

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

Математический факультет
Кафедра функционального анализа и операторных уравнений

**Программная генерация задач ЕГЭ по математике
(раздел «Алгебра и начала анализа») на языке JavaScript**

Бакалаврская работа

Направление 02.03.01 Математика и компьютерные науки

Профиль Математическое и компьютерное моделирование

Зав. кафедрой Каменский д. ф.-м. н., проф. М.И. Каменский 05.06.2024г.

Обучающийся Алендарь С.Д. Алендарь

Руководитель Каменский д. ф.-м. н., проф. М.И. Каменский

Содержание

Введение

ЕГЭ (Единый государственный экзамен) – это государственный экзамен, проводимый в России для выпускников школ. ЕГЭ проводится по 14 предметам, включая французский, испанский, немецкий и с 2019 года китайский язык. Обязательными являются русский язык и базовый или профильный модуль по математике.

ЕГЭ проходит каждый год в период с мая по июль. Цель экзамена – определить уровень знаний и умений выпускников и принять решение о поступлении в высшие учебные заведения. Результаты ЕГЭ засчитываются как вступительное испытание в вузы России. Сдают ЕГЭ выпускники средних школ, которые получили документ об окончании общеобразовательной школы. В целом, это молодые люди в возрасте 16-18 лет.

Для успешной сдачи ЕГЭ необходима тщательная подготовка, и немаловажную роль в этом играет сайт «Решу ЕГЭ»[4]. Он может быть полезен в подготовке к экзамену по математике. Однако за время подготовки к экзаменам школьники быстро сталкиваются с дефицитом заданий, а в некоторых случаях прибегают к списыванию.

«Час ЕГЭ» — компьютерный образовательный проект, разрабатываемый при математическом факультете ВГУ в рамках «OpenSource кластера» и предназначенный для помощи учащимся старших классов подготовиться к тестовой части единого государственного экзамена.

Задания в «Час ЕГЭ» генерируются случайным образом по специализированным алгоритмам, называемых шаблонами, каждый из которых охватывает множество вариантов соответствующей ему задачи. В настоящее время в проекте полностью реализованы тесты по математике с кратким ответом. Планируется с течением времени включить в проект тесты по другим предметам

школьной программы.

Цель - разработка шаблонов, основанных на задачах из ЕГЭ [5]. Большое и разнообразное количество заданий, генерируемых алгоритмами шаблонов, позволяет объективно оценить знания школьников. Благодаря им решаются проблемы списывания и нехватки заданий.

1. Виртуальные машины VirtualBox

1.1. Установка VirtualBox и Ubuntu

Для разработки и запуска шаблонов необходима операционная система Linux. Без неё не возможна работа с "ЧАС ЕГЭ".

Вместо установки полноценной операционной системы на компьютер, мы выбрали более удобный вариант. Мы предпочли скачать программу VirtualBox и поставить на неё Linux. Далее идёт описание процесса установки и запуска Виртуальной машины.

1. Перейходим по ссылке <https://www.virtualbox.org>. Нажимаем кнопку «Download», выбираем «Windows hosts» и устанавливаем VirtualBox.

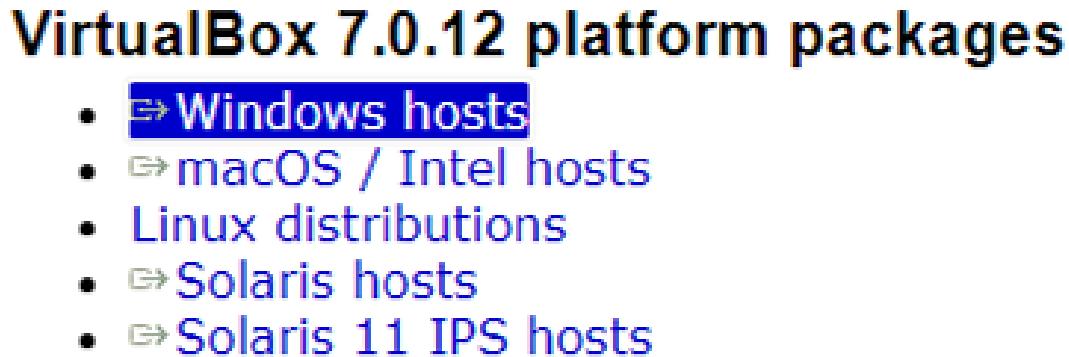


Рис. 1: «Windows hosts».

2. Далее, для того чтобы любая виртуальная машина могла работать, необходимо включить виртуализацию. В данном случае, мы имеем дело с Windows 11. Чтобы проверить включена ли на ней виртуализация, открываем диспетчер задач и переходим в раздел «Производительность». Диспетчер задач можно открыть, нажав по панели задач правой кнопкой мыши и выбрав соответствующий пункт. Или можно зажать клавиши Ctrl+Alt+Del, и в открывшемся

меню выбрать диспетчер задач.

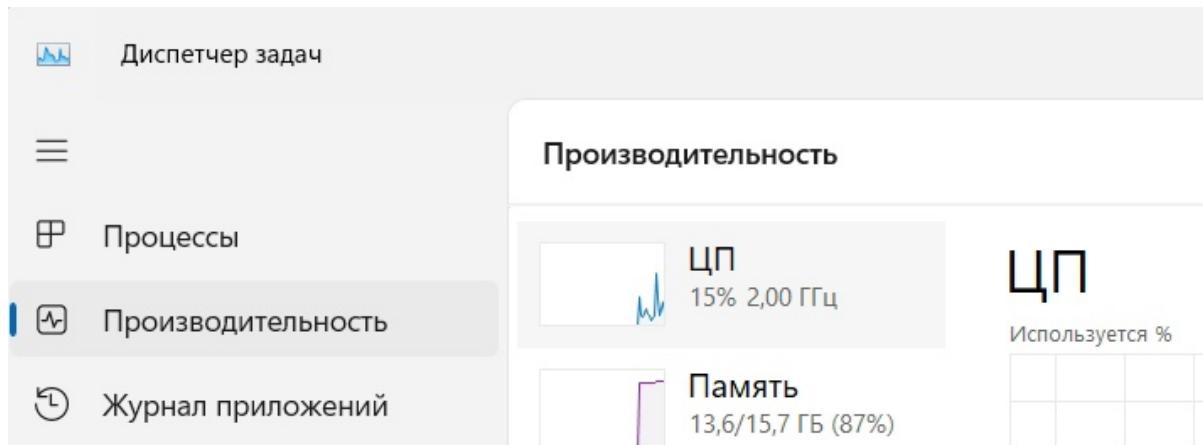


Рис. 2: Диспетчер задач. Производительность.

Открываем окно диспетчера задач на весь экран и внизу увидим, включена ли виртуализация.

Логических процессоров:	2
Виртуализация:	Включено
Кэш L1:	256 КБ
Кэш L2:	2,0 МБ

Рис. 3: Виртуализация.

Если виртуализация выключена, то необходимо войти в BIOS. Для этого нужно нажать на кнопку «Перезагрузить компьютер», и, как только он начинает запускаться, зажать клавишу «Esc» или начать нажимать много раз на клавишу delete, обычно именно эта клавиша используется на стационарных компьютерах для запуска BIOS, пока не запустится «Startup menu». На ноутбуках чаще всего для входа в BIOS используется клавиша F2 (иногда в сочетании с

Fn), но встречаются и другие клавиши или их комбинации, зависящие от производителя (Ctrl + Alt + Ins, Ctrl + Alt + F3, Ctrl + Alt + S и др.). Чтобы открыть BIOS надо нажать клавишу «F10» и перейти в «System Configuration», нажав два раза на кнопку со стрелкой вправо. Затем, с помощью стрелки вниз, перейти к «Virtualization Technology» и нажать клавишу «Enter». Далее нужно выбрать «Enable» и снова нажать «Enter». Затем снова нажать «F10» и выбрать «Yes», после чего компьютер перезагрузится уже с включённой виртуализацией.

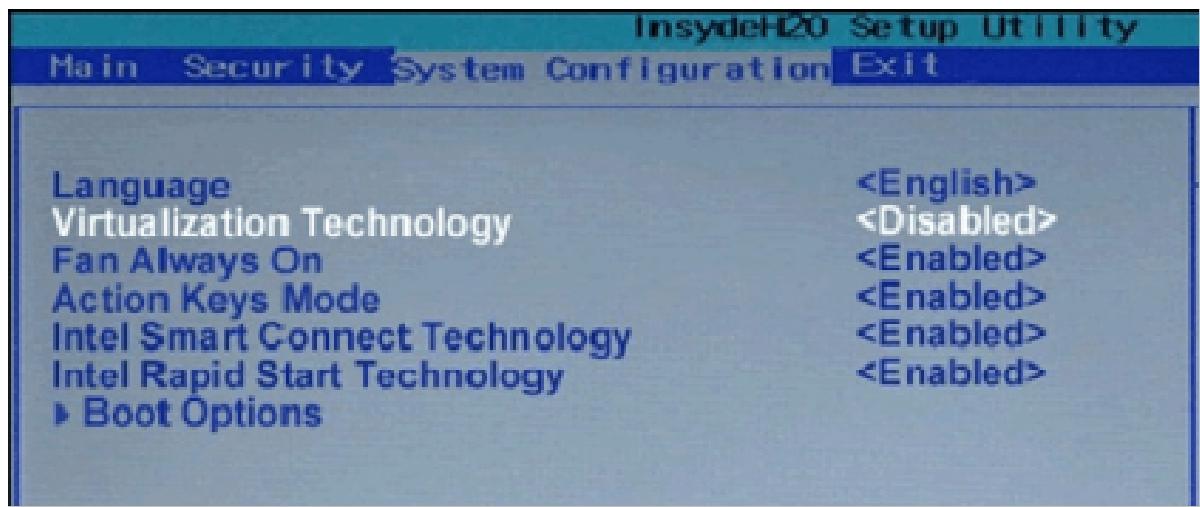


Рис. 4: BIOS. «Virtualization Technology».

3. Скачиваем Ubuntu с официального сайта по ссылке: <https://ubuntu.com/download/desktop>. В нашем случае, мы воспользовались версией Ubuntu 22.04.3. Затем открываем VirtualBox и нажимаем кнопку «Создать». Заполняя все поля, необходимо потянуть ползунок вправо, чтобы увеличить объем оперативной памяти, который будет использоваться виртуальной машиной. Для работы с Ubuntu минимальный размер объёма памяти должен быть не менее 2 гигабайт, иначе она не запустится. В остальном можно принять все установки по умолчанию. После создания, необходимо нажать на кнопку «Настройки», далее «Носители», у надписи «Контроллер: IDE» нажать на значок диска с плюсом.

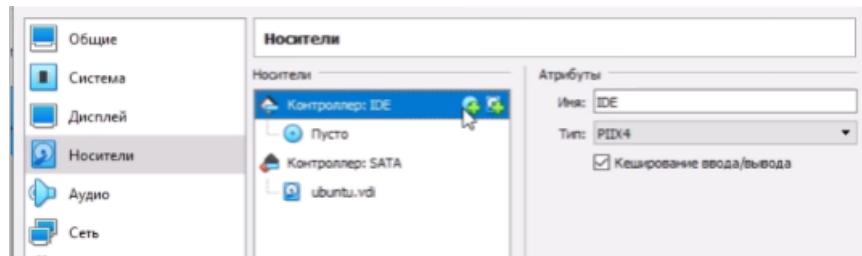


Рис. 5: Настройки. Носители.

Затем «Добавить» и выбрать скачанный ранее файл с Ubuntu. Далее нажимаем кнопку «Запустить». После запуска, выбираем язык, нажимаем «установить Ubuntu», заполняем все поля и выбираем параметры.

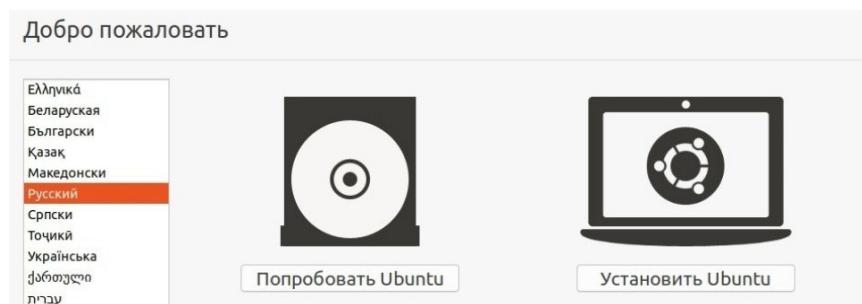


Рис. 6: Ubuntu.

И наконец видим интерфейс Linux.

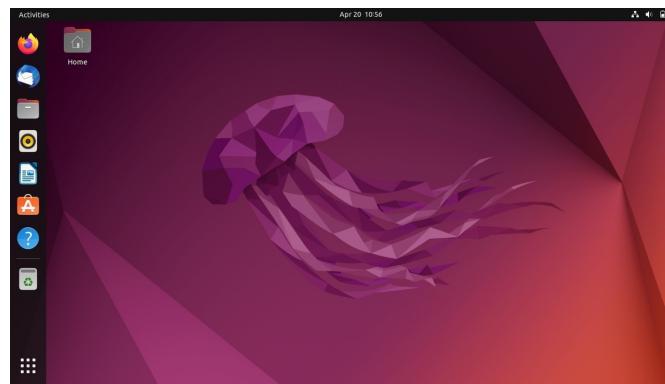


Рис. 7: Linux.

1.2. Снимки виртуальной машины

Обновлением виртуальной машины, или неудачная попытка настройки или просто какая-то ошибка при её запуске, может привести к потери ценных данных, а на переустановку системы и настройку программ может уйти очень много времени. Именно для таких случаев и предназначены снимки в VirtualBox. Они нужны для сохранения прогресса на случай непредвиденной поломки машины и отката до ранее стабильного состояния. Поэтому перед и после обновления или установки чего-то важного, стоит делать снимки.

Для создания снимка своей виртуальной машины нужно нажавть на кнопку с изображением фотоаппарата с плюсом.

Снимок можно удалить, восстановить и посмотреть на его свойства на момент его создания.

Имя	Дата создания
Снимок 1	10.02.2024 17:29
Снимок 2	16.02.2024 12:23
Снимок 3	25.02.2024 15:01
Снимок 4	16.03.2024 18:57
Снимок 5	31.03.2024 13:20
Снимок 5 (с работающей машиной) Сделан 31.03.2024 13:20	09.05.2024 17:14

Рис. 8: Дерево снимков.

При создании можно оставить название по умолчанию и не вводить никакого описания, но если таких снимков в будущем накопится много, можно будет запутаться.

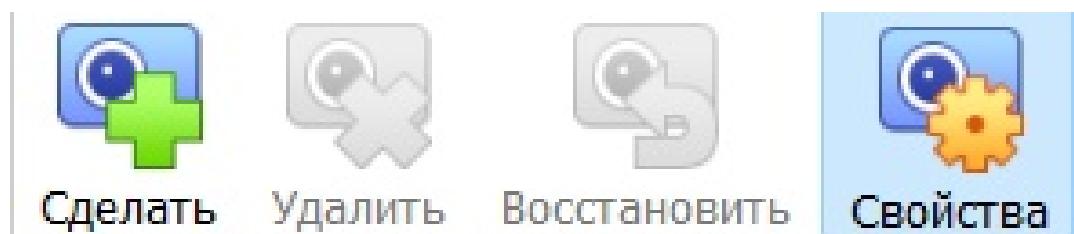


Рис. 9: Действия со снимками.

Для восстановления состояния виртуальной машины нужно выбрать нужный нам снимок, и нажать «Восстановить снимок».

VirtualBox предложит дополнительно создать снимок текущего состояния системы. Нужно поставить галочку и нажать кнопку «Восстановить», чтобы, иметь возможность вернуться к текущему состоянию, если была совершена ошибка или снимок оказался не тот, что нужен.

