оценки АСМ и IOI (в системе называется School). Однако для доступа к вариантам системы оценки в общем интерфейсе доступа к соревнованию DudgeLocal для каждой из систем оценки реализован свой метод, что ведет к тому, что добавление новых систем оценки требует модификации интерфейса. Размер файла с реализацией системы оценки около 7000 байт, около 200 строк. Построение таблицы результатов по готовым записям с системой оценки осуществляется с использованием простой сортировки.

Проведем также анализ еще двух систем, исходные тексты которых недоступны.

Система TestSys

Система TestSys разработана в Санкт-Петербургском государственном университете. Используется для проведения соревнований в СПбГУ, учебно-тренировочных сборов кандидатов в сборную России по информатике, сборов команд-участников чемпионата мира по программированию в Петрозаводске. Система написана на Delphi.

Реализована поддержка систем АСМ и IOI, в том числе современной системы IOI с обратной связью. Предоставляются очень гибкие возможности конфигурирования настроек соревнования. Добавление новой системы, вероятно, требует существенной модификации кода.

Система KATTIS

Система KATTIS [33] разработана в Королевском технологическом институте,

Стокгольм, Швеция. Используется на финале чемпионата мира по программированию. Свободно не распространяется. Вероятно, поддерживает только систему АСМ.

Итоговое сравнение систем автоматического тестирования по критерию работы с различными системами оценки приведено в табл. 1.

ЗАКЛЮЧЕНИЕ

Широкое распространение получили автоматические системы тестирования программных решений и проведения соревнований по программированию. В зависимости от назначения соревнования могут использоваться различные системы оценки. Добавление поддержки новых систем оценки является актуальной задачей.

В работе предложен метод подведения итогов соревнования, не зависящий от конкретной системы оценки. Предложенный ме-

Табл. 1. Сравнение различных проверяющих систем по критерию работы с различными системами оценки

Система тестирования	PCMS	EJudge	DOM-judge	dudge	TestSys	KATTIS
Открытый код	нет	да	да	да	нет	нет
Свободно распространяется	да	да	да	да	нет	нет
Язык программирования	Java	Си	PHP	Java	Delphi	неизвестно
Поддержка систем оценки	не ограничено, реализовано 2	4	1	2	2	1?
Возможность добавления новых систем оценки без модификации кода	да	нет	нет	нет	нет?	неизвестно
Размер кода для подведения итогов соревнования (байт/строка)	~27000 ~900	189879 5518	23210 638	–	неизвестно	неизвестно
Размер переиспользуемого кода, не изменяемого при добавлении систем тестирования (байт/строка)	17832 555	–	–	–	неизвестно	неизвестно
Типичный размер кода для поддержки системы тестирования (байт/строка)	~5000 ~200	–	–	~7000 ~200	неизвестно	неизвестно

тод заключается в абстрагировании понятий «результат проверки подхода», «результат участника по задаче», «результат участника» и «метод сравнения участников». Данный подход позволяет добавлять новые системы оценки с помощью плагинов, реализующих указанные понятия.

Реализация предложенного подхода произведена в системе автоматического проведения соревнований PCMS, разработанной при участии автора. Проведено сравнение системы с альтернативными системами, что позволяет сделать вывод о целесообразности использования разработанного подхода с точки зрения снижения затрат разработчика по внедрению новой системы оценки.

