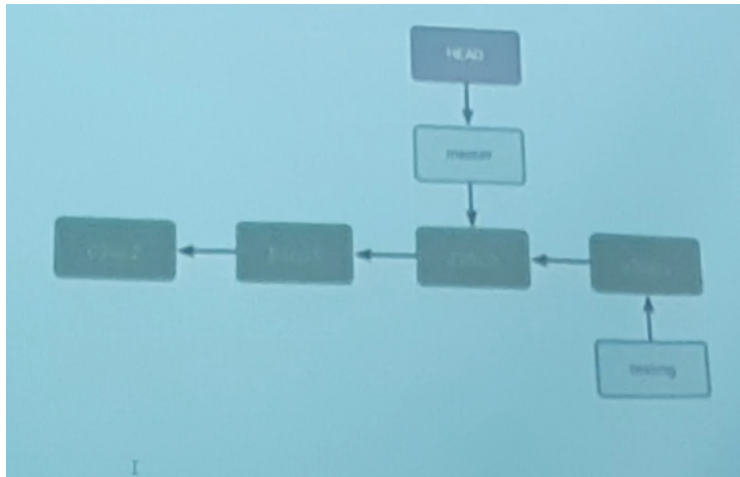


Системы контроля версий

- 1)
- 2)
- 3)
- 4) Система версий SVC
Распределенная система контроля версий
- 5) Что такое ветка branch?
Это параллельная версия репозитория
- 6) К какому типу систем контроля версий относится git
Распределенная
- 7) Минусы локальной системы контроля версий
Большая вероятность потерять все если не использовать бэкапы
Неудобный механизм одновременной работы нескольких пользователей с системой, хранилище просто блокируется пока заблокировавший его пользователь его не разблокирует
Работа только с одним файлом, каждый файл должен контролироваться отдельно
- 8) Особенности централизованной системы контроля версий:
Есть вероятность потери всей истории проекта если нет резервной копии
С репозиторием могут работать несколько человек параллельно
Центральный сервер является уязвимой точкой системы. Его отключение...
- 9) Недостатки системы современных версий (CVS)
Для каждого изменения бинарного файла сохраняется вся версия файла, а не только внесенное изменение
При перемещении, переименовании файла или директории теряются все привязанные к этому файлу или директории изменения
- 10) Достоинства системы совместных версий (CVS)
Позволяет управлять не одним файлом а целыми проектами
Несколько клиентов могут одновременно работать над одним и тем же проектом
Широко распространена и поставляется по умолчанию с большинством операционных систем Linux
- 11) Централизованная система управления версиями создана в 2000 году и основана на технологии клиент-сервер:
Subversion
- 12) Достоинства системы управления версиями Git
Все вышеперечисленное
(гибкая система проектов и слияния веток между собой)
(множество графических оболочек...)
(универсальный сетевой доступ с использованием протоколов http, ftp ...)
- 13) Что такое github
Веб Сервис для хостинга IT проектов и их совместной разработки основанный на git
- 14) Место где СКВ хранит свои метаданные и базу данных объектов проекта
Репозиторий

- 15) Установите в правильном порядке алгоритм для коммита изменений
 - 1) внесение изменений
 - 2) подготовка файлов
 - 3) фиксация изменений
- 16) Как сделать коммит
Сделать изменения, собрать эти изменения командой `git add` или `git commit -a` и указать коммит сообщение после ключа `-m`
- 17) Чем отличается `master` и `origin master`
`master` принадлежит локальному репозиторию, а `origin master` удаленному
- 18) Что делает `git show`
Показывает изменения, сделанные в указанном коммите
- 19) Введите название команды для создания нового репозитория
`git init`
- 20) Введите название команды для индексации изменения только в файле `name_file`
`git add name_file`
- 21) Что делает команда `git add`
Начинает отслеживать указанный файл или файлы
- 22) Введите название команды для перемещения текущих изменений в стэк
`git stash`
- 23) Введите команду которая отправляет запрос на изменение кода в чужом репозитории
`pull request`
- 24) Какой командой можно произвести слияние веток
`git merge branch1 branch2`
- 25) Какой командой можно создать ветку `branch1`. Требуется прописать команду полностью
`git branch branch1`
- 26) Введите название команды которая служит для выгрузки изменений в удаленный репозиторий
`git push`
- 27) В каких состояниях может находиться файл в локальном репозитории
`modified`
`untracked`
`unmodified`
`staged`