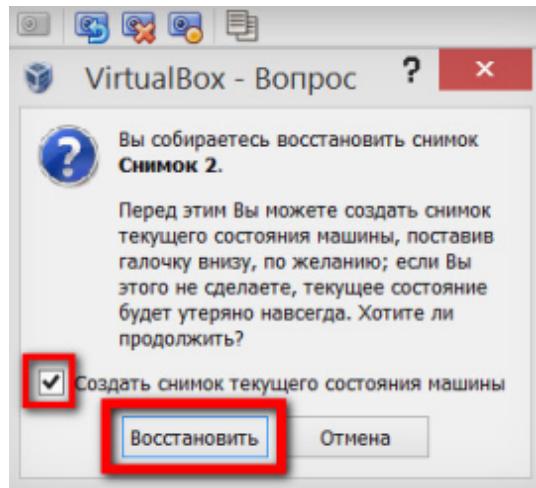


Рис. 10: Восстановление снимка.

Далее можно дать снимку текущего состояния имя и описание и нажать кнопку «OK».

После VirtualBox вернёт состояние системы к моменту создания выбранного снимка.

1.3. Клонирование виртуальной машины

1. В ходе работы с виртуальной машиной возникла необходимость перенести её с одного своего устройства на другое.

Для этого открываем Virtual Box, нажимаем правой кнопкой мыши по своей виртуальной машине и выбираем пункт клонировать. В появившемся окне указываем имя нового клона и его путь, по которому он будет сохранён. Так же в графе «Политика MAC-адреса» выбираем вариант: «Сгенерировать новые MAC-адреса всех сетевых адаптеров».

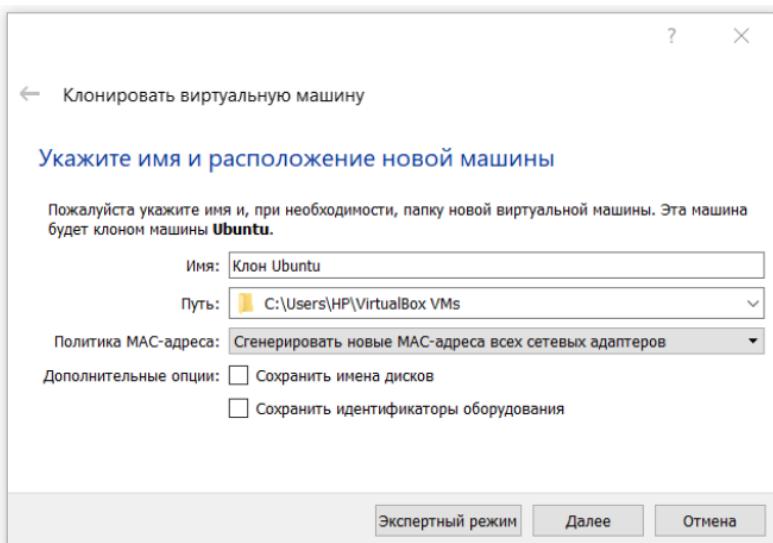


Рис. 11: Окно клонирования.

В следующем окне нужно было указать тип клонирования: полное или связное. При связном клонировании будет создана новая машина, использующая файлы виртуальных жёстких дисков клонируемой машины и нельзя перенести её на другой компьютер без переноса клонируемой. При полном клонировании, будет создана полная копия клонируемой виртуальной машины (включая все файлы виртуальных жёстких дисков). Поэтому выбираем полное клонирование.

В окне с указанием цели клонирования, указываем клонировать всё, чтобы новая машина не только отражала текущее состояние клонируемой машины, но и имела копии всех снимков её древа снимков.

Далее нажимаем на кнопку «клонировать», после чего запускается процесс клонирования. По его завершению переносим новую машину на флэшку. Это можно сделать нажав в Virtual Box на клон правой кнопкой мыши и выбрать пункт «Переместить». Или просто зайти в папку, в которую был сохранён наш клон, и переместить его уже оттуда.

Далее подсоединяем флэшку к другому компьютеру и переносим машину в папку Virtual Box. Открыв Virtual Box, нажимаем вверху на кнопку «Машина» и выбрал пункт добавить. После чего находим свою машину и нажал кнопку «Открыть».

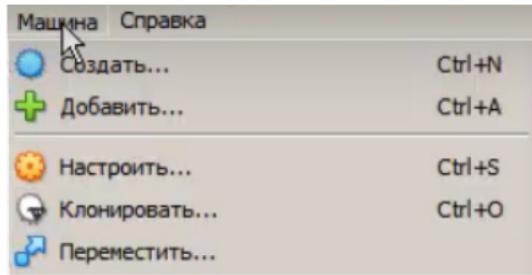


Рис. 12: Добавление виртуальной машины.

Виртуальная машина добавлена. Но зайдя в настройки, можно увидеть, что объём выделенной оперативной памяти составляет всего лишь 2 гигабайта, что слишком мало для работы с машиной. Так как наша машина находится в состоянии «Сохранена», мы не можем изменять её настройки. Поэтому нажимаем правой кнопкой мыши по перенесённому клону, и выбираем пункт «Сбросить сохранённое состояние». Сохранённое состояние, в отличие от снимка, это не состояние виртуальной машины на момент его создания, а актуальное состояние виртуальной машины, с которым работает пользователь, при каждом её запуске.

ке. Далее снова нажимаем правой кнопкой мыши по машине, выбираем пункт «Настроить...» и в «Системе» выделяем нужное количество памяти.

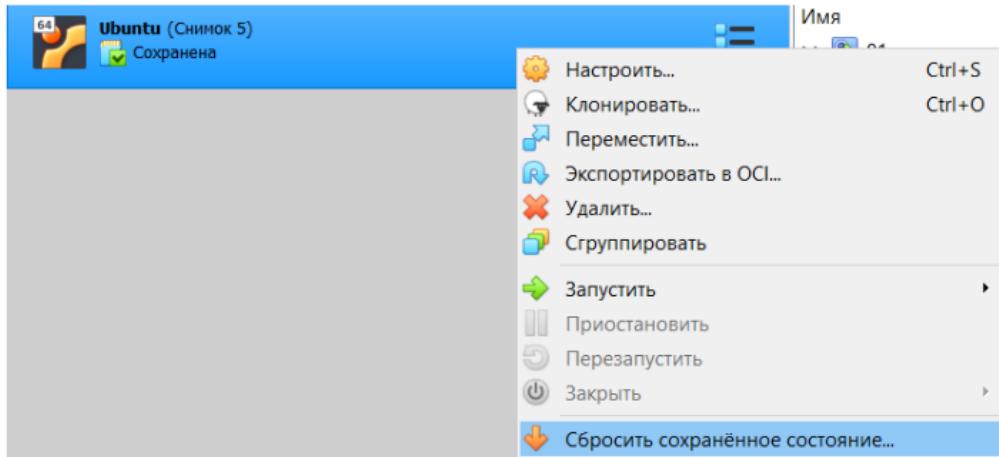


Рис. 13: Настройка памяти.

Далее заходим в «Носители» и выбираем свой жёсткий диск, так как иначе при запуске виртуальной машины мы бы ничего не увидели.

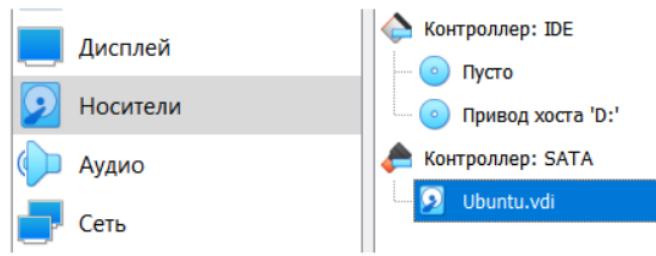


Рис. 14: Настройки. Носители.

Теперь клон виртуальной машины перемещён, добавлен на новый компьютер и с ним можно работать.

Есть и альтернативный способ переноса виртуальной машины с одного устройства на другое, с помощью функций «Экспорт» и «Импорт».

1.4. Экспорт виртуальной машины

Экспорт конфигурации виртуальной машины происходит в файл формата .ova (Open Virtual Appliance).

Это универсальный формат для хранения данных виртуальной машины, файлы .ova могут использоваться в разных программах виртуализации: VirtualBox, VMware Workstation, Microsoft Hyper-V. Виртуальная машина, экспортированная в файл .ova, затем может быть импортирована как в VirtualBox, так и в VMware Workstation, Microsoft Hyper-V.

В меню программы нужно зайти в «Файл» и выбрать пункт «Экспорт конфигураций». В открывшемся окне выбираем машину для экспорта, и нажимаем «Далее».

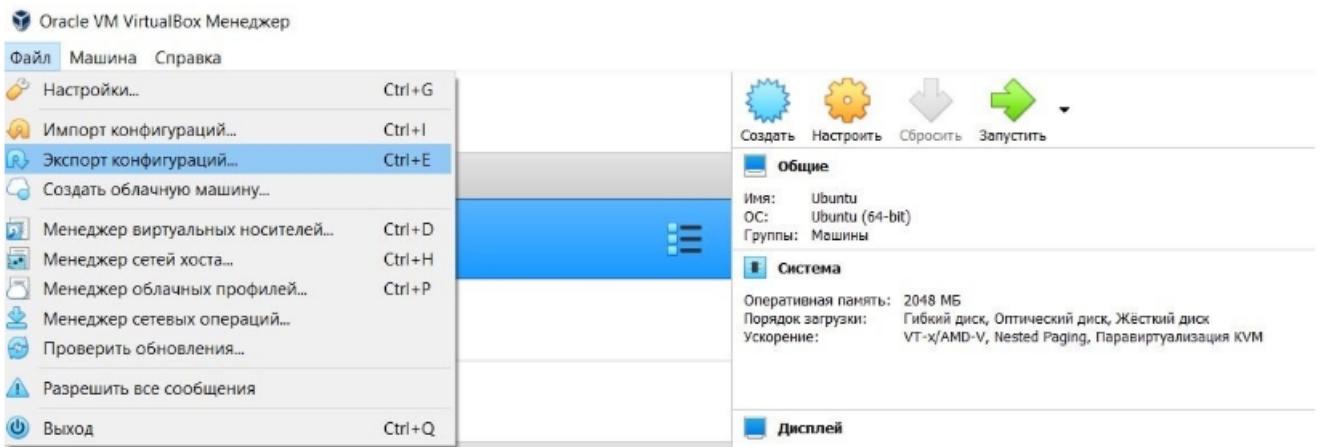


Рис. 15: «Экспорт конфигураций».

Затем выбираем место размещения файла после экспорта. Также лучше выбрать пункт «Включать MAC-адреса всех сетевых адаптеров» и затем нажимаем «Далее».

В следующем окне оставляем всё без изменений, и нажимаем кнопку “Экспорт”. Сам экспорт может занимать несколько минут, в зависимости от размера виртуальной машины.

← Экспорт конфигураций

Укажите параметры экспорта

Пожалуйста укажите формат экспорта конфигураций.

Открытый Формат Виртуализации поддерживает только **ovf** или **ova** расширения. Если Вы используете расширение **ovf**, каждый файл будет записан отдельно. Если Вы используете расширение **ova**, все файлы будут скомбинированы в один архив Открытого Формата Виртуализации.

Облачная Инфраструктура Oracle поддерживает только экспорт на удалённые облачные серверы. Главный виртуальный диск каждой из выбранных машин будет выгружен на удалённый сервер.

Формат: Открытый формат виртуализации 1.0

Пожалуйста выберите имя файла для экспорта конфигурации. Кроме того, Вы можете задать определённое количество опций, влияющих на размер и содержимое конечного архива.

Файл: C:\Users\user\Documents\Ubuntu.ova

Политика MAC-адреса: Включать MAC-адреса всех сетевых адаптеров

Дополнительно: Создать Manifest-файл

Включить ISO файлы образов

Рис. 16: MAC-адреса сетевых адаптеров.

После экспорта в указанном месте создается файл, который уже необходимо будет импортировать.

1.5. Импорт виртуальной машины

Импорт виртуальной машины

Теперь необходимо скопировать файл на флэшку. Или же можно воспользоваться облочным хранилищем, так как возможно при попытке перенести его на флешку, может возникнуть ошибка: «Файл слишком велик для конечной файловой системы». Она возникает, если передаётся файл размером более четырёх гигабайт на носитель, неспособный с ним работать. Для устранения этой ошибки можно воспользоваться форматированием флешки или разбитием файла с виртуальной машины на несколько частей. Был также опробован способ сжатия виртуальной машины в zip файл, но она оказалась практически без сжимающей составляющих.

Далее на втором компьютере заходим в программу Virtual Box и нажимаем вверху «Файл» и выбираем пункт «Импорт конфигураций».

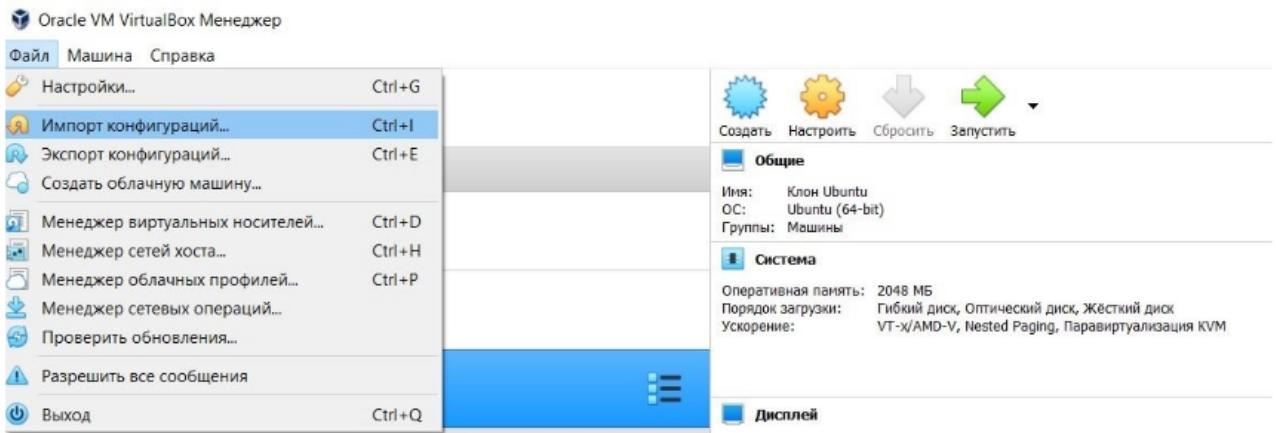


Рис. 17: Импорт конфигураций.

В окне импорта выбираем место размещения файла виртуальной машины, нажимаем «Далее». В следующем окне можно изменить параметры импорта, например, увеличить количество процессоров. Также желательно «Включать

(сгенерировать новые) MAC-адреса всех сетевых адаптеров», и нажимаем «Импорт».

Импорт также как и экспорт в зависимости от размера виртуальной машины может занимать несколько минут.

После импорта виртуальная машина появляется в списке и с ней уже можно будет работать.

1.6. Увеличение объёма памяти виртуальной машин

В какой-то момент при работе за виртуальной машиной, высветилось сообщение о том, что память заканчивается.

Виртуальные жесткие диски — это форматы файлов образа диска, которые имеют функциональные возможности, как и у физического жесткого диска. В первую очередь они предназначены для использования с виртуальными машинами. Белоусов С., в своей статье отмечает: "Как и физический диск, виртуальный носитель имеет размер и ёмкость, которые указывают при создании диска. Только в отличие от физического носителя его можно расширять".

Для начала вводим в поисковике Linux-а «Диски» и, нажав на иконку диска, смотрим насколько он заполнен, и сколько памяти осталось.

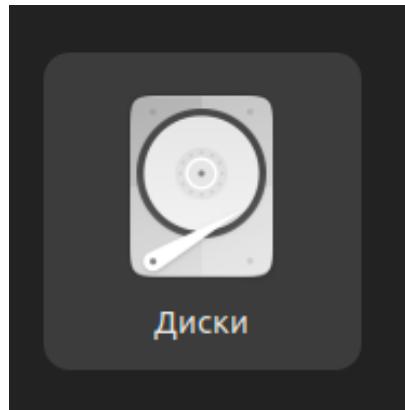


Рис. 18: Иконка диска.

Также количество оставшейся на диске памяти можно проверить, введя в терминале команду: `df`. Но по умолчанию она выводит информацию в байтах. Для более простой оценки свободного места, можно использовать команду: `df -h`. Она уже выводит размеры в гигабайтах, мегабайтах и килобайтах.

Необходимо выключить виртуальную машину, а не сохранить, чтобы можно было изменить её настройки. Далее в меню VirtualBox выбираем пункт «Файл» и нажимаем на «Менеджер виртуальных носителей...».