Литература

1. Васильев В.Н., Елизаров Р.А., Казаков М.А., Парфенов В.Г., Станкевич А.С. Использование олимпиад и творческих конкурсов при подготовке высококвалифицированных кадров в области информационных технологий // Труды XI Всероссийской научно-методической конференции «Телематика-2004». СПб.: СПбГУ ИТМО, 2004. С. 437–438.
2. Елизаров Р.А., Станкевич А.С. Система управления соревнованиями по программированию как система обработки данных // Телекоммуникации и информатизация образования, 2003. № 3. С. 64–85.
3. Корнеев Г.А., Станкевич А.С. Методы тестирования решений задач на соревнованиях по программированию // Труды II межвузовской конференции молодых учёных. СПб.: СПбГУ ИТМО, 2005. С. 36–40.
4. Корнеев Г.А., Маврин П.Ю., Станкевич А.С. Построение компонентных систем с поддержкой динамической реконфигурации // Труды XVI Всероссийской научно-методической конференции «Телематика-2009». СПб.: СПбГУ ИТМО, 2009. С. 289.
5. Сайт чемпионата мира по программированию // <http://icpc.baylor.edu>.
6. Сайт Северо-восточного Европейского полуфинала чемпионата мира по программированию // <http://neerc.ifmo.ru>
7. Командный чемпионат мира по программированию ACM 2010/2011. Северо-Восточный Европейский регион / Под ред. проф. В.Н. Васильева и проф. В.Г. Парфенова. СПб.: СПбГУ ИТМО, 2010. 244 с.
8. Сайт Всероссийской командной олимпиады школьников по программированию // <http://neerc.ifmo.ru/school>.
9. Одиннадцатая Всероссийская олимпиада школьников по информатике и программированию / Под ред. В.Н. Васильева, В.Г. Парфенова, А.С. Станкевича. СПб.: СПбГУ ИТМО, 2010. 262 с.
10. Сайт чемпионата Урала // <http://acm.usu.ru>.
11. Сайт Всесибирской олимпиады им. Подтосина // <http://olimpic.nsu.ru>.
12. Сайт Петрозаводских учебно-тренировочных сборов // <http://karelia.snarknews.info>.
13. Denis Vlasov. The Open Cup and Petrozavodsk Programming Training Camp // Proceedings of Collaborative Learning Institute Symposium, 2010.
14. Сайт Russian Code Cup // <http://russiancodecup.ru>.
15. Сайт интернет-соревнований университета Вальядолид, Испания // <http://acm.uva.es>.
16. Сайт интернет-соревнований Уральского государственного университета // <http://acm.timus.ru>.
17. Сайт интернет-соревнований Саратовского государственного университета // <http://acm.sgu.ru>.
18. Сайт интернет-олимпиад по информатике // <http://neerc.ifmo.ru/school/io>.
19. Станкевич А.С., Харченко Т.В. Всероссийские интернет-олимпиады по информатике и программированию // Компьютерные инструменты в образовании, 2002. № 3–4. С. 60–70.
20. Сайт международной олимпиады по информатике // <http://ioinformatics.org>.
21. Кирюхин В.М. Информатика. Всероссийские олимпиады. Вып. 1. М.: Просвещение, 2008.
22. Сайт олимпиад школьников по информатике США // <http://www.uwp.edu/sws/usaco>.
23. Сайт индивидуальной олимпиады по информатике и программированию // <http://neerc.ifmo.ru/school/oiip>.
24. Сайт открытой олимпиады школьников по программированию // <http://olympiads.ru/zaoch>.
25. Сайт соревнований по программированию TopCoder // <http://topcoder.com/tc>.
26. Сайт соревнований по программированию CodeForces // <http://codeforces.ru>.

27. Сайт системы проведения соревнований по программированию EJudge // <http://ejudge.ru>.
28. Сайт московских олимпиад по информатике // <http://acm.msu.ru>.
29. Сайт летней компьютерной школы // <http://lksh.ru>.
30. Сайт проверяющей системы DOMjudge // <http://domjudge.sourceforge.net>.
31. Tobias Werth. DOMjudge – An Automated Judging System for Programming Contests // Proceedings of Collaborative Learning Institute Symposium, 2010.
32. Сайт проверяющей системы dudge // <http://code.google.com/p/dudge>.
33. Сайт проверяющей системы KATTIS // <http://kattis.csc.kth.se>.

Abstract

Many automatic systems for programming contest management and automated program testing are currently used for various contests. Different scoring models for the contest can be used dependent on the purpose of the contest. Quickly and easily adding new scoring models is essential for multipurpose testing system.

This article describes general approach to creating standings for the contest independent of its scoring model. Implementation of this approach in PCMS testing system is described and compared to other testing systems.

Keywords: program testing, scoring model, automated testing system.



Наши авторы, 2011.
Our authors, 2011.

*Станкевич Андрей Сергеевич,
доцент кафедры «Компьютерные
технологии» СПбГУ ИТМО,
stankev@gmail.com.*