28) Чем является head изображенная на рисунке



Указатель который указывает на текущую ветку

29) Что такое удаленный репозиторий

репозиторий находящийся на удаленном сервере

30) В каких состояниях могут находиться файлы в git

Измененном

Зафиксированном

Подготовленном

31) Что такое рабочий каталог

Это извлеченная из базы копия определенной версии проекта

32) Что такое каталог Gita

Это место где git хранит метаданные и базу данных вашего проекта

33) Что такое область подготовленных файлов

Это обычный файл обычно хранящийся в каталоге gita который содержит информацию о том что должно войти в следующий коммит

34) Сопоставьте систему контроля версий с описанием

Vazaar – система разработка которой поддерживается компанией Canonical. Система в основном написана на языке Python...

Mercurial – распределенная система написанная на языке python с некоторыми расширениями на C

git – распределенная система контроля версий, разработана Линусом Торвальдсом

codeville – Написанная на python распределенная система инновационный алгоритм объединения изменений (merge). Система исп при разработке оригинального BitTorrent

Darcs – распределенная система контроля версий написанная на Haskell используемая проектом Buildbot

Monotone – система написанная на C++ и исп SQLite как хранилище ревизий

35) Введите название команды для извлечения текущих изменений из стека

git stash pop

1)

2)

3)

- 4)
- 5) Вставьте недостающую команду
git clone <https://github.com/rtu-mirea/trpp-second-1>
git branch new_branch
*git checkout new_branch
git add
git commit -m "new commit in new_branch"...
- 6) Вставьте недостающую команду
git add new_file
git stash
git checkout master
*git stash pop
...
- 7) Вставьте недостающую команду
git clone <https://github.com/rtu-mirea/trpp-second-1>
*git branch new_branch
git checkout new_branch
git add -A
...
- 8) Вставьте недостающую команду
git add new_file
*git stash
...

Bash

- 1) Укажите какую строку нужно написать в файле чтобы он считался bash скриптом
#!/bin/bash
- 2) Каким способом можно записать в переменную значение команды в pwd
mydir = \$(pwd)
mydir = `pwd`
- 3) Какая команда возвращает истинное значение если n1 равно n2
b1 -eq b2
- 4) Сопоставьте команды файлов с описанием
-r file – проверяет существует ли файл и доступен ли он для чтения
-w file – проверяет существует ли файл и доступен ли он для записи
-e file – проверяет существует ли файл
-d file – проверяет существует ли файл и является ли он дерикторией
-f file – проверяет существует ли файл и является ли он файлом
- 5) Какие существуют циклы в bash скриптах
while
for
- 6) Требуется установить в качестве разделителя строки (\n). Напишите команду
IFS=\$'\n'
- 7) Каким образом можно вывести значение переменной person на экран
echo = "\$person is a good boy"

- 8) Как попасть в домашнюю директорию своего пользователя
cd \$HOME
- 9) Что выведет echo \$var1
- ```
#!/bin/bash
var1=$((5+5))
echo $var1
var2=$((var1 * 2))
echo $var2
var3=$((var2 / 4))
echo $var3
...
echo $var5
10
```

**Вопрос 9**  
Пока нет ответа  
Балл: 1,00  
Отметить вопрос

Что делает команда git show?

- ☐ Показывает содержимое файла
- ☐ Показывает состояние проекта
- ☒ Показывает изменения, сделанные в указанном коммите
- ☐ Показывает время

←

Предыдущая страница

**Вопрос 17**  
Пока нет ответа  
Балл: 1,00  
Отметить вопрос

Что такое ветка (Branch)?

- ☐ это рабочая (локальная) копия документов
- ☐ место, где СКВ хранит свои метаданные и базу данных объектов проекта
- ☒ это параллельная версия репозитория
- ☐ репозиторий, расположенный на локальном компьютере разработчика в каталоге

←

Предыдущая страница

Вопрос **19**

Пока нет ответа

Балл: 1,00

🚩 Отметить  
вопрос

Что делает команда git status?