В открывшемся окне заходим в свойства диска и находим его размер. Двигая за ползунок, изменяем его объём с 16 гигабайт до 33 и нажимаем «Применить».

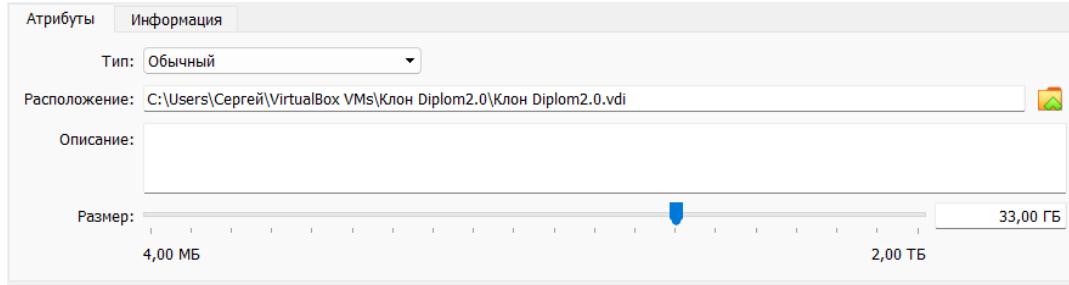


Рис. 19: Изменение размера диска.

При повторном открытии виртуальной машины было обнаружено, что размер памяти не изменился. Мы пришли к выводу, что для того чтобы увеличить объём памяти жёсткого диска, нужно сделать клон текущего состояния виртуальной машины или удалить снимки. К сожалению изменять размер диска, просто нельзя, пока существуют его точки сохранения. Поэтому снова выключаем машину, и делаем клон её текущего состояния. Он делается точно так же, как когда мы делали клон для переноса его с одного устройства на другое. Единственное отличие в том, что в этот раз выбираем клонировать не всё, а только текущее состояние.

Увеличив в «Менеджере виртуальных носителей...» уже объём памяти клона, заходим в настройки виртуальной машины, увеличиваем размер памяти в «Системе» и выбираем соответствующий диск в «Носителях», так же как это делали, после переноса клона на другое устройство. Далее запускаем новую машину и заходим в «Диск», где видим, что его размер стал больше, и появилась дополнительная область памяти.

Чтобы задействовать его, нажимаем на иконку настроек, сдвигаем ползунок до конца вправо, чтобы можно было использовать всю новую память и

нажимаем «Применить», после чего вводим свой пароль.

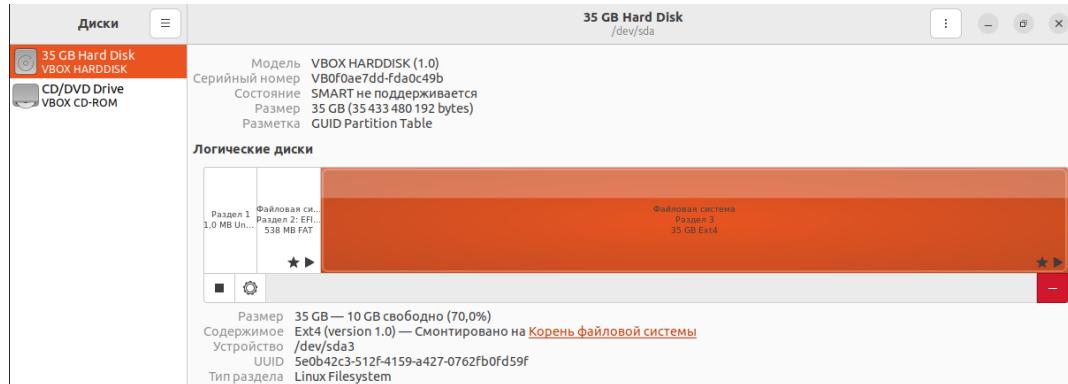


Рис. 20: Увеличенный диск.

Но опять ничего не получилось. Чтобы наконец всё заработало, нужен установочный файл Ubuntu нашей виртуальной машины. Он остался на прошлом устройстве, с которого мы и клонировали машину. Воспользовавшись гуглом, перенесли файл, так как флэшка не позволяет загрузить в неё что-либо весом более 4 гигабайт без её форматирования.

После перенесения установочный файл на новое устройство, необходимо добавить его в настройках виртуальной машины. Для этого нажать «Настроить», зайти в «Носители», кликнуть по пустому диску, и в «Оптическом приводе» выбрать файл установки.

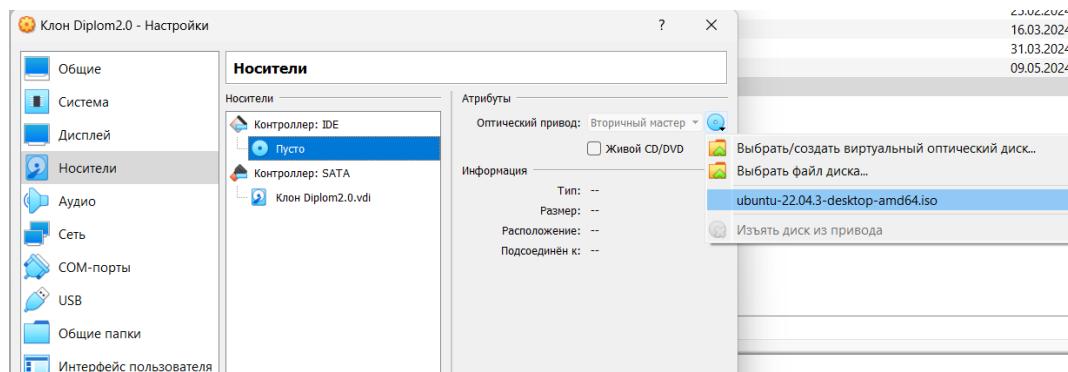


Рис. 21: Добавление установочного файла.

При запуске виртуальной машины, нужно выбрать пункт "Try or Install

Ubuntu" и далее нажать "Try Ubuntu".

Необходимо так же установить приложение GParted.

- Первым шагом нужно обновить списки пакетов, введя в терминале команду: «`sudo apt-get update`».
- Далее установить GParted командой: `sudo apt-get install gparted`.
- Ввести пароль, когда это предложат. Сам пароль и то как он вводится будет не видно.
- После установки GParted, его наконец-то можно запустить из меню приложений или с помощью команды: `sudo gparte`.

В открывшемся меню нужно кликнуть правой кнопкой мыши по области `/dev/sda3` и выбрать пункт «Изменить размер/Переместить». Далее, двигая за ползунок с правого края, можно наконец позволить виртуальной машине использовать всё добавленное, свободное пространство. Снова нажимаем на кнопку «Изменить размер/Переместить». И в меню GParted кликаем по зелёной галочке, чтобы применить все внесённые изменения. После недолгой загрузки закрываем GParted.

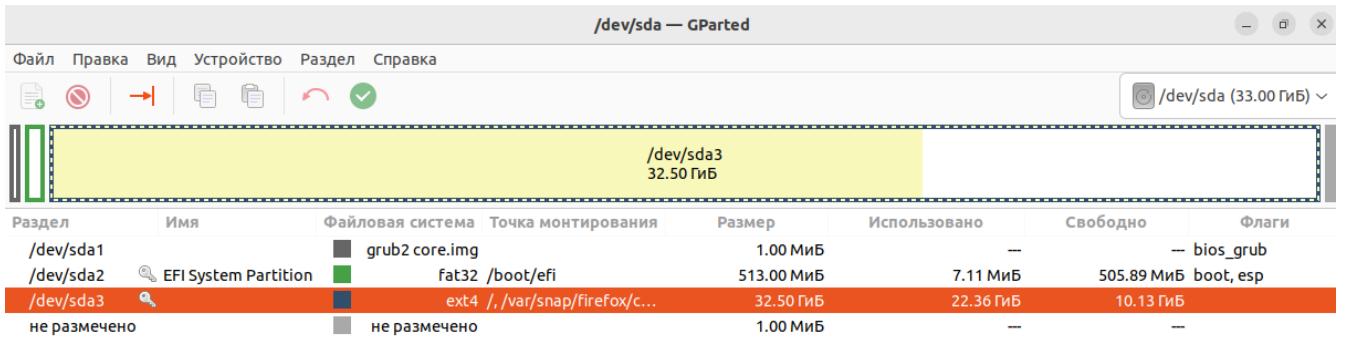


Рис. 22: Gparted.

Последний раз убеждаемся, что размер диска увеличен и свободное пространство используется, с помощью команды: `df -h`.

Чтобы при каждом запуске виртуальной машины, установочный файл не

Файл.система	Размер	Использовано	Дост	Использовано%	Смонтировано в
tmpfs	1,2G	1,6M	1,2G	1%	/run
/dev/sda3	32G	23G	8,1G	74%	/
tmpfs	6,0G	0	6,0G	0%	/dev/shm
tmpfs	5,0M	4,0K	5,0M	1%	/run/lock
/dev/sda2	512M	6,1M	506M	2%	/boot/efi
tmpfs	1,2G	1,7M	1,2G	1%	/run/user/1000

Рис. 23: Вывод команды df -h.

предлагал установить Ubuntu, нужно удалить его в настройках виртуальной машины так же, как мы его и добавляли.

2. Работа с GitHub

2.1. Установка git, nodejs, npm и grunt

Для установки всего необходимого, нам нужно открыть терминал, нажав на его иконку.

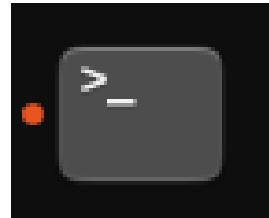


Рис. 24: Терминал.

Далее, для скачивания, необходимо ввести соответствующие команды:

```
git – sudo apt install git  
nodejs – sudo apt install nodejs  
npm - sudo apt install npm  
grunt - sudo apt install grunt
```

Чтобы удостовериться, что всё правильно скачалось, можно узнать версию данного продукта. Например: `git --version`

2.2. Работа с репозиторием

Для работы с репозиторием сначала нужно зарегистрироваться на GitHub, перейдя по ссылке: <https://github.com/> и заполнить всю необходимую информацию о себе [6].

Затем переходим по ссылке <https://github.com/nickkolok/chas-ege/>. Далее нажимаем на зелёную кнопку с надписью «Code», и копируем ссылку репозитория. Лучше сделать это сразу, потому что Ubuntu на VirtualBox может сильно нагружать компьютер, и открыть вкладку с браузером может быть проблематично из-за нагрузки. (Если возникли проблемы с копированием ссылки, то можно открыть браузер внутри Ubuntu, перейти по ссылке и скопировать ссылку репозитория в нём.)

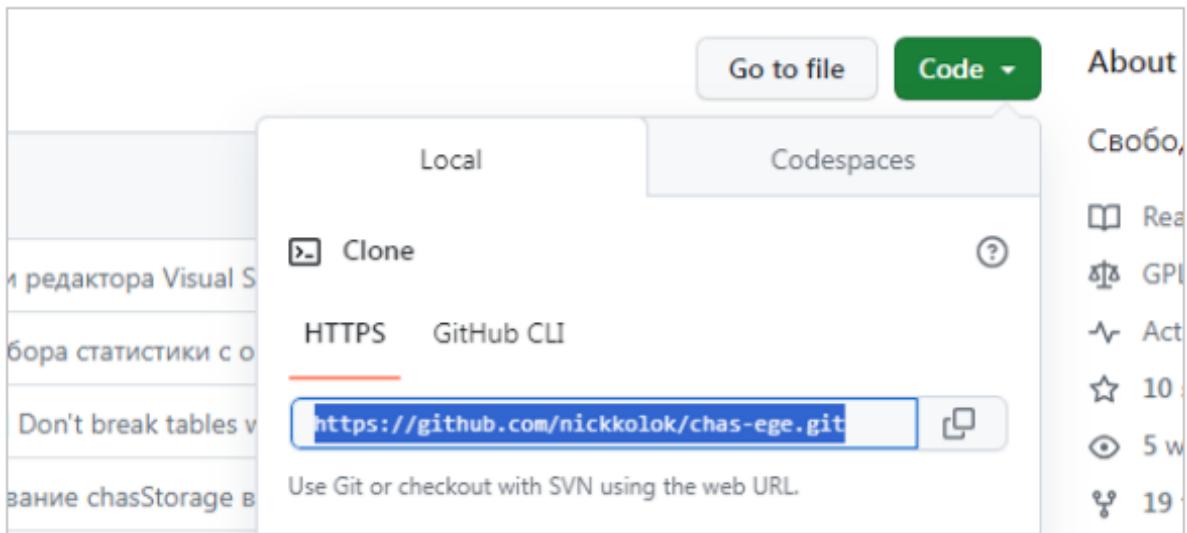


Рис. 25: Github. Ссылка на репозиторий.

- Далее снова заходим в терминал и создаём папку на рабочем столе командой: `mkdir <название папки>`. Можно убедиться, что папка создана, с помощью команды: `ls`. Мы увидим все папки на рабочем столе, среди которых должна быть только что созданная. Затем заходим в папку командой: `cd <название папки>`, и клонируем себе репозиторий командой: `git clone <ссылка на`

репозиторий >

- Добавляем себе ссылку на основной репозиторий проекта с помощью команды: git remote add upstream <ссылка на репозиторий> и убеждаемся, что он подключился, командой: git fetch upstream
- Собираем проект командой: grunt. Важно выполнять эту команду в папке, в которую мы и склонировали репозиторий.
- Открываем файл dist/sh/otladka.html в браузере командой: «open otladka.html», и запускаем любой шаблон, для проверки, в открывшемся окне.

```
aleNDAR@aleNDAR-VirtualBox:~/repo
aleNDAR@aleNDAR-VirtualBox:~/repo$ cd chas-ege
aleNDAR@aleNDAR-VirtualBox:~/repo/chas-ege$ cd dist
aleNDAR@aleNDAR-VirtualBox:~/repo/chas-ege/dist$ cd sh
aleNDAR@aleNDAR-VirtualBox:~/repo/chas-ege/dist/sh$ ls
autoapidoc.html      formula.js    matr.html     otladka.js   polnmat.html  sluchcounter.html sortstr.html
chas-ege_to_reshuege.js katalog.html mini.html    pechmat.html poln.min.js   sluch.html       test.html
dvig_lz.js           katalog.js   mini.js     pechmat.js  sklon1.html  sluchmat.js    timetest.html
formula.html         legacy.html  otladka.html poln.js      sklon1.js   sluchmat.min.js timetest.js
```

Рис. 26: Путь к файлу otladka.html.

2.3. Создание SSH ключа

Создание SSH-ключа для возможности подключиться к удалённому репозиторию происходит в несколько этапов. Для начала нужно проверить наличие ключа, введя следующие команды:

```
cd / .ssh
```

```
ls
```

Если файлов с названиями `id_dsa` и `id_dsa.pub` нет (открытый и приватный ключ), то можно создать их используя команду:

```
ssh-keygen -o
```

Далее нужно открыть содержимое файла `dsa.pub` командой:

```
cat / .ssh/id_dsa.pub
```

Далее добавили свой ключ себе в аккаунт на GitHub, чтобы иметь возможность подключиться к удалённому репозиторию [2]. А также прописали ещё одну команду, необходимую для подключения.

```
alenadar@alenadar-VirtualBox:~/repo/chas-ege$ git remote -v
myfork https://github.com/kalendar2/chas-ege.git (fetch)
myfork https://github.com/kalendar2/chas-ege.git (push)
origin https://github.com/nickkolok/chas-ege.git (fetch)
origin https://github.com/nickkolok/chas-ege.git (push)
alenadar@alenadar-VirtualBox:~/repo/chas-ege$ git remote set-url origin git@github.com:nickkolok/chas-ege.git
alenadar@alenadar-VirtualBox:~/repo/chas-ege$ git remote -v
myfork https://github.com/kalendar2/chas-ege.git (fetch)
myfork https://github.com/kalendar2/chas-ege.git (push)
origin git@github.com:nickkolok/chas-ege.git (fetch)
origin git@github.com:nickkolok/chas-ege.git (push)
alenadar@alenadar-VirtualBox:~/repo/chas-ege$ git status
Текущая ветвь: devol
```

Рис. 27: Подключению к удалённому репозиторию с помощью команды.

Переходим в репозиторий на GitHub «ЧАС-ЕГЭ» и нажимаем кнопку «Fork» справа вверху.

