ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

Специальный курс №6

Код модуля
1161052(1)

Модуль
Специальные курсы 7 семестра

Екатеринбург

Оценочные материалы составлены автором(ами):

№ п/п	Фамилия, имя, отчество	Ученая степень, ученое звание	Должность	Подразделение
1	Домашних Иван Алексеевич	без ученой степени, без ученого звания	Старший преподаватель	департамент математики, механики и компьютерных наук

Согласовано:

Управление образовательных программ Ю.Д. Маева

Авторы:

- **Домашних Иван Алексеевич, Старший преподаватель, департамент математики, механики и компьютерных наук**

1. СТРУКТУРА И ОБЪЕМ ДИСЦИПЛИНЫ Специальный курс №6

1.	**Объем дисциплины в зачетных единицах**	6	
2.	**Виды аудиторных занятий**	Лекции Практические/семинарские занятия	
3.	**Промежуточная аттестация**	Экзамен	
4.	**Текущая аттестация**	Домашняя работа	2

2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ (ИНДИКАТОРЫ) ПО ДИСЦИПЛИНЕ МОДУЛЯ Специальный курс №6

Индикатор – это признак / сигнал/ маркер, который показывает, на каком уровне обучающийся должен освоить результаты обучения и их предъявление должно подтвердить факт освоения предметного содержания данной дисциплины, указанного в табл. 1.3 РПМ-РПД.

Таблица 1

Код и наименование компетенции	**Планируемые результаты обучения (индикаторы)**	**Контрольно-оценочные средства для оценивания достижения результата обучения по дисциплине**
1	**2**	**3**
ПК-1 -Способен демонстрировать общенаучные базовые знания в математических и естественных науках, фундаментальной информатики и информационных технологиях	З-1 - Сделать обзор базовых понятий в математических и естественных науках, фундаментальной информатики и информационных технологиях П-1 - Иметь практический опыт сбора информации в математических и естественных науках, фундаментальной информатики и информационных технологиях У-1 - Обобщать полученные знания в математических и естественных науках, фундаментальной информатики и информационных технологиях	Домашняя работа № 1 Домашняя работа № 2 Лекции Практические/семинарские занятия Экзамен
ПК-3 -Способен собирать,	З-1 - Изложить основы проектирования и элементы	Домашняя работа № 1 Домашняя работа № 2

обрабатывать и интерпретировать экспериментальные данные, необходимые для проектной и производственно-технологической деятельности, а также разрабатывать новые алгоритмические, методические и технологические решения в конкретной сфере профессиональной деятельности	архитектурных решений информационных систем П-1 - Подготовить техническое задание на разработку информационной системы У-1 - Интегрировать в практическую деятельность профессиональные стандарты в области информационных технологий	Лекции Практические/семинарские занятия Экзамен
ПК-4 -Способен к анализу требований и разработке вариантов реализации информационной системы, оценке качества, надежности и эффективности информационной системы в конкретной профессиональной сфере	З-1 - Объяснить методику анализа требований и вариантов реализации информационных систем П-1 - Имеет практический опыт разработки вариантов реализации информационных систем У-1 - Оценивать качество, надежность и эффективность информационной системы	Домашняя работа № 1 Домашняя работа № 2 Лекции Практические/семинарские занятия Экзамен
ПК-5 -Способен устанавливать и администрировать программные системы; реализовывать техническое сопровождение информационных систем; интегрировать информационные системы с используемыми аппаратно-программными комплексами	З-1 - Перечислить методики установки и администрирования программных систем П-1 - Имеет практический опыт разработки интеграции информационных систем с использованием аппаратно-программных комплексов У-1 - Реализовывать техническое сопровождение информационных систем	Домашняя работа № 1 Домашняя работа № 2 Лекции Практические/семинарские занятия Экзамен
ПК-6 -Способен применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных,	З-1 - Характеризовать методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии П-1 - Выполнять разработку программного обеспечения на	Домашняя работа № 1 Домашняя работа № 2 Лекции Практические/семинарские занятия Экзамен

операционные системы, электронные библиотеки и пакеты программ, сетевые технологии	современных языках программирования П-2 - Осуществлять обоснованный выбор передовых методов IT-области в профессиональной деятельности У-1 - Систематизировать и оценивать современные языки программирования с точки зрения профессиональной деятельности	

3. ПРОЦЕДУРЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ В РАМКАХ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ МОДУЛЯ В БАЛЬНО-РЕЙТИНГОВОЙ СИСТЕМЕ (ТЕХНОЛОГИЧЕСКАЯ КАРТА БРС)