- ☐ a. Показывает место, занимаемое репозиторием на жестком диске и кол-во выделенного под репозиторий месте
- ☐ b. Показывает имя и email нашего пользователя, а также является ли он авторизованным в системе GitHub или нет
- ☒ c. Показывает состояние проекта: кол-во untracked, deleted, new и прочих файлов, количество коммитов, на которое отличается локальная версия репозитория от удаленного и так далее
- ☐ d. Такой команды нет, есть только команда git show

[Предыдущая страница](#)

[Следующая страница](#)

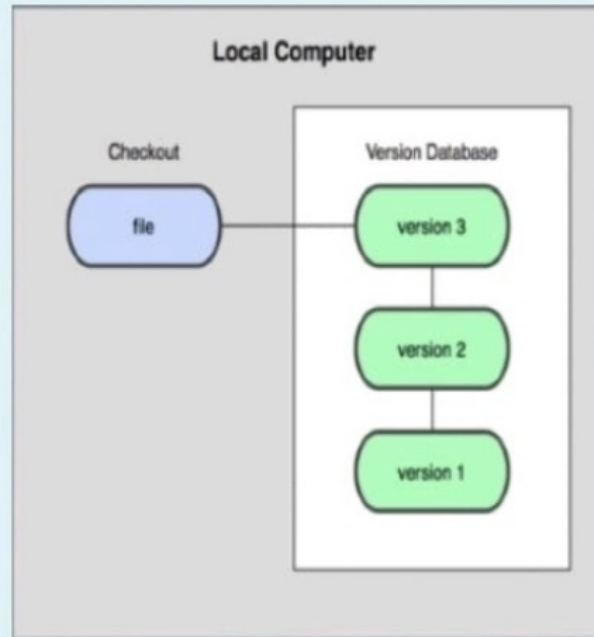
Вопрос 20

Пока нет ответа

Балл: 1,00

Отметить  
вопрос

Какая система управления версиями изображена на картинке?



- ☐ Централизованная система управления версиями
- ☐ Распределенная система контроля версий
- ☐ Система совместных версий SVC
- ☒ Локальная система управления версиями

Вопрос 21

Пока нет ответа

Балл: 1,00

Отметить  
вопрос

Что выведет \$HOME?

Ответ: ДОМАШНЮЮ ДИРЕКТОРИЮ

Предыдущая страница

Следующая страница

Моя страница

Поиск

1 сообщение

Ответить

Новости

Вопрос 22

Пока нет ответа

Балл: 1,00

Отметить вопрос

Что такое репозиторий Git?

☐ a. Любая папка, находящаяся внутри Git

☒ b. Репозиторий Git представляет собой каталог файловой системы, в котором находятся файлы конфигурации репозитория, файлы журналов, хранящие операции, выполняемые над репозиторием, индекс, описывающий расположение файлов, и хранилище, содержащее собственно файлы

☐ c. Папка .git/ и все входящие в нее

☐ d. Любая директория/папка в моей ОС

Предыдущая страница

Следующая страница

Вопрос 23

Пока нет ответа

Балл: 1,00

Отметить вопрос

Какие каталоги существуют по умолчанию:

☒ src/main/java

☒ src/test/java

☒ src/main/resources

☐ src/test/resources

Предыдущая страница

Вопрос 24

Пока нет ответа

Балл: 1,00

Отметить вопрос

Как скачать ветку their\_branch, если она уже есть в удаленном (remote) репозитории, но нет локально?

☐ a. Команда: git get origin their\_branch

☐ b. Команда: git clone origin their\_branch

☒ c. Команда: git clone their\_branch

☐ d. Команда: git fetch origin their\_branch



Вопрос 25

Пока нет ответа

Балл: 1,00

Отметить  
вопрос



Что выведет `echo $var2`?

```
#!/bin/bash
var1=$((5 + 5))
echo $var1
var2=$(($var1 * 2))
echo $var2
var3=$(($var2 / 4))
echo $var3
var4=$(($var3 % 5))
echo $var4
var5=$(($var4 ** 2))
echo $var5
```

Ответ:

1

тут, видимо, ответ 20

Вопрос 18

Пока нет ответа

Балл: 1,00

Отметить  
вопрос



Введите команду, которая очищает все сборочные директории в gradle

Ответ:

stash pop

Предыдущая страница

Вопрос 12

Пока нет ответа

Балл: 1,00

Отметить  
вопрос



Как проиндексировать несколько файлов одной командой?