Открыв VirtualBox и войдя в свою виртуальную машину, открыли терминал и перешли в папку с «Час ЕГЭ» с помощью команды `cd git/chas-ege`.

Представление гиту выглядит следующим образом:

```
git config --global user.name «Фамилия Имя»
```

```
git config --global user.email «электронная почта пользователя»
```

Можно придать выводу гита красные и зелёные цвета с помощью команд:

```
git config --global color.ui true
```

Добавление своего форка на гитхаб в список удалённых репозиториев: `git remote add myfork git@github.com:«GitHubNik»/chas-ege.git`, где «GitHubNik» - ник пользователя на гитхабе.

Основные команды для создания и отправления изменений в удалённый репозиторий:

Переключение на основную ветку (devel):

```
git checkout devel
```

Её обновление (В некоторых случаях применима также команда: `git pull origin devel`):

```
git fetch origin devel
```

Создание новой ветки:

```
git checkout -b newtask-777
```

Проверка изменений, а также самой ветки:

```
git status
```

Добавление всех изменений:

```
git add .
```

Добавление всех изменений:

```
git commit -m «Внесены изменения в файл ...»
```

Отправка изменений в удалённый репозиторий:

```
git push myfork newtask-777:myfork newtask-777:
```

2.4. Игнорирование файлов

Для того, чтобы сообщить Git, какие ненужные файлы или каталоги нужно игнорировать, можно создать `.gitignore` файл.

Для этого необходимо открыть Терминал и перейти к расположению репозитория Git. Далее создаётся файл для репозитория, с помощью команды `touch .gitignore`.

Если файл уже был отправлен в репозиторий, необходимо отменить отслеживание файла, прежде чем добавлено правило игнорирования, с помощью команды: `git rm -cached FILENAME`.

Файл невидим, как и все файлы с точкой в начале названия файла, но его можно увидеть с помощью команды `ls -a`. Зайдя в него, нужно записать путь к файлу или каталогу, который будет игнорироваться.

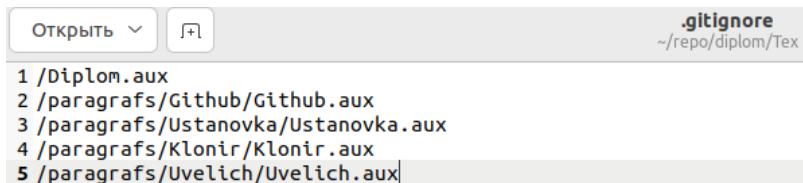


Рис. 28: Файл `.gitignore` с списком файлов для игнорирования в нём.

Это не единственный способ задания игнорирования файла или каталога. Можно также сообщить Git всегда игнорировать определенные файлы во всех репозиториях на компьютере. Для этого, например, каталог нужно добавить в файл с именем `ignore`, расположенным внутри каталога `/.config/git`.

Также можно вообще не создавать файл `.gitignore`. Этот метод можно использовать для локально создаваемых файлов, которые не должны создавать другие пользователи. Для этого, используя текстовый редактор, нужно открыть файл, вызываемый `.git/info/exclude` в корневом каталоге репозитория Git.

3. Практическая реализация шаблонов

3.1. Физический смысл производной, нахождение скорости процесса

Пример пары заданий одного типа, на основе которых создан следующий шаблон.

Задание 3. Материальная точка движется прямолинейно по закону $x(t)=\frac{1}{3}t^3-5t^2+45t+130$, где x – расстояние от точки отсчёта в метрах, t – время в секундах, измеренное с момента начала движения. В какой момент времени её скорость была равна 20 м/с?

Задание 6. Материальная точка движется прямолинейно по закону $x(t)=\frac{1}{6}t^3-2t^2+6t+250$, где x – расстояние от точки отсчёта в метрах, t – время в секундах, измеренное с момента начала движения. В какой момент времени её скорость была равна 96 м/с?

Первая часть шаблона выглядит следующим образом:

```
1 (function() {
2   'use strict';
3
4   var a = sluchch(0.5, 1, 0.5);
5   var b = 0;
6   if (a == 0.5) {
7     b = 6;
8   } else {
9     b = 3;
10 }
11
12 function plusmin(member) {
13   var el = math.parse(member);
14   return math.simplify(el).toTex();
15 }
```

```

16
17 var x2 = (sluchch(-15, -1) * (b / 3)).pm();
18 if (x2 < 0) {
19     var x1 = Math.abs(x2) + sluchch(6, 36, 2);
20     var ans = x1;
21 } else {
22     var x1 = -x2 - sluchch(6, 36, 2);
23     var ans = x2;
24 }
25
26 var c = Math.abs(a * x1 * x2) + sluchch(2, 100).pm();
27 var d = sluchch(-999, 999);

```

Здесь объявляются переменные таким образом, чтобы генерируемые значения, максимально соотносились с исходной задачей.

В этой части также используются специальные встроенные функции, такие как:

`sluchch(a, b)` – функция случайным образом возвращает число из диапазона от `a` до `b`. Если функции добавить третье число `sluchch(a, b, c)`, то она будет возвращать случайно число уже с шагом `c`.

`Math.abs(a)` – функция возвращает модуль числа `a`.

`function plusmin(member)` – функция принимает аргумент, содержащий в себе пример, наподобие: `1*a+-b`. И возвращает его упрощённый вариант: `a-b`.

Вторая часть выглядит следующим образом:

```

1 chas2.task.setTask({
2     text: 'Материальная точка движется прямолинейно по закону:' +
3         '$$ x(t)=\frac{1}{2}at^2+bx+c $' + plusmin('x^3+' + ((-a * (x1 + x2)) /
4             2)) + plusmin('x^2+' + (a * x1 * x2 + c)) + plusmin('x+' + d) + ', $$'
5         + 'где $x$ - расстояние от точки отсчёта в метрах, $t$ - время в секундах
6         , измеренное с момента начала движения.' +
7         ' В какой момент времени её скорость была равна ' + '$v$' + ' м/с?', +
8         analys: 'Найдём закон изменения скорости:' +
9         '$$ v(t)=x^{\prime }(t)=\frac{d}{dt}(x(t))=a^{\prime }(t)x(t)+b^{\prime }(t)x(t)+c^{\prime }(t)$' + plusmin(a + 'x^2+' + (-a * (x1 + x2))) +

```

```

plusmin('x+' + (a * x1 * x2 + c)) +
8      '$$ Чтобы найти, в какой момент времени $t$ скорость была равна $' + c +
' $ м/с, решим уравнение: $$',
9      + plusmin(a + 'x^2+' + (-a * (x1 + x2))) + plusmin('x+' + (a * x1 * x2 +
c)) + ' = ' + c +
10     ' \Rightarrow ' + plusmin(a + 'x^2+' + (-a * (x1 + x2))) + 'x' +
plusmin('+' + (a * x1 * x2)) + ' = 0 $$',
11     '$$ D = b^2-4ac = ' + (-a * (x1 + x2)) + '^2+4 \cdot ' + -a * a * x1 *
x2 + ' = ' + (Math.pow((-a * (x1 + x2)), 2) - (4 * a * a * x1 * x2)) +
12     '$$ + '$$ \begin{cases} x_1= \frac{-b+\sqrt{D}}{2a} = ' +
13     '\frac{' + a * (x1 + x2) + '+\sqrt{' + (Math.pow((-a * (x1 + x2)), 2) -
(4 * a * a * x1 * x2)) + '} }{ ' + 2 * a + '}', +
14     ', = ' + x1 + '\sqrt{' + 'x_2= \frac{-b-\sqrt{D}}{2a}', +
15     ', = ' + '\frac{' + a * (x1 + x2) + '-\sqrt{' + (Math.pow((-a * (x1 +
x2)), 2) - (4 * a * a * x1 * x2)) + '} }{ ' + 2 * a + '}', +
16     ', = ' + x2 + '\end{cases}$$',
17
18 answers: ans,
19 });
20 })();
21 //119979

```

Вторая часть программы делится на несколько составных частей:

условие задачи, записанное после `text`:

решение, записанное после `analys`:

ответ, записанный после `answer`

В этом шаблоне мы вводим переменные, генерирующие случайные значения, которые используем для составления ответа. И на основе ответа мы уже составляем саму задачу и её решение.

В этой части также используются некоторые команды \LaTeX [3], например:

`\frac` – функция выводящая дробь на экран.

`\Rightarrow` – функция выводящая двойную стрелку вправо

`\begin{cases}` и `\end{cases}` – окружение, выводящая фигурную скобку на

экран.

Пример сгенерированных шаблоном задач

Материальная точка движется прямолинейно по закону:

$$x(t) = \frac{1}{6}x^3 + \frac{-7}{2}x^2 - 76x - 796,$$

где x - расстояние от точки отсчёта в метрах, t - время в секундах, измеренное с момента начала движения.
В какой момент времени её скорость была равна -40 м/с?

Найдём закон изменения скорости:

$$v(t) = x'(t) = \frac{x^2}{2} - 7x - 76$$

Чтобы найти, в какой момент времени t скорость была равна -40 м/с, решим уравнение:

$$\frac{x^2}{2} - 7x - 76 = -40 \Rightarrow \frac{x^2}{2} - 7x - 36 = 0$$

$$D = b^2 - 4ac = -7^2 + 4 \cdot 18 = 121$$

$$\begin{cases} x_1 = \frac{-b + \sqrt{D}}{2a} = \frac{7 + \sqrt{121}}{1} = 18 \\ x_2 = \frac{-b - \sqrt{D}}{2a} = \frac{7 - \sqrt{121}}{1} = -4 \end{cases}$$

Материальная точка движется прямолинейно по закону:

$$x(t) = \frac{1}{6}x^3 + \frac{9}{2}x^2 + 40x - 721,$$

где x - расстояние от точки отсчёта в метрах, t - время в секундах, измеренное с момента начала движения.
В какой момент времени её скорость была равна 60 м/с?

Найдём закон изменения скорости:

$$v(t) = x'(t) = \frac{x^2}{2} + 9x + 40$$

Чтобы найти, в какой момент времени t скорость была равна 60 м/с, решим уравнение:

$$\frac{x^2}{2} + 9x + 40 = 60 \Rightarrow \frac{x^2}{2} + 9x - 20 = 0$$

$$D = b^2 - 4ac = 9^2 + 4 \cdot 10 = 121$$

$$\begin{cases} x_1 = \frac{-b + \sqrt{D}}{2a} = \frac{-9 + \sqrt{121}}{1} = -20 \\ x_2 = \frac{-b - \sqrt{D}}{2a} = \frac{-9 - \sqrt{121}}{1} = 2 \end{cases}$$

3.2. Вычисления и преобразования

Задачи из ЕГЭ по математике базового уровня тип №16

$$\frac{\log_7(11^6)}{2\log_7 11}, \quad \frac{\log_7(11^6)}{2\log_7 11}, \quad \frac{\log_5(11^6)}{2\log_5 11}, \quad \frac{\log_3(5^2)}{2\log_3 5}, \quad \frac{\log_7(11^8)}{2\log_7 11}, \quad \frac{\log_7(4^6)}{3\log_7 4}.$$

Шаблон задач

```
1 function(){
2   retryWithError(function() {
3     'use strict';
4     let a = s1(2, 19);
5     let b = s1(2, 9);
6     let d = s1(2, 9);
7     NAtask.setEvaluationTask({
8       expr: '(varlog(' + b + ', ' + a + '^' + d * s1(1, 9) + '))/(' + d +
9         '(varlog(' + b + ', ' + a + ')))',
10      authors: ['Алэндарь Сергей'],
11    });
12  }, 10000);
13 }();
```

Отличительной особенностью этих шаблонов от предыдущего, является другая библиотека - `setEvaluationTask`. Помимо самой генерации задачи, на основе переменных их первой части, она так же сама решает сгенерированный пример, и проверяет его ответ.

Если ответ не является удовлетворительным, а именно таким, что пользователь просто не сможет ввести его на клавиатуре (Ответ представляет из себя бесконечную десятичную дробь, или просто очень длинную), то программа, отслеживает эту ошибку с помощью цикла `retryWithError` и предпринимает ещё попытку составить задание с цельным ответом. Количество таких попыток указано внизу кода программы. Если за установленное число попыток, программа так и не получит правильно сгенерированную задачу, то она выведет сообщение

об ошибке.

За отображение логарифма на экране здесь отвечает специальная встроенная функция: `varlog`

Сгенерированные по шаблону задачи

$$\frac{\log_7 7^{20}}{5 \log_7 7} \quad \frac{\log_3 18^{45}}{9 \log_3 18} \quad \frac{\log_9 19^{40}}{5 \log_9 19} \quad \frac{\log_2 8^{12}}{3 \log_2 8} \quad \frac{\log_5 10^2}{2 \log_5 10} \quad \frac{\log_9 3^{15}}{3 \log_9 3}$$

Также эта библиотека способна сама убирать некоторые скобки, если они не влияют на решение задания, или сама выполнить арифметическую операцию. Но в некоторых случаях, это не является необходимостью. Например как в этой задаче базового типа. Здесь скобки нужны лишь для наглядности, и не оказывают никакого влияния на решение этого примера, но их всё же необходимо сохранить.

1) $(0,01)^2 \cdot 10^4 : 3^{-2}$ 3) $(0,1)^2 \cdot 10^4 : 2^{-3}$

2) $(0,01)^2 \cdot 10^5 : 4^{-2}$ 4) $(0,1)^5 \cdot 10^3 : 7^{-2}$

Чтобы программа не высчитывала сама то, что стоит оставить, используются следующие команды:

`forceBrackets` - для сохранение скобок.

`divideColon` - для вывода знака деления «:».

Шаблон задачи, использующий эти специальные команды

```
1 function() {
2   retryWhileError(function() {
3     'use strict';
4     NAtask.setEvaluationTask({
5       expr: 'forceBrackets(' + [0.01, 0.1].iz() + ')^' + sl(2, 6) + '*' +
6         'divideColon(10^' + sl(2, 6) + ', ' +
7           sl(2, 9) + '^' + sl(-1, -5) + ')',
```

```
8     authors: [ 'Алeндарь Сергеy' ] ,  
9     } );  
10    }, 10000);  
11 } )();
```

Сгенерированные по шаблону задания

$$(0,01)^2 \cdot 10^6 : 5^{-2} \quad (0,1)^5 \cdot 10^5 : 6^{-2}$$

$$(0,1)^5 \cdot 10^6 : 9^{-4} \quad (0,01)^3 \cdot 10^3 : 6^{-3}$$

3.3. Логарифмические уравнения

Задача №77381 тип №6

Решите уравнение $\log_5(7-x) = \log_5(3-x) + 1$.

Решение.

Заметим, что $1 = \log_5 5$ и используем формулу $\log_a b + \log_a c = \log_a bc$. Имеем:

$$\log_5(7-x) = \log_5(3-x) + 1 \Leftrightarrow \log_5(7-x) = \log_5(3-x) + \log_5 5 \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} 3-x > 0, \\ 7-x = 5(3-x) \end{cases} \Leftrightarrow \begin{cases} -x > -3, \\ 7-x = 15-5x \end{cases} \Leftrightarrow \begin{cases} x < 3, \\ x = 2 \end{cases} \Leftrightarrow x = 2.$$

Ответ: 2.