3.1. Процедуры текущей и промежуточной аттестации по дисциплине

1. Лекции: коэффициент значимости совокупных результатов лекционных занятий – 0.50		
Текущая аттестация на лекциях	**Сроки – семестр, учебная неделя**	**Максимальная оценка в баллах**
Домашняя работа	17	*100*
Весовой коэффициент значимости результатов текущей аттестации по лекциям – 0.50		
Промежуточная аттестация по лекциям – экзамен **Весовой коэффициент значимости результатов промежуточной аттестации по лекциям – 0.50**		
2. Практические/семинарские занятия: коэффициент значимости совокупных результатов практических/семинарских занятий – 0.50		
Текущая аттестация на практических/семинарских занятиях	**Сроки – семестр, учебная неделя**	**Максимальная оценка в баллах**
Домашняя работа	17	100
Весовой коэффициент значимости результатов текущей аттестации по практическим/семинарским занятиям– 1.00		
Промежуточная аттестация по практическим/семинарским занятиям–нет **Весовой коэффициент значимости результатов промежуточной аттестации по практическим/семинарским занятиям– 0.00**		
3. Лабораторные занятия: коэффициент значимости совокупных результатов лабораторных занятий –не предусмотрено		
Текущая аттестация на лабораторных занятиях	**Сроки – семестр, учебная неделя**	**Максимальная оценка в баллах**

Весовой коэффициент значимости результатов текущей аттестации по лабораторным занятиям -не предусмотрено		
Промежуточная аттестация по лабораторным занятиям –нет **Весовой коэффициент значимости результатов промежуточной аттестации по лабораторным занятиям – не предусмотрено**		
4. Онлайн-занятия: коэффициент значимости совокупных результатов онлайн-занятий –не предусмотрено		
Текущая аттестация на онлайн-занятиях	**Сроки – семестр, учебная неделя**	**Максимальная оценка в баллах**
Весовой коэффициент значимости результатов текущей аттестации по онлайн-занятиям -не предусмотрено		
Промежуточная аттестация по онлайн-занятиям –нет **Весовой коэффициент значимости результатов промежуточной аттестации по онлайн-занятиям – не предусмотрено**		

3.2. Процедуры текущей и промежуточной аттестации курсовой работы/проекта

Текущая аттестация выполнения курсовой работы/проекта	**Сроки − семестр, учебная неделя**	**Максимальная оценка в баллах**
Весовой коэффициент текущей аттестации выполнения курсовой работы/проекта– не предусмотрено		
Весовой коэффициент промежуточной аттестации выполнения курсовой работы/проекта– защиты – не предусмотрено		

4. КРИТЕРИИ И УРОВНИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ МОДУЛЯ

4.1. В рамках БРС применяются утвержденные на кафедре/институте критерии (признаки) оценивания достижений студентов по дисциплине модуля (табл. 4) в рамках контрольно-оценочных мероприятий на соответствие указанным в табл.1 результатам обучения (индикаторам).

Таблица 4

Критерии оценивания учебных достижений обучающихся

Результаты обучения	**Критерии оценивания учебных достижений, обучающихся на соответствие результатам обучения/индикаторам**
Знания	Студент демонстрирует знания и понимание в области изучения на уровне указанных индикаторов и необходимые для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Умения	Студент может применять свои знания и понимание в контекстах, представленных в оценочных заданиях, демонстрирует освоение умений на уровне указанных индикаторов и необходимых для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Опыт /владение	Студент демонстрирует опыт в области изучения на уровне указанных индикаторов.

Другие результаты	Студент демонстрирует ответственность в освоении результатов обучения на уровне запланированных индикаторов. Студент способен выносить суждения, делать оценки и формулировать выводы в области изучения. Студент может сообщать преподавателю и коллегам своего уровня собственное понимание и умения в области изучения.

4.2 Для оценивания уровня выполнения критериев (уровня достижений обучающихся при проведении контрольно-оценочных мероприятий по дисциплине модуля) используется универсальная шкала (табл. 5).

Таблица 5

Шкала оценивания достижения результатов обучения (индикаторов) по уровням

Характеристика уровней достижения результатов обучения (индикаторов)				
№ п/п	**Содержание уровня выполнения критерия оценивания результатов обучения (выполненное оценочное задание)**	**Шкала оценивания**		
		Традиционная характеристика уровня		**Качественная характеристика уровня**
1.	Результаты обучения (индикаторы) достигнуты в полном объеме, замечаний нет	Отлично (80-100 баллов)	Зачтено	Высокий (В)
2.	Результаты обучения (индикаторы) в целом достигнуты, имеются замечания, которые не требуют обязательного устранения	Хорошо (60-79 баллов)		Средний (С)
3.	Результаты обучения (индикаторы) достигнуты не в полной мере, есть замечания	Удовлетворительно (40-59 баллов)		Пороговый (П)
4.	Освоение результатов обучения не соответствует индикаторам, имеются существенные ошибки и замечания, требуется доработка	Неудовлетворительно (менее 40 баллов)	Не зачтено	Недостаточный (Н)
5.	Результат обучения не достигнут, задание не выполнено	Недостаточно свидетельств для оценивания		Нет результата