- ☐ a. `git add TEXT1.txt ADD TEXT2.txt ADD TEXT3.txt`
- ☐ b. `git add TEXT1.txt, TEXT2.txt, TEXT3.txt`
- ☒ c. `git add TEXT1.txt TEXT2.txt TEXT3.txt`



Предыдущая страница

Следующая страница

To

Вопрос **14**

Пока нет ответа

Балл: 1,00

Отметить  
вопрос



Какая команда возвращает истинное значение, если n1 равно n2:

- ☐ n1 -ne n2
- ☐ n1 -gt n2
- ☐ n1 -le n2
- ☐ n1 -ge n2
- ☒ n1 -eq n2

Вопрос **15**

Пока нет ответа

Балл: 1,00

Отметить  
вопрос

Maven - это

- ☒ система автоматической сборки проектов на основе описания их структуры в файлах на языке POM
- ☐ фреймворк на JVM для построения легковесных модульных приложений
- ☐ Java библиотека, призванная освободить разработчика от написания шаблонного кода
- ☐ система автоматической сборки, построенная на принципах Apache Ant и Apache Maven

Предыдущая страница

Следующая страница

1 из 116

Поделиться · Сохранить к себе · Ещё ▾

Вопрос **1**

Пока нет ответа

Балл: 1,00

Отметить  
вопрос



Виды автоматизации в системе сборки, применяемые в различных инструментах:

- ☒ автоматизация по запросу (on-demand automation): запуск пользователем сценария в командной строке
- ☐ запланированная автоматизация (scheduled automation): непрерывная интеграция, происходящая в виде ночных сборок
- ☒ условная автоматизация (triggered automation): непрерывная интеграция, выполняющая сборку при каждом подтверждении изменения кода (commit) в системе управления версиями

Следующая страница

Вопрос 11

Пока нет ответа

Балл: 1,00

Отметить вопрос

Введите команду, которая собирает приложение целиком в gradle

Ответ:

Предыдущая страница

Следующая

Вопрос 10

Пока нет ответа

Балл: 1,00

Отметить вопрос

Как проиндексировать файлы и сделать коммит одной командой?

☐ a. git commit --add -m "Comment"

☒ b. git commit -add -m "Comment"

☐ c. git commit -a -m "Comment"

☐ d. git commit-add -m "Comment"

☐ e. git commit -m "Comment"

Предыдущая страница

Система контроля (управления) версий - обеспечение для облегчения работы с меняющейся информацией

Скв включает задачи:

- Объединение наработок и экспериментных изменений, ведущихся несколькими разработками
- Сравнение и поиск ранее зафиксированных изменений исходного кода
- Хранение истории зафиксированных изменений исходного кода
- Фиксация изменений и отслеживание для файлов, полученных в проекте

Определить скв по картинке:

Локальная система контроля версий

Централизованная система контроля версий

## Децентрализованная система контроля версий

branch - параллельная версия репозитория  
Ветвь (ветка, branch) – механизм, который служит для ветвления дерева ревизий файла

Гит - самый популярный и распределенный

### Локальная

- Большая вероятность потерять всё
- Блокируется одним пока не закончит работу
- Работа с одним файлом, каждый контролируется отдельно
- Неудобный механизм одновременной работы нескольких пользователей

### Централизованная скв

- Вероятность потери всей истории при отсутствии резервной копии
- С репозитории работают несколько человек параллельно
- Центральный сервер - уязвимая сторона системы, остановка приведет к остановке взаимодействия

### Система совместных версий cvs

#### Недостатки

- Для каждой версии сохраненного файла сохраняется вся версия файла, а не изменения
- При перемещении или переименовании файла или директории теряются все изменения

#### Достоинства

- Широко распространена и поставляется по-умолчанию с большинство Linux
- Несколько клиентов могут работать над одним проектом
- Позволяет управлять не одним файлом, а целым проектом

Централизованная система созданная в 2000 году и осн на технологии клиент-сервер - Subversion

### Достоинства упр. версий Гит

- Гибкая система упр. Версий и слияния веток
- Множество графических оболочек
- Обеспечивает сетевой доступ http ftp ssh scp и др
- Всё вышеперечисленное - правильный ответ

Гитхаб - веб-сервис для хостинга ит проектов и совместной разработки осн на гит

Место где система контроля версий хранит метаданные и базы данных, объекты проекта - репозиторий

О

Алгоритм, чтоб систем контроль версий сработал правильно (для коммитных untracked изменений):

1. Внесение изменений
2. Подготовка файлов
3. Фиксация изменений

Как сделать коммит: сделать изменения, собрать изменения `git add` или `git commit -a` и указать коммит сообщения `-m`

Отличия `master` и `origin/master` - мастер локальный, `origin` удаленный

`Git show` показывает все изменения в указанном коммите

Команда для инициализации нового локального репозитория `git init`

Команда для индексации изменений в файле `*название файла*`  
`git add *название файла*`