Первая часть шаблона задачи №77381

Здесь, помимо ввода переменных, и подбора подходящего корня, происходит проверка корня на его вхождение в ОДЗ, а также проверяется сколько знаков после запятой он имеет. В данном случае, программа контролирует, чтобы ответ имел не более 3 знаков. Сделано это, чтобы пользователю было удобнее вводить ответ.

```
1 (function() {
2   retryWhileError(function() {
3     'use strict';
4
5     var v = s11();
6     var d = v ? '+' : '-';
7     var znak = v ? 1 : (-1);
8
9     var a = sluchch(2, 3);
10    var b;
11    if (a === 2) {
12      b = znak * sluchch(4, 64, 8).pm();
13    } else {
14      b = sluchch(15, 69, 27);
```

```

15    }
16
17    var y = sluchch(1, 3);
18    var c = sluchch(1, 3);
19    var t = -znak * (12 + b) / a.pow(c);
20    var u = sluchch(1, 4);
21    var z = y * a.pow(c) + u;
22
23    var x = (t * a.pow(c) - b) / (z.plusminus() - y.plusminus() * a.pow(c));
24    genAssertZ1000(x, "Корень имеет более 3 знаков после запятой");
25
26    var x1 = -b.plusminus() / z;
27    var x2 = -t.plusminus() / y;
28    genAssertZ1000(x1, "'Корень имеет более 3 знаков после запятой");
29    genAssertZ1000(x2, "'Корень имеет более 3 знаков после запятой");
30    genAssert(x > -b.plusminus() / z, "Корень не попадает в ОДЗ");
31    genAssert(x > -t.plusminus() / y, "Корень не попадает в ОДЗ");
32
33    var ans = x1;
34    var ans1;
35    var q = b;
36    var j = z;
37
38    if (x2 > x1) {
39        ans = x2;
40        q = t;
41        j = y;
42    }
43
44    if (ans.isZ()) { //Дробная часть для смешанного числа
45        ans1 = ' ';
46    } else {
47        ans = Math.trunc(ans);
48        ans1 = '\frac{' + (q.abs() % j) + '}{' + j + '}';
49        if (ans === 0) {
50            if (-q.plusminus() < 0) {
51                ans = '-';
52            } else ans = ' ';

```

```
53     }
54 }
```

Вторая часть шаблона задачи №77381

Ещё одной отличительной особенностью этого шаблона, является его библиотека `setAdditiveEquationTask`. Благодаря ней, части уравнения, записанные в квадратные скобки после слова `parts`: будут каждый раз переставляться при запуске программы, что создаёт больше разнообразия в задачах, использующих этот код.

Эта часть также содержит некоторые специальные встроенные функции:
`[a,b,c].slag()` – функция выводящая пример, в котором все элементы массива слагаемыми в различном порядке.

`a.pow(b)` – функция возводящая число в степень И некоторые команды из LATEX, например:

`\cdot` – выводящая знак умножения в виде точки.

```
1   chas2.task.setAdditiveEquationTask({
2     parts: ['-\log_{' + a + '}{{(' + [b.plusminus(), z + 'x']}.slag() + ')}}',
3       '\log_{' + a + '}{{(' + [t.plusminus(),
4         y + 'x'
5       ]}.slag() + ')}}', c],
6     roots: x,
7     enablePartsSubtraction: [1, 0],
8   }, {
9     tags: {
10       log: 1,
11     },
12     analys: 'Решение:' +
13       '$$ \log_{' + a + '}{{(' + b.plusminus() + '+' + z + 'x' + ')}} = ' +
14       '\log_{' + a + '}{{(' + t.plusminus() + '+' +
15         y + 'x' + ')}} + {{' + c + '}} $$ ' +
16       '$${' + c + '} = \log_{' + a + '}{{(' + a + '^' + c + ')}}, \\\\
17       \log_{a}{b} + \log_{a}{c} = ' +
18       '\log_{a}{bc} \\ \Rightarrow \log_{' + a + '}{{(' + t.plusminus()
```

```

+ '+' + y + 'x' + ')}+\\"log_{' + a +
16   '}{' + a.pow(c) +
17   '}=\\"log_{' + a + '}{{(( ' + t.plusminus() + '+' + y + 'x' + ')}\\"cdot'
+ '{' + a.pow(c) +
18   '} } \\" \\"Rightarrow$$' +
19   '$$\\Rightarrow\\ \\begin{cases} {' + b.plusminus() + '+' + z + 'x' +
'>0} \\\\ {' + t.plusminus() + '+' + y +
20   'x' +
21   '>0}\\"\\ {{(' + b.plusminus() + '+' + z + 'x' + ')}=\\" + t.plusminus()
+ '+' + y + 'x' + ')}\\"cdot' + a.pow(c) +
22   '\\\\end{cases}\\Rightarrow \\begin{cases} {' + z + 'x}>' +
(-b.plusminus() + '} \\"\\ {{(' + y + 'x}>' + (-t.plusminus() +
23   '}\\{{{(' + b.plusminus() + '+' + z + 'x' + ') = ' + t.plusminus() *
a.pow(c) + '+' + y * a.pow(c) + 'x' +
24   '}\\\\end{cases}\\Rightarrow \\begin{cases} x>\\frac{' +
(-b.plusminus() + '}{{(' + z + ')}, +
25   '}\\\\~\\\\ x>\\frac{' + (-t.plusminus() + '}{{(' + y + ')}}\\\\~\\\\ {{(' +
(-y * a.pow(c) + z).plusminus() +
26   '}x=' + (t * a.pow(c) - b) + '}\\\\end{cases}\\Rightarrow' +
27   '\\begin{cases} x>' + ans + ',' + ans1 + '}\\\\~\\\\ x={'} + (t *
a.pow(c) - b) / ((-y * a.pow(c) + z).plusminus() +
28   '}\\\\end{cases}$$' +
29
30   'ОТВЕТ: $x=' + (t * a.pow(c) - b) / ((-y * a.pow(c) +
z).plusminus())+'.' + '$',
31   });
32   );
33 })();

```

Задачи сгенерированные по шаблону

Найдите корень уравнения

$$-\log_2(60 + 5x) = -2 - \log_2(x + 18)$$

Решение:

$$\log_2(60 + 5x) = \log_2(18 + x) + 2$$

$$2 = \log_2 2^2, \quad \log_a b + \log_a c = \log_a bc \Rightarrow \log_2(18 + x) + \log_2 4 = \log_2((18 + x) \cdot 4) \Rightarrow$$

$$\Rightarrow \begin{cases} 60 + 5x > 0 \\ 18 + x > 0 \\ (60 + 5x) = (18 + x) \cdot 4 \end{cases} \Rightarrow \begin{cases} 5x > -60 \\ x > -18 \\ 60 + 5x = 72 + 4x \end{cases} \Rightarrow \begin{cases} x > \frac{-60}{5} \\ x > \frac{-18}{1} \\ x = 12 \end{cases} \Rightarrow \begin{cases} x > -12 \\ x = 12 \end{cases}$$

Ответ: $x = 12$.

Найдите корень уравнения

$$-\log_3(2x + 9) + \log_3(69 + 20x) = 2$$

Решение:

$$\log_3(69 + 20x) = \log_3(9 + 2x) + 2$$

$$2 = \log_3 3^2, \quad \log_a b + \log_a c = \log_a bc \Rightarrow \log_3(9 + 2x) + \log_3 9 = \log_3((9 + 2x) \cdot 9) \Rightarrow$$

$$\Rightarrow \begin{cases} 69 + 20x > 0 \\ 9 + 2x > 0 \\ (69 + 20x) = (9 + 2x) \cdot 9 \end{cases} \Rightarrow \begin{cases} 20x > -69 \\ 2x > -9 \\ 69 + 20x = 81 + 18x \end{cases} \Rightarrow \begin{cases} x > \frac{-69}{20} \\ x > \frac{-9}{2} \\ 2x = 12 \end{cases} \Rightarrow \begin{cases} x > -3\frac{9}{20} \\ x = 6 \end{cases}$$

Ответ: $x = 6$.

3.4. Прогрессии

Код класса арифметической прогрессии

```
1 class ArithmeticProgression {
2     constructor(first, difference) {
3         this.first = first;
4         this.difference = difference;
5         this.nmember = 0;
6     }
7
8     member(nmember) {
9         this.nmember = nmember;
10        return this.first + this.difference * (this.nmember - 1);
11    }
12
13    summa(nmember) {
14        this.nmember = nmember;
15        return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16            this.nmember;
17    }
18 }
19 let a;
20 let b;
21 let n = -1;
22 let str = prompt('Введите первый член прогрессии.', '');
23 a = Number(str);
24 str = prompt('Введите разность арифметической прогрессии.', '');
25 b = Number(str);
26 while (n <= 0) {
27     str = prompt('Введите номер члена прогрессии. Число должно быть больше нуля
28 !', '');
29     n = Number(str);
30 }
31 let progress = new ArithmeticProgression(a, b);
32 }
```

```
33 console.log(progress.member(n));  
34 console.log(progress.summa(n));
```

В классе присутствуют функции, которые высчитывают с помощью формул определённый член прогрессии, а также сумму прогрессии.

Во второй части кода пользователь вводит значения, которые передаются в класс и принимаются в качестве аргументов функциями, после чего ответ выводится на экран.

Задачи на арифметическую прогрессию

1. Грузовик перевозит партию щебня массой 210 тонн, ежедневно увеличивая норму перевозки на одно и то же число тонн. Известно, что за первый день было перевезено 2 тонны щебня. Определите, сколько тонн щебня было перевезено за девятый день, если вся работа была выполнена за 14 дней.

2. Вере надо подписать 640 открыток. Ежедневно она подписывает на одно и то же количество открыток больше по сравнению с предыдущим днем. Известно, что за первый день Вера подписала 10 открыток. Определите, сколько открыток было подписано за четвертый день, если вся работа была выполнена за 16 дней.

Шаблон использующий библиотеку со встроенным классом арифметической прогрессии

```
1 (function() {  
2   retryWhileError(function() {  
3     'use strict';  
4  
5     let a = sluchch(2, 10);  
6     let b = sluchch(2, 4);  
7     let n = sluchch(10, 24);  
8     let l1 = s11();  
9     let l2 = 1 - l1;  
10    let n2 = n - sluchch(5, 7);  
11    let arr = ['Лена', 'Ира', 'Настя', 'Маша', 'Катя', 'Вера', 'Саша', 'Оля',  
12      'Даша', 'Тоня', 'Аня', 'Ира', 'Женя'].iz();
```

```

12 let arr2 = [ 'открытка', 'валентинка', 'приглашение', 'распоряжение',
13   'письмо', 'поздравление', 'ценник', 'книга', 'документ' ].iz();
14 let arr3 = [ 'за первый день ' + arr + ' подписала ' + chislitlx(a, arr2,
15   'v'), 'вся работа была выполнена за ' + chislitlx(n, 'день') ];
16
17 NAtask.setTask({
18   text: sklonlxkand(arr).de + ' надо подписать ' +
19   chislitlx(progression.sum(n), arr2, 'v') +
20   '. Ежедневно она подписывает на одно и то же количество ' +
21   sklonlxkand(arr2).rm +
22   ' больше по сравнению с предыдущим днём.' +
23   ' Известно, что ' + arr3[11] + '.' +
24   ' Определите, сколько ' + sklonlxkand(arr2).rm + ' было подписано за ' +
25   + om.porchisl[n2].i[0] +
26   ' день, если ' + arr3[12] + '.' ,
27
28   answers: progression.member(n2),
29 });
}, 10000);
})();

```

Задачи, сгенерированные по шаблонам, использующим код арифметической прогрессии.

Бригада маляров красит забор длиной 145 метров , ежедневно увеличивая норму покраски на одно и то же число метров. Известно, что за первый и последний день в сумме бригада покрасила 58 метров забора. Определите, сколько дней бригада маляров красила весь забор.

Рабочие прокладывают тоннель длиной 140 метров, ежедневно увеличивая норму прокладки на одно и то же число метров. Известно, что за первый день рабочие проложили 11 метров туннеля. Определите, сколько метров туннеля проложили рабочие в последний день, если вся работа была выполнена за 7 дней.

Саше надо решить 390 задач. Он решает на одно и то же количество задач больше по сравнению с предыдущим днем. Известно, что за первый день Саша решил 6 задач. Определите, сколько задач решил Саша в последний день, если со всеми задачами он справился за 13 дней.

Задачи на геометрическую прогрессию

Код геометрической прогрессии

```
1 class GeometricProgression {
```

Сергею надо собрать 630 подписей. Каждый день он собирает на 3 подписи больше, чем в предыдущий день. Известно, что все подписи он собрал за 20 дней. Сколько подписей Сергей собрал в последний день, если в первый день он собрал 3 подписи?

Вере надо подписать 640 открыток. Ежедневно она подписывает на одно и то же количество открыток больше по сравнению с предыдущим днем. Известно, что за первый день Вера подписала 10 открыток. Определите, сколько открыток было подписано за четвертый день, если вся работа была выполнена за 16 дней.

```
2     constructor(first, factor) {
3
4         this.first = first;
5
6         this.factor = factor;
7
8         this.nmember = 0;
9
10    }
11
12
13    member(nmember) {
14
15        this.nmember = nmember;
16
17        return this.first * Math.pow(this.factor, this.nmember - 1);
18    }
19
20
21    summa(nmember) {
22
23        this.nmember = nmember;
24
25        if (this.factor < 0) {
26
27            throw new Error("Нельзя найти сумму знакопеременной прогрессии!");
28        }
29
30        return (this.first * (Math.pow(this.factor, this.nmember) - 1)) /
31            (this.factor - 1);
32    }
33
34
35    infinitySumma() {
36
37        if (this.factor < 0) {
38
39            throw Error;
40        }
41
42        if (this.factor > 1) {
43
44            throw new Error("Прогрессия не является бесконечной!");
45        }
46
47        return this.first / (1 - this.factor);
48    }
49
50}
```

```

32 let a;
33 let b;
34 let n = -1;
35 let str = prompt('Введите первый член прогрессии.', '');
36 a = Number(str);
37 str = prompt('Введите знаменатель прогрессии.', '');
38 b = Number(str);
39 while (n <= 0) {
40     str = prompt('Введите номер члена прогрессии. Число должно быть больше нуля
41     !', '');
42     n = Number(str);
43 }
44 let progress = new GeometricProgression(a, b);
45
46 console.log(progress.member(n));
47 try {
48     console.log(progress.summa(n));
49     console.log(progress.infinitySumma());
50 } catch (error) {
51     if (error.name == "Error") {
52         console.log(error.message);
53     } else {
54         throw error;
55     }
56 }
```

Помимо функций вычисления определённого члена прогрессии и её суммы, здесь также присутствует функция, вычисляющая сумму бесконечно убывающей прогрессии. Для этого она проверяет, является ли знаменатель прогрессии меньше единицы, и не является ли он меньше нуля, ведь в таком случае прогрессия является знакочередующейся, и следовательно её сумму уже нельзя найти.

Во второй части пользователь также вводит переменные для прогрессии, и происходит вывод значений на экран.

Задачи, сгенерированные по шаблонам, использующим код геометрической прогрессии.

Бизнесмен Плюшкин получил в 2000 году прибыль в размере 1100000 рублей. Каждый следующий год его прибыль увеличивалась на 20% по сравнению с предыдущим годом. Сколько рублей заработал Плюшкин за 2003 год?

Компания «Альфа» начала инвестировать средства в перспективную отрасль в 2001 году, имея капитал в размере 3500 долларов. Каждый год, начиная с 2002 года, она получала прибыль, которая составляла 100% от капитала предыдущего года. А компания «Бета» начала инвестировать средства в другую отрасль в 2007 году имея капитал в размере 4500 долларов, и, начиная с 2008 года, ежегодно получала прибыль, составляющую 200% от капитала предыдущего года. На сколько долларов капитал одной из компаний был больше капитала другой к концу 2009 года, если прибыль из оборота не изымалась?

3.5. Наибольшее и наименьшее значение функции

Код задачи №26695

```
1 function() {
2     retryWhileError(function() {
3         'use strict';
4         let a = sl(1, 100);
5         let b = sl(1, 100);
6         let c = sl(1, 100);
7         let arr1 = [ '+', '-' ];
8         let arr2 = [ 'sin', 'cos' ];
9         let maxmin1 = sl1();
10        let maxmin2 = (maxmin1 + 1) % 2;
11        NATask.setMinimaxFunctionTask({
12            expr: arr1[maxmin1] + a + arr2.iz() + '(x)' + arr1.iz() + b + 'x' +
13            arr1.iz() + c,
14            leftEnd: [ '-pi/2', '-3pi/2', '0' ].iz(),
15            rightEnd: [ 'pi/2', '3pi/2', '0' ].iz(),
16            primaryStep: 0.1,
17            secondaryStep: 0.0001,
18            authors: [ 'Алэндарь Сергей' ],
19        });
20    }, 10000);
21 }();  
//26695
```

Одной из особенностей этого шаблона является, использующаяся в нём библиотека `setMinimaxFunctionTask`, которая берёт на себя большую часть работы по подбору значений, а также отображению задачи на экране. Но всё же трудности могут возникнуть при подборе значений для `primaryStep`: и `secondaryStep`: , отвечающих за первичный и вторичный перебор значений максимума или минимума соответственно. Сначала поиск проходит первичным шагом, затем в окрестности предполагаемого ответа, происходит поиск вторичным шагом. Чем меньше задать шаги поиска, тем больше будет нагрузка и

дольше программа будет подбирать значение, а если сделать наоборот, и задать шаги большего размера, то увеличится скорость поиска, но пострадает точность решения.