5. СОДЕРЖАНИЕ КОНТРОЛЬНО-ОЦЕНОЧНЫХ МЕРОПРИЯТИЙ ПО ДИСЦИПЛИНЕ МОДУЛЯ

5.1. Описание аудиторных контрольно-оценочных мероприятий по дисциплине модуля

5.1.1. Лекции

Самостоятельное изучение теоретического материала по темам/разделам лекций в соответствии с содержанием дисциплины (п. 1.2. РПД)

5.1.2. Практические/семинарские занятия

Примерный перечень тем

1. content based DHT, consistent hashing, rarest1st BitTorrent
2. graph based BGP. DNS based SMTP/HTTP
3. crypto based git, bittorrent. Криптография
4. Blockchain BitCoin, etc

Примерные задания

Для заданного файла с данными, дерево хэшей определяется, как двоичное, с размером блока 1KB и хэш функцией SHA-256. Для простоты, размер файла данных кратен 1KB (нет незавершенного блока). Раскладка дерева в файле хэшей: in order, aka bin numbers.

Хэш всего дерева, он же хэш файла, определяется, как SHA256 хэш *файла пиков*. Файл пиков содержит 32 ячейки, где в ячейке i либо хэш пика на уровне i, либо нули, если пика такого пика нет. В статье Russ Cox термина "peak" нет, там говорится hashes of complete trees. В RFC7574 также используется термин munro. (Размер такого файла пиков 65x32 байт).

Подпись накладывается на хэш дерева по алгоритму Ed25519.

1. создать дерево хэшей для файла, по спеке, с bin раскладкой по RFC7574 `*.hashtree` Хэши сохраняем в hex, с переводом строки (65 байт на хэш, можно читать как текстом, так и по смещению).
2. положить пиковые хэши в отдельный файл `*.peaks`, в том же формате (размер файла пиков 65*32=2080 байт).
3. посчитать хэш файла, алгоритм SHA256, на bash Сохранить в файл `*.root` (65 байт).
4. создать доказательство целостности для блока данных `*.proof` Доказательство состоит из последовательности дядиных хэшей от 0 этажа вверх, в том же hex формате. Доказательство позволяет проверить блок данных об его пиковый хэш.
5. проверить доказательство целостности для блока `verify` Тут мы должны проверить, что пики соответствуют хэшу файла, а дядины хэши позволят проверить, что блок соответствует своему пику.
6. подписать файл `sign` (файлы ключей `.pub` `.sec`, подпись в файле `*.sign`)
7. проверить подпись `check`.

Результаты складывать в файлы с соответствующим расширением, например `datafile.hashtree`, `datafile.peaks`, `datafile.sign`. Исходные данные брать так же, из файлов. При вызове программы первый аргумент всегда - название исходного файла. Пример:

```
$ python peaks-ivanov.py pikachu.mov
reading pikachu.mov.hashtree...
putting the peaks into pikachu.mov.peaks...
all done!
$ bash root-petrova.sh pikachu.mov
hashing pikachu.mov.peaks
all done!
$ clang++ proof-sidorov.cpp -lsodium -o proof-sidorov
$ ./proof-sidorov pikachu.mov 18
reading pikachu.mov.hashtree...
putting the proof into pikachu.mov.18.proof...
all done!
```

`verify`, `check` передают результат кодом возврата (0/не 0) и пишут на экран.

Задание: создать git-подобную "файловую систему". Бэк-ендом будет обычная файловая система. Все блобы хранятся как файлы, их SHA-256 хэш это их имя. Есть два типа файлов: блобы и директории. Блоб - это просто файл, директорий - отсортированный список вида

```
file.txt: 68e9ea8ccf107dd46bdff3ce133a0764c3c7420bc8405d2c5527f127e84ced60
dir1/     84786d4331818e6be54f105f48aa7e6d7c13e0aa504c1d7ac26a3b9dc7edb4e1
```

Разделители `:\t` для файлов и `/\t` для директориев, перевод строки `\n`, включая после последней строки. Итого, формат без вольностей. Имя файла - UTF8 от пробела и выше, исключая разделители `UTF8 - [ \t:/]`.