Что делает команда `git add`? Начинает отслеживать указанный файл или файлы

Название команды для сохранения текущих изменений в стек `git stash`

Команда для отправки запроса на изменение кода в чужом репозитории `git pull request`

Команда для слияния веток `git merge branch1 branch2`

Команда создания ветки `git branch branch1`

Название команды для выгрузки изменений в удаленный репо: `git push`

Состояния файла в локальном репо: `git stash`  
`Modified`, `untracked`, `unmodified`, `staged`

Чем является `Head` на рисунке

Указатель на текущую ветку

Что такое удаленный репозиторий? Репозиторий на удаленном сервере

Состояния файлов в гит:  
Измененное, зафиксированное, подготовленное

Рабочий каталог - извлеченная из базы копия определенной версии проекта

Каталог гита: место где гит хранит метаданные и базы данных объектов вашего проекта

Область подготовленных файлов- обычный файл, хранящийся в каталоге гита, хранящий инфу о том, что пойдет в commit

Сопоставьте систему версий с описаний

ÿ Система, разработка которой проводится компанией кононикал и написана на питоне и исп в частности в sql - Bazaar

ÿ На питоне с расширениями на си, относится скв Mercurial

ÿ Разработана Линусом Торвальдсом - Git

ÿ Система на питон, распределенная и тд - Codeville

ÿ Распределенная система на хаски и в бинбут - Darcs

ÿ Система на си++, исп sql как хранилище версий Monotone

Команда для извлечения изменений из стека- git stash drop

Строка в файле чтоб считался баш скриптом

```
#!/bin/bash
```

Каким скриптом сохранить значение команды pwd - Print Working Directory

```
mydir=$(pwd)
```

```
mydir = 'pwd'
```

Какая команда возвращает истинное значение если n1 = n2

```
n1 -eq n2
```

Сопоставить команды проверки файлов с описанием:

-r file - Проверяет существование и доступность для чтения файла

-w file - Проверяет существование и доступность для записи файла

-e file - Проверка существования файла

-d file - Проверяет существование и является ли директорией файл

-f file - Проверяет существование и то что является файлом

Циклы в баш скриптах

```
while
```

```
for
```

Перенос строки в качестве разделителя

```
IFS = '$\n'
```

Способ вывода значения переменной person на экран

```
echo "$person is a good boy"
```

Перейти в домашнюю директорию своего пользователя

```
cd
```

```
cd~
```

```
cd $HOME
```

Что выведется в каждом из случаев

```
echo var1 – 10
echo var2 – 20
echo var3 – 5
echo var4 – 0
echo var5 – 0
```

Сопоставить

\$# - количество возможных вариантов

\$0 - имя скрипта

\$5 - введенный параметр

\$? - вернувшееся значение из функции

Подключение библиотеки testlib к скрипту

.testlib

В какую переменную запишется значение команды read без указания переменной

В переменную \$REPLY

С помощью какого символа производится

запись конец файла

>>

Команда очистки всех сборочных директорий gradle

clean

Команда сборки приложения целиком в gradle

Чем отличается master и origin master

Команда создания документации gradle

javadoc

Раздел зависимостей проекта gradle

dependencies

Раздел gradle для указания версии явы для работы

java

checkstyle - инструмент с открытым кодом для проверки кода на соответствие стандартам

marven система сборки на основе описания их структуры в файлах на языке xml

Виды автоматизации в системе сборки в различных инструментах

• Условная автоматизация (triggered automation) - выполняет сборки при каждом подтверждении изменения кода commit

• Запланированная автоматизация (scheduled automation) - интеграция происходящая как правило в ночных сборах

ÿ Автоматизация по запросу (on-demand automation)- запуск пользователем сценариев в командной строке

Какой командой можно определить состояние контейнеров docker  
docker ps -a

Команда просмотра всех имеющихся образов  
docker images

Команда для загрузки питона 3.7.2  
docker pull python:3.7.2

Команда чтобы попасть в контейнер testserver и запустить баш  
docker exec -it testserver bash

Ключ назначения переменных окружения внутри контейнера  
-e

Dockerfile - сценарий, который состоит из последовательности команд и аргументов, необходим для создания образа  
Инструкция Dockerfile для открытия порта  
EXPOSE

Инструкция Dockerfile задающая базовый образ  
FROM

Инструкция Dockerfile копирующая папки и файлы в контейнер  
ADD

Контейнер Docker - автономный исполняемый пакет программного обеспечения со всем необходимым для запуска приложения

Образ докер - неизменные шаблоны для создания одинаковых контейнеров