Задания сгенерированные по шаблону задачи №26695.

Найдите наименьшее значение функции $y = -15 \sin x + 90x + 5$ на полуинтервале $\left[0; \frac{3\pi}{2}\right)$.

Производная функции: $y' = 90 - 15 \cos x$

5

Найдите наименьшее значение функции $y = -20 \cos x + 59 + 93x$ на отрезке $\left[0; \frac{3\pi}{2}\right]$.

Производная функции: $y' = 93 - 20 \sin x$

39

3.6. Нахождение точек экстремума функции

Тригонометрическая задача №77492

Найдите точку максимума функции $y = (2x - 3) \cos x - 2 \sin x + 5$, принадлежащую промежутку $\left(0; \frac{\pi}{2}\right)$.

[Спрятать решение](#)

Решение.

Найдем производную заданной функции:

$$y' = 2 \cos x - (2x - 3) \sin x - 2 \cos x = (3 - 2x) \sin x.$$

На заданном промежутке (первая четверть без граничных точек) синус не обращается в нуль и принимает только положительные значения. Поэтому единственный нуль производной — число 1,5.

Определим знаки производной функции: она положительна при $x < 1,5$ и отрицательна при $x > 1,5$. Поэтому искомая точка максимума — число 1,5.

Код для задачи №77492

```
1 function() {
2   'use strict';
3
4   var arr1 = ['максимума', 'минимума'];
5   var arr2 = ['положительный', 'отрицательный'];
6   var maxmin1 = sl(0, 1);
7   var maxmin2 = (maxmin1 + 1) % 2;
8
9   var ans = sl(2, 14, 2) / 10;
10  var b = sl(5, 50, 5);
11  var a = ans * b;
12  var d = sluchch(-99, 99);
13
14  if (maxmin1 == 1) {
15    b = -b;
16    a = -a;
17  }
18
19  function plusmin(member) {
20    var el = math.parse(member);
21    el = math.simplify(el);
22    el = math.simplify(el, [
23      l: 'n1*n2',
24      r: 'n1 n2'
```

```

25     });
26
27     return el.toTex({
28       implicit: 'hide'
29     });
30
31   chas2.task.setTask({
32     text: 'Найдите точку ' + arr1[maxmin1] + ' функции '
33     '$$ y=( ' + plusmin(b + 'x-' + a) + ')^{\cos} + plusmin('x-' + b) +
34     '\sin ' +
35     'plusmin('x+' + d) + ', $$' +
36     'принадлежащую промежутку $(0; \frac{\pi}{2})$.',
37     analys: '$$ y^{\prime}=( ' + plusmin(b + 'x-' + a) + ')^{\cos} \cos x + (' +
38     plusmin(b + 'x-' + a) + ')^{\cos} \cos x + ' +
39     '(\cos x)^{\prime} + (' + -b + '\sin x)^{\prime}$$' +
40     '$$ y^{\prime}= ' + b + '\cos x - (' + plusmin(b + 'x-' + a) + ')^{\cos} \sin x + ' +
41     'plusmin('x-' + b) + '\cos x' +
42     '\Rightarrow ' +
43     'y^{\prime}=-(' + plusmin(b + 'x-' + a) + ') \sin x$$' +
44     '$$ y^{\prime}=0' + ' \Rightarrow (' + plusmin(b + 'x-' + a) + ') \sin x=0$$' +
45     '$$1) \quad plusmin(b + 'x-' + a) = 0' + ' \Rightarrow ' +
46     'b + x = a + ' + '\Rightarrow x = a + ans' + '$$' +
47     '$$2) \quad \sin x = 0' + ' \Rightarrow ' +
48     'x = \pi n, \quad \text{Промежутку } (0; \frac{\pi}{2}) \text{ принадлежит только точка } $' + ans +
49     '$. Подставляя любые значения из промежутка $(0; ' + ans +
50     ')$ в найденную производную, мы узнаём, что она имеет ' + arr2[maxmin1] +
51     ' знак, а на промежутке $( ' + ans + ' ; \frac{\pi}{2})$ ' +
52     'Производная функции меняет знак с ' +
53     'sklonlxkand(arr2[maxmin1]).re + ' на ' + arr2[maxmin2] + ', ' +
54     'значит, ' +
55     '$ - это искомая точка ' + arr1[maxmin1] + '.',
56
57   answers: ans,
58 });
59 })();

```

В коде этой задачи не использовалась библиотека `setMinimaxFunctionTask`, потому что здесь стояла другая цель. Здесь нужно найти точку максимума (минимума) функции, а не её максимальное (минимальное) значение.

Также здесь присутствуют подробное пояснениеющее решение, которое нельзя было получить при использовании библиотеки из прошлой задачи.

Решения тригонометрической задачи, сгенерированные шаблоном

Найдите точку минимума функции

$$y = (6 - 30x) \cos x + 30 \sin x - 84,$$

принадлежащую промежутку $(0; \frac{\pi}{2})$.

$$\begin{aligned}y' &= (6 - 30x)' \cos x + (6 - 30x)(\cos x)' + (30 \sin x)' \\y' &= -30 \cos x - (6 - 30x) \sin x + 30 \cos x \Rightarrow y' = -(6 - 30x) \sin x \\y' &= 0 \Rightarrow (6 - 30x) \sin x = 0 \\1) \quad 6 - 30x &= 0 \Rightarrow -30x = -6 \Rightarrow x = 0.2 \\2) \quad \sin x &= 0 \Rightarrow x = \pi n, \quad n \in \mathbb{Z}\end{aligned}$$

Промежутку $(0; \frac{\pi}{2})$ принадлежит только точка 0.2. Подставляя любые значения из промежутка $(0; 0.2)$ в найденную производную, мы узнаём, что она имеет отрицательный знак, а на промежутке $(0.2; \frac{\pi}{2})$ положительный. Производная функции меняет знак с отрицательного на положительный, значит, 0.2 - это искомая точка минимума.

Найдите точку максимума функции

$$y = (15x - 15) \cos x - 15 \sin x + 55,$$

принадлежащую промежутку $(0; \frac{\pi}{2})$.

$$\begin{aligned}y' &= (15x - 15)' \cos x + (15x - 15)(\cos x)' + (-15 \sin x)' \\y' &= 15 \cos x - (15x - 15) \sin x - 15 \cos x \Rightarrow y' = -(15x - 15) \sin x \\y' &= 0 \Rightarrow (15x - 15) \sin x = 0 \\1) \quad 15x - 15 &= 0 \Rightarrow 15x = 15 \Rightarrow x = 1 \\2) \quad \sin x &= 0 \Rightarrow x = \pi n, \quad n \in \mathbb{Z}\end{aligned}$$

Промежутку $(0; \frac{\pi}{2})$ принадлежит только точка 1. Подставляя любые значения из промежутка $(0; 1)$ в найденную производную, мы узнаём, что она имеет положительный знак, а на промежутке $(1; \frac{\pi}{2})$ отрицательный. Производная функции меняет знак с положительного на отрицательный, значит, 1 - это искомая точка максимума.

Заключение

За всё время работы было разработано следующее количество шаблонов:

Тип №10 ЕГЭ профильного уровня на арифметические прогрессии: 9

Тип №10 ЕГЭ профильного уровня на геометрические прогрессии: 2

Тип №6 ЕГЭ профильного уровня на логарифмические уравнение: 3

Тип №8 ЕГЭ профильного уровня на физический смысл производной,
нахождение скорости процесса: 1

Тип №12 ЕГЭ профильного уровня на наибольшее и наименьшее
значение функции: 12

Тип №16 ЕГЭ базового уровня на вычисления и преобразования: 23

Также были подготовлены тезисы, которые были отправлены для заочного
участия в 74 МСНТК.

Данная работа помогла оказать вклад в развитии образовательного про-
екта «Час ЕГЭ», а разработанные шаблоны, будут способствовать в подготовке
школьников к тестовой части ЕГЭ по математике.

Помимо этого, сама дипломная работа будет служить руководством для
будущих студентов, которые будут также принимать участие в разработке шаб-
лонов и развитии проекта "Час ЕГЭ".

Список использованных источников

Список литературы

- [1] VirtualBox. Виртуальные диски. Их типы. Расширение виртуального носителя // Хекслет Рус : [сайт]. - 2021. - URL: <https://ru.hexlet.io/blog/posts/virtualbox-virtualnye-diski-ih-tipy-rasshirenie-virtualnogo-nositelya> (Дата обращения 22.03.2024)
2. Колесников Е. Н. Что такое GitHub и как им пользоваться / Е. Н. Колесников // SkillBox Media : [сайт]. - 2024. - URL:<https://skillbox.ru/media/code/cto-takoe-github-i-kak-im-polzovatsya/> (Дата обращения: 02.04.2024)
3. Львовский С. М. «Набор и верстка в системе LaTeX» - URL: <https://old.mccme.ru//free-books//llang/newllang.pdf> (дата обращения: 16.04.2024).
4. Образовательный портал «РЕШУ ЕГЭ» - URL: <https://ege.sdamgia.ru>
5. Открытый банк задач ЕГЭ по Математике. Профильный уровень. – URL: <https://prof.mathege.ru>
6. Полноценная платформа для разработчиков для создания, масштабирования и доставки защищенного программного обеспечения GitHub – URL: <https://github.com/nickkolok/chas-ege/> (дата обращения: 20.04.2024).
7. Полный интерактивный тест - Тренажёр «Час ЕГЭ» - URL:<https://math.vsu.ru/chas-ege/sh/polnmat.html> (дата обращения: 18.04.2024).

Приложение

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = s1(2, 13);
5     let b = s1(2, 6);
6     let c = s1(2, 10);
7     NATask.setEvaluationTask({
8       expr: a * c + '/` + `(` + b + `sqrt(` + a + `)` + `)` + `^2`,
9       authors: ['Алэндарь Сергей'],
10      });
11    }, 10000);
12  })();
13 //514504
```

Найдите значение выражения:

$$\frac{63}{(5\sqrt{7})^2}$$

Найдите значение выражения:

$$\frac{56}{(5\sqrt{8})^2}$$

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = s1(2, 6);
5     let b = s1(3, 10);
6     let c = a * b * b;
7     let d=1,e=1;
8     if (s11()) {
9       d = s1(2, 15);
```

```

10     } else {
11         e = sl(2, 15);
12     }
13     NATask.setEvaluationTask({
14         expr: d + 'sqrt(' + c + ')' + '/' + e + 'sqrt(' + a + ')',
15         authors: ['Алэндарь Сергей'],
16     });
17 }, 10000);
18 })();
19 //509770

```

Найдите значение выражения:

$$\frac{11\sqrt{500}}{\sqrt{5}}$$

Найдите значение выражения:

$$\frac{\sqrt{100}}{8\sqrt{4}}$$

```

1 (function() {
2     retryWhileError(function() {
3         'use strict';
4         let a = sl(2, 7);
5         let b = sl(2, 5);
6         let c = a * b * b - sl(1, 7).pm();
7         NATask.setEvaluationTask({
8             expr: ['(' + [b + 'sqrt(' + a + ')', 'sqrt(' + c +
9                 ')'].shuffle().join('+') + ')',
10                '(' + [b + 'sqrt(' + a + ')', 'sqrt(' + c + ')'].shuffle().join('-') +
11                ')'].shuffle().join('*'),
12                authors: ['Алэндарь Сергей'],
13            });
14 }, 10000);
15 })();

```

```
14 // 506880
```

Найдите значение выражения:

$$(4\sqrt{5} + \sqrt{83}) \cdot (4\sqrt{5} - \sqrt{83})$$

Найдите значение выражения:

$$(\sqrt{127} + 5\sqrt{5}) \cdot (\sqrt{127} - 5\sqrt{5})$$

```
1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = s1(2, 19);
5     let b = s1(1, 2);
6     let c;
7     let d = s1(2, 5);
8     if (a <= 9) {
9       c = s1(1, 2);
10    } else {
11      c = 0;
12    }
13    NATask.setEvaluationTask({
14      expr: ['varlog(sqrt(' + Math.pow(a, b) + ') , ' + Math.pow(a, b + c) +
15      ') , ' + varlog(sqrt(' + a + '), ' + a + '^' + +
16      d + ') ,
17      ].iz(),
18      authors: ['Алендарь Сергей'],
19    });
20  }, 10000);
21 })();
22 //527441
```

Найдите значение выражения:

$$\log_{\sqrt{4}} 4^3$$

Найдите значение выражения:

$$\log_{\sqrt{5}} 25$$

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = s1(2, 7);
5     let b = s1(2, 6);
6     let d = s1(9, 10) - a;
7
8     NATask.setEvaluationTask({
9       expr: '(' + b + ')^((' + d + 'varlog(' + b + ',' + a + ')))',
10      authors: ['Алехандро Сергеев'],
11    });
12  }, 10000);
13})();
14 //506277
```

Найдите значение выражения:

$$5^6 \log_5 4$$

Найдите значение выражения:

$$4^7 \log_4 3$$

```
1 (function() {
2   retryWithError(function() {
```

```

3   'use strict';
4   let a = sl(2, 8);
5   let b = slKrome(a, 2, 8)
6   NATask.setEvaluationTask({
7     expr: 'sqrt(' + a + '^' + sl(2, 6, 2) + '*' + b + '^' + sl(2, 6, 2) +
8   ')',
9     authors: ['Алендарь Сергей'],
10    });
11   }, 10000);
12 })();
13 //509212

```

Найдите значение выражения:

$$\sqrt{4^4 \cdot 6^4}$$

Найдите значение выражения:

$$\sqrt{8^6 \cdot 5^2}$$

```

1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = sl(2, 10);
5     let b = sl(3, 9);
6     let c = a * b * b;
7     NATask.setEvaluationTask({
8       expr: slKrome(b, 1, 5).pm() + '/' + b + ['sqrt(' + a + ')', 'sqrt(' + c +
9     + ')'].shuffle().join('*'),
10      authors: ['Алендарь Сергей'],
11    });
12   }, 10000);
13 })();
14 // 506508

```

```

1 (function() {

```

Найдите значение выражения:

$$\frac{-4}{3} \cdot \sqrt{3} \cdot \sqrt{27}$$

Найдите значение выражения:

$$\frac{5}{6} \cdot \sqrt{144} \cdot \sqrt{4}$$

```
2  retryWithError(function() {
3      'use strict';
4      let a = sl(1.1, 1.9, 0.1);
5      let b = sl(2, 9);
6      let c = a * b * b;
7      NAtask.setEvaluationTask({
8          expr: 'divideColon(' + (a * b) + '*10^' + sl(1, 3) + ', ' + a + '*10^',
9          + sl(-3, -1) + ')',
10         authors: ['Алендарь Сергей'],
11     });
12 }, 10000);
13 })();
14 //1395
```

Найдите значение выражения:

$$(7,8 \cdot 10^2) : (1,3 \cdot 10^{-2})$$

Найдите значение выражения:

$$(14,4 \cdot 10^3) : (1,6 \cdot 10^{-1})$$

```
1 (function() {
2     retryWithError(function() {
3         'use strict';
```

```

4 let a = sl(1.1, 9.9, 0.1);
5 let b = sl(2, 9);
6 let znak = (1).pm();
7 NATask.setEvaluationTask({
8     expr: ['(' + b + '*10^' + znak * sl(1, 9) + ')',
9     '(' + a + '*10^' + znak * sl(-9, -1) + ')'].shuffle().join('*'),
10    authors: ['Алэндарь Сергей'],
11 });
12 }, 10000);
13 })();
14 //510193

```

Найдите значение выражения:

$$9,2 \cdot 10^{-5} \cdot 3 \cdot 10^7$$

Найдите значение выражения:

$$5,4 \cdot 10^{-7} \cdot 3 \cdot 10^6$$

```

1 (function() {
2     retryWhileError(function() {
3         'use strict';
4         let a = sl(2, 9, 0.1);
5         let b = sl(2, 9, 0.1);
6         let c = sl(2, 9, 0.1);
7         NATask.setEvaluationTask({
8             expr: ['(' + a + '*10^' + sl(1, 5).pm() + ')',
9             '(' + b + '*10^' + sl(1, 5).pm() + ')', '(' + c + '*10^' + sl(1, 5).pm()
+ ')'].iz(sl(2,3)).join('+'),
10            authors: ['Алэндарь Сергей'],
11        });
12     }, 10000);
13 })();
14 //509209

```