Требуется реализовать 6 команд: put, get, ls, rm, mkdir, check. Аргументы каждой команды:

```
./put-ivanov subdir/README 84786d4331818e6be54f105f48aa7e6d7c13e0aa504c1d7ac26a3b9dc7edb4e1 < README
./get-petrov dir/subdir/README 834af9243b300ed2fb9235e74e158c0bcdfd34ef9590ba93b56c70e9f970040f
./ls-sidorov dir/ 834af9243b300ed2fb9235e74e158c0bcdfd34ef9590ba93b56c70e9f970040f
./rm-boshirov dir 834af9243b300ed2fb9235e74e158c0bcdfd34ef9590ba93b56c70e9f970040f
./mkdir-smith dir/newsub 834af9243b300ed2fb9235e74e158c0bcdfd34ef9590ba93b56c70e9f970040f
./check-averbuch dir/ 834af9243b300ed2fb9235e74e158c0bcdfd34ef9590ba93b56c70e9f970040f
```

Итого, 2 аргумента: путь и корневой хэш. Каждая пишущая команда *выводит корневой хэш новой версии* файловой системы, а читающая выводит данные (get содержимое файла, ls рекурсивно дерево файлов). Разные реализации одной команды должны выдавать побитно идентичный результат. Если результат отличается - кто-то точно неправ. Команда check параноит дерево файлов на предмет любых нарушений спецификации либо отсутствующих данных, результат выводит в свободной форме.

Разрешённые языки:

- C
- C++
- go
- bash
- node.js
- Zig
- Rust
- OCaml
- Python
- Java

LMS-платформа

1. https://github.com/decentralized-hse

5.2. Описание внеаудиторных контрольно-оценочных мероприятий и средств текущего контроля по дисциплине модуля

Разноуровневое (дифференцированное) обучение.

Базовый

5.2.1. Домашняя работа № 1

Примерный перечень тем

1. Fuzz

Примерные задания

Задание: взять любое решение прошлого семестра из практики Форматы сериализации и нафаззить до 3х багов с нарушением round-trip guarantee. То есть, найти валидные C структуры, которые код преобразует в другой формат и обратно так, что результат не соответствует оригиналу. Если получится краш - ещё лучше. Для багов сделать фикс и PR.

Для golang рекомендую использовать встроенный фаззер go-fuzz. Для C и C++ рекомендую libfuzzer. С остальными сам не работал, ничего не рекомендую. Для Java, Rust и других популярных языков популярные фаззеры тоже есть.

LMS-платформа

1. https://github.com/decentralized-hse

5.2.2. Домашняя работа № 2

Примерный перечень тем

1. Коллаборация

Примерные задания

Задача: написать преобразование из "квадратного" C-формата в ваш и обратно, *с побитной точностью* (round-trip guarantee). Ваш:

1. protobuf (protobuf)
2. JSON (json)
3. Cap'n proto (capnproto)
4. Flat buffers (flat)
5. Key: value (kv)
6. sqlite (sqlite)
7. XML (xml)
8. sstable (sstable)

Оригинальный C формат (bin), little endian:

```
struct Student {
    // имя может быть и короче 32 байт, тогда в хвосте 000
    // имя - валидный UTF-8
    char    name[32];
    // ASCII [\w]+
    char    login[16];
    char    group[8];
    // 0/1, фактически bool
    uint8_t practice[8];
    struct {
        // URL
        char    repo[59];
        uint8_t mark;
    } project;
    // 32 bit IEEE 754 float
    float   mark;
}
```

LMS-платформа

1. https://github.com/decentralized-hse

5.3. Описание контрольно-оценочных мероприятий промежуточного контроля по дисциплине модуля

5.3.1. Экзамен

Список примерных вопросов

1. Экзамен заключается в работе над проектом. Проект делается группой в 2-3 человека и должен затрагивать темы курса: криптографию, федерирование и маршрутизацию, метрики и инцентивизацию, коллаборацию и консенсус. Проект делается на любой платформе - под Web, мобильные устройства или десктоп, на любом удобном языке. Проект можно сдавать в разной степени готовности от работающего эскиза/PoC до полного dogfooding, когда приложение уже используется. Степень готовности проекта будет влиять на итоговую оценку. Студенты сами формулируют цели, которые решает проект (например, цель разработки веб-сайта может быть такой: посетители сайта будут уверены, что существует угроза глобального потепления). Оценивание происходит исходя из особенности проекта, технологии и способы реализации

LMS-платформа − не предусмотрена

5.4 Содержание контрольно-оценочных мероприятий по направлениям воспитательной деятельности

Направление воспитательной деятельности	Вид воспитательной деятельности	Технология воспитательной деятельности	Компетенция	Результаты обучения	**Контрольно-оценочные мероприятия**

Профессиональное воспитание	профориентационная деятельность	Технология формирования уверенности и готовности к самостоятельной успешной профессиональной деятельности	ПК-4	У-1	Домашняя работа № 1 Домашняя работа № 2 Лекции Практические/семинарские занятия Экзамен