```

1 (function() {

```

Найдите значение выражения:

$$6,9 \cdot 10^5 + 4,2 \cdot 10^2 + 2,3 \cdot 10^1$$

Найдите значение выражения:

$$8,4 \cdot 10^{-2} + 3,6 \cdot 10^{-2}$$

```
2  retryWhileError(function() {  
3      'use strict';  
4      NAtask.setEvaluationTask({  
5          expr: '(-10)^' + sl(0, 5) + '+(-10)^' + sl(0, 5) + '+(-10)^' + sl(0, 5),  
6          authors: ['Алэндарь Сергей'],  
7      });  
8  }, 10000);  
9 })();  
10 //511746
```

Найдите значение выражения:

$$(-10)^2 + (-10)^3 + (-10)^3$$

Найдите значение выражения:

$$(-10)^2 + (-10)^3 + (-10)^1$$

```
1 (function() {  
2     retryWhileError(function() {  
3         'use strict';  
4         let c = sl(2, 20);  
5         let znak = (1).pm();  
6         NAtask.setEvaluationTask({  
7             expr: ['(' + c + '^' + znak * sl(1, 10) + '*' + c + '^' + (-znak) *  
8             sl(1, 10) +  
9             ')', c + '^' + sl(1, 10).pm()].shuffle().join('/'),  
10            authors: ['Алэндарь Сергей'],  
11        });  
12    });  
13 })();  
14 //511746
```

```
10     });
11 }, 10000);
12 })();
13 //509607
```

Найдите значение выражения:

$$\frac{11^6}{11^{-7} \cdot 11^9}$$

Найдите значение выражения:

$$\frac{16^{-8} \cdot 16^2}{16^{-6}}$$

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = sl(2, 9);
5     let b = sl(1, 9).pm();
6     let c = slKrome([b, 0, 1, -1], -9, 9);
7     let d = slKrome([b, c, 0, 1, -1], -9, 9);
8     NAtask.setEvaluationTask({
9       expr: [a + '^' + b, '(' + a + '^' + c + '/' + a + '^' + d +
')'].shuffle().join('*'),
10      authors: ['Алендарь Сергей'],
11    });
12 }, 10000);
13 })();
14 //508403
```

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = sl(1.1, 9.9, 0.1);
5     let b = slKrome([a], 1.1, 9.9, 0.1)
6     NAtask.setEvaluationTask({
```

Найдите значение выражения:

$$5^{-7} \cdot \frac{5^6}{5^{-8}}$$

Найдите значение выражения:

$$\frac{4^7}{4^{-5}} \cdot 4^{-4}$$

```
7     expr: [‘(‘ + a + ‘*10^2)’, ‘(‘ + b + ‘*10^3)’].shuffle().join(‘+’),
8     authors: [‘Алендарь Сергей’],
9   );
10  }, 10000);
11 })();
12 //511955
```

Найдите значение выражения:

$$4,5 \cdot 10^2 + 1,5 \cdot 10^3$$

Найдите значение выражения:

$$3,4 \cdot 10^2 + 3,3 \cdot 10^3$$

```
1 (function() {
2   retryWhileError(function() {
3     ‘use strict’;
4     let a = s1(10, 99);
5     let b = s1(1.1, 9.9, 0.1);
6     NAtask.setEvaluationTask({
7       expr: a + ‘*10-’ + b + ‘*10^2’,
8       authors: [‘Алендарь Сергей’],
9     });
10  });
11 })();
12 //511955
```

```
10 }, 10000);  
11 })();  
12 //510955
```

Найдите значение выражения:

$$87 \cdot 10 - 3,4 \cdot 10^2$$

Найдите значение выражения:

$$77 \cdot 10 - 4,1 \cdot 10^2$$

```
1 (function() {  
2   retryWithError(function() {  
3     'use strict';  
4     let a = sl(2, 9);  
5     let b = a * sl(2, 9) * 0.1;  
6     NAtask.setEvaluationTask({  
7       expr: '(' + b + '*10^' + sl(1, 5) + ') / (' + a + '*10^' + sl(-5, -1) +  
8       ')',  
9       authors: ['Алехандро Сергеев'],  
10      });  
11    }, 10000);  
12 })();  
13 //10000801
```

Найдите значение выражения:

$$\frac{4,8 \cdot 10^3}{6 \cdot 10^{-2}}$$

Найдите значение выражения:

$$\frac{2,4 \cdot 10^2}{3 \cdot 10^{-1}}$$

```

1 (function() {
2   retryWithError(function() {
3     'use strict';
4     NAtask.setEvaluationTask({
5       expr: 'forceBrackets(0.1)^' + sl(2, 6) + '*10^' + sl(2, 6) + '*' + sl(2,
6         9) + '^' + sl(2, 6),
7       authors: ['Алендарь Сергей'],
8     });
9   }, 10000);
10 })();
//10000001

```

Найдите значение выражения:

$$(0,1)^6 \cdot 10^6 \cdot 2^2$$

Найдите значение выражения:

$$(0,1)^5 \cdot 10^6 \cdot 8^2$$

```

1 (function() {
2   retryWithError(function() {
3     'use strict';
4     NAtask.setEvaluationTask({
5       expr: [
6         '(' + [0.01, 0.1].iz() + '^' + sl(2, 5) + '/' + '10^' + sl(-1, -5) +
7         ')',
8         '10^' + sl(2, 6)].shuffle().join('*'),
9       authors: ['Алендарь Сергей'],
10      });
11    }, 10000);
12  })();
//512907

```

```

1 (function() {
2   retryWithError(function() {
3     'use strict';

```

Найдите значение выражения:

$$\frac{0,01^4}{10^{-5}} \cdot 10^5$$

Найдите значение выражения:

$$10^3 \cdot \frac{0,01^3}{10^{-1}}$$

```
4 NAtask.setEvaluationTask({
5     expr: sl(2, 99) + '*(-1)^' + sl(0, 9).pm() + '+' + sl(2, 99) + '*(-1)^',
6     + sl(0, 9).pm(),
7     authors: ['Алендарь Сергей'],
8     });
9 }, 10000);
10 })();
11 //512214
```

Найдите значение выражения:

$$79 \cdot (-1)^{-1} + 21 \cdot (-1)^{-4}$$

Найдите значение выражения:

$$18 \cdot (-1)^{-7} + 16 \cdot (-1)^6$$

```
1 (function() {
2     retryWhileError(function() {
3         'use strict';
4         let c = sl(2, 9);
5         NAtask.setEvaluationTask({
6             expr: ['(' + c + '^' + sl(1, 9).pm() + ')', + '^', + sl(2, 5).pm(), c +
7             '^', + sl(-15, -2)].shuffle().join('/'),
8             authors: ['Алендарь Сергей'],
9         });
10 })();
11 //512214
```

```
8     });
9   }, 10000);
10 })();
11 //508353
```

Найдите значение выражения:

$$\frac{7^{-6}}{(7^3)^{-4}}$$

Найдите значение выражения:

$$\frac{3^{-6}}{(3^{-9})^2}$$

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = sl(2, 9);
5     let b = slKrome([a], 2, 9);
6     let c = slKrome([a, b], 2, 9);
7     NAtask.setEvaluationTask({
8       expr: a + '*' + c + '^' + sl(2, 5) + '+' + b + '*' + c + '^' + sl(2, 5),
9       authors: ['Алендарь Сергей'],
10      });
11    }, 10000);
12  })();
13 //509647
```

```
1 (function() {
2   retryWithError(function() {
3     'use strict';
4     let a = sl(1, 30, 0.5).pm();
5     let b = sl(1, 30, 0.5);
6     let c = sl(1, 15);
7     let d = c + sl(3, 30);
```

Найдите значение выражения:

$$4 \cdot 8^2 + 5 \cdot 8^3$$

Найдите значение выражения:

$$9 \cdot 4^2 + 3 \cdot 4^5$$

```
8 let arr1 = [ '+', '-' ];
9 let arr2 = [ '-', '+' ];
10 let maxmin = s11();
11 NAtask.setMinimaxFunctionTask({
12     expr: '' + a + arr1[maxmin] + b + ' x ' + arr2[maxmin] + [ ' x sqrt(x)
13 , ' x^(3/2) ].iz(),
14     leftEnd: '' + c,
15     rightEnd: '' + d,
16     primaryStep: 1,
17     secondaryStep: 0.001,
18     authors: [ 'Алеко Симонов' ],
19 });
20 }, 10000);
21 })();
22 //77460
```

Найдите наибольшее значение функции $y = x\sqrt{x} + -4 - 27,5x$ на полуинтервале $[3; 28]$.

Производная функции: $y' = \sqrt{x} + x \cdot \frac{1}{2\sqrt{x}} + 0 - 27.5$

Найдите наименьшее значение функции $y = -21,5 - 7,5x + x\sqrt{x}$ на полуинтервале $[12; 16]$.

Производная функции: $y' = 0 - 7.5 + \sqrt{x} + x \cdot \frac{1}{2\sqrt{x}}$

```
1 (function() {
2     retryWhileError(function() {
```

```

3   'use strict';
4
5   let a = sl(1, 9);
6   let b = sl(1, 50);
7   let c = sl(1, 100);
8   let d = sl(1, 100);
9   let arr1 = [ '+', '-' ];
10
11  NATask.setMinimaxFunctionTask({
12      expr: a + 'x^5' + arr1[0] + b + 'x^3' + [arr1[1] + c + 'x',
13      arr1[0] + d].join(),
14      leftEnd: '- ' + sl(4, 20),
15      rightEnd: ' ' + sl(0, 2).random(),
16      primaryStep: 1,
17      secondaryStep: 0.001,
18      authors: ['Алендарь Серге́й'],
19  });
20 }, 10000);
21 })();
22 //315835

```

Найдите наибольшее значение функции $y = 3x^5 + 40x^3 + 82$ на полуинтервале $(-11; 1]$.

Производная функции: $y' = 3 \cdot 5 \cdot x^{5-1} + 40 \cdot 3 \cdot x^{3-1}$

Найдите наибольшее значение функции $y = 8x^5 + 31x^3 + 43$ на отрезке $[-8; -1]$.

Производная функции: $y' = 8 \cdot 5 \cdot x^{5-1} + 31 \cdot 3 \cdot x^{3-1}$

```

1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = sl(2, 20, 2);
5     let b = sl(1, 50);
6     let l = sl(-5, 0);
7     let r = l + sl(2, 5);

```

```

8  let arr1 = [ '+', '-' ];
9  NAtask.setMinimaxFunctionTask({
10    expr: 'e^(2x) - ' + a + 'e^x' + arr1.iz() + b,
11    leftEnd: '' + l,
12    rightEnd: '' + r,
13    primaryStep: 1,
14    authors: ['Алэндарь Сергей'],
15  });
16 }, 10000);
17 })();
18 //315127

```

Найдите наименьшее значение функции $y = 37 + e^{2x} - 2e^x$ на отрезке $[-1; 2]$.

Производная функции: $y' = e^{2x} \cdot \ln(e) \cdot 2 - 2 \cdot e^x \cdot \ln(e)$

Найдите наименьшее значение функции $y = 3 + e^{2x} - 2e^x$ на полуинтервале $(-4; -2]$.

Производная функции: $y' = e^{2x} \cdot \ln(e) \cdot 2 - 2 \cdot e^x \cdot \ln(e)$

```

1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = s1(2, 20, 2);
5     let b = s1(1, 100);
6     let c = s1(1, 50);
7     let arr1 = [ '+', '-' ];
8     let arr2 = [ 'sin', 'cos' ];
9     let maxmin1 = s11();
10    let maxmin2 = (maxmin1 + 1) % 2;
11    NAtask.setMinimaxFunctionTask({
12      expr: a + arr2.iz() + '(x)' + arr1.iz() + b + 'x/pi' + arr1.iz() + c,
13      leftEnd: ['-2pi/3', '-5pi/6'].iz(),
14      rightEnd: '0',

```

```

15     primaryStep: 0.1,
16     secondaryStep: 0.001,
17     authors: [ 'Алендарь Сергей' ],
18   });
19 }, 10000);
20 })();
21 //26700

```

Найдите наименьшее значение функции $y = 31 + 14 \cos x - \frac{45x}{\pi}$ на полуинтервале $\left(\frac{-5\pi}{6}; 0\right]$.

Производная функции: $y' = -\sin x (14) - \frac{45}{\pi}$

Найдите наибольшее значение функции $y = 25 + \frac{4x}{\pi} + 18 \sin x$ на отрезке $\left[\frac{-2\pi}{3}; 0\right]$.

Производная функции: $y' = \frac{4}{\pi} + 18 \cdot \cos x$

--

```

1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = s1(3, 50);
5     let b = s1(1, 100);
6     let c = s1(1, 50);
7     let arr1 = [ '+', '-' ];
8     let maxmin1 = s11();
9     let maxmin2 = (maxmin1 + 1) % 2;
10    NAtask.setMinimaxFunctionTask({
11      expr: arr1[maxmin1] + a + 'tan(x)' + arr1[maxmin2] + a + 'x' +
12        [ ', arr1.iz() + c + 'pi' ].iz() + arr1.iz() + b,
13      leftEnd: [ '0', '-pi/4' ].iz(),
14      rightEnd: [ '0', 'pi/4' ].iz(),
15      primaryStep: 0.00001,
16      secondaryStep: 0.001,
17      authors: [ 'Алендарь Сергей' ],
18    });
19  }, 10000);
20 })();
21 //26700

```

```

15     secondaryStep: 0.000001,
16     authors: [’Алендарь Серге́й’],
17   );
18 }, 1000);
19 })();
20 //26704

```

Найдите наибольшее значение функции $y = -21 \tan(x) + 21x + 84$ на отрезке $\left[0; \frac{\pi}{4}\right]$.

Производная функции: $y' = 21 - 21 \sec(x)^2$

Найдите наименьшее значение функции $y = 37 + 26 \tan(x) - 26x$ на полуинтервале $\left[0; \frac{\pi}{4}\right)$.

Производная функции: $y' = 26 \cdot \sec(x)^2 - 26$

```

1 (function() {
2   retryWhileError(function() {
3     ’use strict’;
4     let a = sl(1, 9, 0.5);
5     let b = sl(1, 100);
6     let arr1 = [’+’, ’-’];
7     let arr2 = [’ ’, ’-’];
8     let maxmin1 = sl(0, 1);
9     let maxmin2 = (maxmin1 + 1) % 2;
10    NATask.setMinimaxFunctionTask({
11      expr: ’ ’ + arr2[maxmin1] + (a * 2) + ’tan(x)’ + arr1[maxmin2] + (a * 4)
12      + ’x’ + arr1[maxmin1] + a + ’pi’ + arr1[maxmin2] + b,
13      leftEnd: ’-pi/3’,
14      rightEnd: ’pi/3’,
15      primaryStep: 0.001,
16      authors: [’Алендарь Серге́й’],
17    });

```

```
17 } , 10);  
18 })();  
19 //26704
```

Найдите наименьшее значение функции $y = 8\pi + 16 \tan(x) - 32x - 60$ на интервале $\left(-\frac{\pi}{3}; \frac{\pi}{3}\right)$.

Производная функции: $y' = 16 \cdot \sec(x)^2 - 32$

Найдите наибольшее значение функции $y = -8 \tan(x) + 16x - 4\pi + 100$ на полуинтервале $\left[-\frac{\pi}{3}; \frac{\pi}{3} \right)$.

Производная функции: $y' = 16 - 8 \sec(x)^2$

```
1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = sl(1, 3);
5     let b = sl(1, 50);
6     let c = sl(1, 15);
7     let d = sl(1, 15);
8     let e = c + sl(3, 30);
9     let arr1 = ['+', '-'];
10    let arr2 = ['-', '+'];
11    let arr3 = ['-', ''];
12    let maxmin = sl1();
13    NATask.setMinimaxFunctionTask({
14      expr: '' + '(' + a + 'x^2' + arr1[maxmin] + b + 'x' + arr2[maxmin] + b +
15      ')', +
16      'e^( ' + arr3[maxmin] + 'x' + [arr1[maxmin] + c, ''].iz() + ')',
17      leftEnd: '' + d,
18      rightEnd: '' + e,
19      primaryStep: 1,
20      secondaryStep: 0.01,
21      authors: ['Алендарь Серге́й'],
```

```

21     });
22 }, 1000);
23 })();
24 //77480

```

Найдите наименьшее значение функции $y = (18x + 2x^2 - 18)e^{-x}$ на отрезке $[6; 32]$.

Производная функции: $y' = (18 + 2 \cdot 2 \cdot x^{2-1}) \cdot e^{-x} - (18x + 2x^2 - 18) \cdot e^{-x} \cdot \ln(e)$

Найдите наименьшее значение функции $y = (1x^2 + 30x - 30)e^{-x}$ на полуинтервале $[2; 29]$.

Производная функции: $y' = (2 \cdot x^{2-1} + 30) \cdot e^{-x} - (x^2 + 30x - 30) \cdot e^{-x} \cdot \ln(e)$

```

1 (function() {
2   retryWhileError(function() {
3     'use strict';
4     let a = sl(1, 50);
5     let b = sl(1, 300);
6     let c = sl(1, 50);
7     let arr1 = [', ', '-'];
8     let arr2 = ['+', '-'];
9     NATask.setMinimaxFunctionTask({
10       expr: [arr1.iz() + 'x^3', arr1.iz() + a + 'x^2', arr1.iz() + b +
11         'x'].iz(sl(2, 3)).join('+') + arr2.iz() + c,
12       leftEnd: '-' + sl(1, 9, 0.5),
13       rightEnd: '' + sl(1, 9, 0.5),
14       primaryStep: 1,
15       secondaryStep: 0.001,
16       authors: ['Алендарь Сергей'],
17     });
18   }, 10);
19 })();
20 //77438

```

Найдите наименьшее значение функции $y = -x^3 + 49x^2 - 28$ на полуинтервале $[-1.5; 6)$.

Производная функции: $y' = 49 \cdot 2 \cdot x^{2-1} - 3 \cdot x^{3-1}$

Найдите наибольшее значение функции $y = -28x + 26x^2 + 40$ на интервале $(-5; 5)$.

Производная функции: $y' = 26 \cdot 2 \cdot x^{2-1} - 28$

```
1 class ArithmeticProgression {
2     constructor(first, difference) {
3         this.first = first;
4         this.difference = difference;
5         this.nmember = 0;
6     }
7
8     member(nmember) {
9         this.nmember = nmember;
10        return this.first + this.difference * (this.nmember - 1);
11    }
12
13    summa(nmember) {
14        this.nmember = nmember;
15        return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16        this.nmember;
17    }
18
19 let a = sluchch(8, 30);
20 let b = sluchch(2, 4);
21 let n = sluchch(4, 11);
22 let progression = new ArithmeticProgression(a, b);
23 let q = a + progression.member(n);
24
25 NAtask.setTask({
26     text: 'Бригада маляров красит забор длиной ' +
```

```

1      chislitlx(progression.summa(n), 'метр') +
2      ', ежедневно увеличивая норму покраски на одно и то же число метров.' +
3      ' Известно, что за первый и последний день в сумме бригада покрасила ' +
4      chislitlx(q, 'метр') + ' забора. ' +
5      ' Определите, сколько дней бригада маляров красила весь забор. ',
6      answers: n,
7  );
8 //99579

```

Бригада маляров красит забор длиной 70 метров , ежедневно увеличивая норму покраски на одно и то же число метров. Известно, что за первый и последний день в сумме бригада покрасила 28 метров забора. Определите, сколько дней бригада маляров красила весь забор.

Бригада маляров красит забор длиной 360 метров , ежедневно увеличивая норму покраски на одно и то же число метров. Известно, что за первый и последний день в сумме бригада покрасила 80 метров забора. Определите, сколько дней бригада маляров красила весь забор.

```

1 class ArithmeticProgression {
2     constructor(first, difference) {
3         this.first = first;
4         this.difference = difference;
5         this.nmember = 0;
6     }
7
8     member(nmember) {
9         this.nmember = nmember;
10        return this.first + this.difference * (this.nmember - 1);
11    }
12
13    summa(nmember) {
14        this.nmember = nmember;
15        return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16            this.nmember;
17    }
18 }
19 let a = sluchch(3, 11);

```

```

20 let b = sluchch(2, 4);
21 let n = sluchch(4, 11);
22 let progression = new ArithmeticProgression(a, b);
23
24 NAtask.setTask({
25   text: ' Рабочие прокладывают тоннель длиной ' +
26     chislitlx(progression.summa(n), 'метр') +
27     ', ежедневно увеличивая норму прокладки на одно и то же число метров.' +
28     ' Известно, что за первый день рабочие проложили ' + chislitlx(a, 'метр') +
29     ' туннеля. ' +
30     ' Определите, сколько метров туннеля проложили рабочие в последний день,' +
31     ' если вся работа была выполнена за ' + chislitlx(a, 'день') + '.',
32   answers: progression.member(n),
33 });
34 //99580

```

Рабочие прокладывают тоннель длиной 98 метров, ежедневно увеличивая норму прокладки на одно и то же число метров. Известно, что за первый день рабочие проложили 5 метров туннеля. Определите, сколько метров туннеля проложили рабочие в последний день, если вся работа была выполнена за 7 дней.

Рабочие прокладывают тоннель длиной 160 метров, ежедневно увеличивая норму прокладки на одно и то же число метров. Известно, что за первый день рабочие проложили 6 метров туннеля. Определите, сколько метров туннеля проложили рабочие в последний день, если вся работа была выполнена за 8 дней.

```

1 class ArithmeticProgression {
2   constructor(first, difference) {
3     this.first = first;
4     this.difference = difference;
5     this.nmember = 0;
6   }
7
8   member(nmember) {
9     this.nmember = nmember;
10    return this.first + this.difference * (this.nmember - 1);
11  }
12
13  summa(nmember) {

```

```
14     this.nmember = nmember;
15
16     return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
17     this.nmember;
18 }
19
20 let a = sluchch(2, 10);
21 let b = sluchch(2, 4);
22 let n = sluchch(10, 24);
23 let n2 = n - sluchch(5, 7);
24
25 let progression = new ArithmeticProgression(a, b);
26
27 NAtask.setTask({
28     text: 'Грузовик перевозит партию щебня массой ' +
29         chislitlx(progression.summa(n), 'тонна') +
30         ', ежедневно увеличивая норму перевозки на одно и то же число тонн.' +
31         'Известно, что за первый день рабочие проложили ' + chislitlx(a, 'метр') +
32         '+ туннеля. ' +
33         'Определите, сколько тонн щебня было перевезено на ' +
34         om.porchesl[n2].i[0] +
35         ' день, если вся работа была выполнена за ' + chislitlx(a, 'день') + '.',

36     answers: progression.member(n2),
37 });
38 //99583
```

Грузовик перевозит партию щебня массой 608 тонн, ежедневно увеличивая норму перевозки на одно и то же число тонн. Известно, что за первый день было перевезено 8 тонн щебня. Определите, сколько тонн щебня было перевезено на десятый день, если вся работа была выполнена за 16 дней.

Рабочие прокладывают тоннель длиной 160 метров, ежедневно увеличивая норму прокладки на одно и то же число метров. Известно, что за первый день рабочие проложили 6 метров туннеля. Определите, сколько метров туннеля проложили рабочие в последний день, если вся работа была выполнена за 8 дней.

```
1 class ArithmeticProgression {  
2     constructor(first, difference) {  
3         this.first = first;
```

```

4     this.difference = difference;
5     this.nmember = 0;
6 }
7
8 member(nmember) {
9     this.nmember = nmember;
10    return this.first + this.difference * (this.nmember - 1);
11 }
12
13 summa(nmember) {
14     this.nmember = nmember;
15     return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16         this.nmember;
17 }
18
19 let a = sluchch(1, 15);
20 let b = sluchch(2, 4);
21 let n = sluchch(4, 23);
22 var arr = ['Петя', 'Вася', 'Миша', 'Олег', 'Федя', 'Руслан', 'Саша', 'Игорь',
23   'Артур', 'Толя', 'Борис', 'Гриша', 'Женя'];
24 var rand = Math.floor(Math.random() * arr.length);
25
26 let progression = new ArithmeticProgression(a, b);
27
28 NAtask.setTask({
29     text: sklonxkand(arr[rand]).de + 'надо решить ' +
30         chislitlx(progression.summa(n), 'задача', 'v') +
31         '. Он решает на одно и то же количество задач больше по сравнению с предыду-
32         щим днём.' +
33         ' Известно, что за первый день ' + arr[rand] ' + ' решил ' + chislitlx(a,
34         'задача', 'v') +
35         ' Определите, сколько задач решил '+ arr[rand] '+' в последний день,' +
36         ' если со всеми задачами он справился за ' + chislitlx(n, 'день') + '.',

37     answers: progression.member(n),
38 });
39 //99581

```

Борису надо решить 70 задач. Он решает на одно и то же количество задач больше по сравнению с предыдущим днем. Известно, что за первый день Борис решил 4 задачи. Определите, сколько задач решил Борис в последний день, если со всеми задачами он справился за 7 дней.

Жене надо решить 483 задачи. Он решает на одно и то же количество задач больше по сравнению с предыдущим днем. Известно, что за первый день Женя решил 15 задач. Определите, сколько задач решил Женя в последний день, если со всеми задачами он справился за 14 дней.

```
1 class ArithmeticProgression {
2     constructor(first, difference) {
3         this.first = first;
4         this.difference = difference;
5         this.nmember = 0;
6     }
7
8     member(nmember) {
9         this.nmember = nmember;
10        return this.first + this.difference * (this.nmember - 1);
11    }
12
13    summa(nmember) {
14        this.nmember = nmember;
15        return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16            this.nmember;
17    }
18
19 let a = sluchch(2, 10);
20 let b = sluchch(2, 4);
21 let n = sluchch(10, 24);
22 let n2 = n - sluchch(5, 7);
23 let progression = new ArithmeticProgression(a, b);
24
25 NAtask.setTask({
26     text: 'Турист едет из одного города в другой, каждый день проходя больше, '
27     +
28 }
```

```

27     'чем в предыдущий день, на одно и то же расстояние.' +
28     ' Известно, что за первый день турист прошёл ' + chislitlx(a, 'километр') +
29     + '.' +
30     ' Определите, сколько километров прошёл турист за ' + om.porhisl[n2].i[0] +
31     +
32     ' день, если весь путь он прошёл за ' + chislitlx(a, 'день') + ', а расстоя-
33     ние между городами составляет '+
34     chislitlx(progression.summa(n), 'километр') + '.',
35     answers: progression.member(n2),
36   );
37 //99582

```

Турист идет из одного города в другой, каждый день проходя больше, чем в предыдущий день, на одно и то же расстояние. Известно, что за первый день турист прошел 4 километра. Определите, сколько километров прошел турист за десятый день, если весь путь он прошел за 16 дней, а расстояние между городами составляет 424 километра.

Турист идет из одного города в другой, каждый день проходя больше, чем в предыдущий день, на одно и то же расстояние. Известно, что за первый день турист прошел 7 километров. Определите, сколько километров прошел турист за двенадцатый день, если весь путь он прошел за 18 дней, а расстояние между городами составляет 432 километра.

```

1 class ArithmeticProgression {
2   constructor(first, difference) {
3     this.first = first;
4     this.difference = difference;
5     this.nmember = 0;
6   }
7
8   member(nmember) {
9     this.nmember = nmember;
10    return this.first + this.difference * (this.nmember - 1);
11  }
12
13  summa(nmember) {
14    this.nmember = nmember;
15    return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16      this.nmember;
17  }

```

```

17 }
18
19 let a = sluchch(1, 5);
20 let b = sluchch(2, 4);
21 let n = sluchch(3, 11);
22
23 let progression = new ArithmeticProgression(a, b);
24
25 let q = a + progression.member(n);
26
27 NAtask.setTask({
28   text: 'Улитка ползет от одного дерева до другого.' +
29     ' Каждый день она проползает на одно и то же расстояние больше, чем в преды-
30     дущий день.' +
31     ' Известно, что за первый и последний дни улитка проползла в общей сложност-
32     и ' + chislitlx(q, 'метр') +
33     '. Определите, сколько дней улитка потратила на весь путь, если расстояние
34     между деревьями равно ' +
35     chislitlx(progression.summa(n), 'метр') + '.', ,
36   answers: n,
37 });
38 //99584

```

Улитка ползет от одного дерева до другого. Каждый день она проползает на одно и то же расстояние больше, чем в предыдущий день. Известно, что за первый и последний дни улитка проползла в общей сложности 8 метров. Определите, сколько дней улитка потратила на весь путь, если расстояние между деревьями равно 12 метров.

Улитка ползет от одного дерева до другого. Каждый день она проползает на одно и то же расстояние больше, чем в предыдущий день. Известно, что за первый и последний дни улитка проползла в общей сложности 32 метра. Определите, сколько дней улитка потратила на весь путь, если расстояние между деревьями равно 144 метра.

```

1 class ArithmeticProgression {
2   constructor(first, difference) {
3     this.first = first;
4     this.difference = difference;
5     this.nmember = 0;
6   }

```

```

7
8     member(nmember) {
9         this.nmember = nmember;
10        return this.first + this.difference * (this.nmember - 1);
11    }
12
13    summa(nmember) {
14        this.nmember = nmember;
15        return ((2 * this.first + this.difference * (this.nmember - 1)) / 2) *
16            this.nmember;
17    }
18
19 let a = sluchch(2, 10);
20 let b = sluchch(2, 4);
21 let n = sluchch(10, 24);
22 let n2 = n - sluchch(5, 7);
23 var arr = ['Лена', 'Ира', 'Настя', 'Маша', 'Катя', 'Вера', 'Саша', 'Оля',
24           'Даша', 'Тоня', 'Аня', 'Ира', 'Женя'];
25 var rand = Math.floor(Math.random() * arr.length);
26
27
28 NAtask.setTask({
29     text: sklonxkand(arr[rand]).de + 'надо подписать ' +
30           chislitlx(progression.summa(n), 'открытка', 'v') +
31           '. Ежедневно она подписывает на одно и то же количество открыток больше по
32           сравнению с предыдущим днём.' +
33           ' Известно, что за первый день ' + arr[rand] ' + ' подписала ' +
34           chislitlx(a, 'открытка', 'v') + '. ' +
35           ' Определите, сколько было подписано за ' + om.porchiisl[n2].i[0] +
36           ' день, если вся работа была выполнена за ' + chislitlx(n, 'день') + '.', ,
37     answers: progression.member(n2),
38 });
39 //99585

```

```

1 class GeometricProgression {
2     constructor(first, factor) {

```

Оле надо подписать 364 открытки. Ежедневно она подписывает на одно и то же количество открыток больше по сравнению с предыдущим днем. Известно, что за первый день Оля подписала 10 открыток. Определите, сколько открыток было подписано за седьмой день, если вся работа была выполнена за 13 дней.

Лене надо подписать 924 открытки. Ежедневно она подписывает на одно и то же количество открыток больше по сравнению с предыдущим днем. Известно, что за первый день Лена подписала 4 открытки. Определите, сколько открыток было подписано за четырнадцатый день, если вся работа была выполнена за 21 день.

```
3     this.first = first;
4     this.factor = factor;
5     this.nmember = 0;
6 }
7
8 member(nmember) {
9     this.nmember = nmember;
10    return (this.first * Math.pow(this.factor, this.nmember - 1)).toFixed(0);
11 }
12 }
13
14 let b = sluchch(3, 25);
15 let c;
16 let a = sluchch(500000, 1500000, 100000);
17
18 if (a < 1000000) {
19     n = 3;
20 } else {
21     b = sluchch(10, 20, 10);
22     n = sluchch(4, 5);
23 }
24
25 let c = 1 + b / 100;
26 c = c.toFixed(2);
27 let progression = new GeometricProgression(a, c);
28 var arr = ['Печенов', 'Оладьев', 'Коржов', 'Плюшкин', 'Прянников', 'Баранкин',
29             'Бубликов', 'Батонов', 'Ватрушкин', 'Булкин'];
30 var rand = Math.floor(Math.random() * arr.length);
```

```

31 NAtask.setTask({
32     text: 'Бизнесмен ' + arr[rand] + ' получил в 2000 году прибыль в размере ' +
33         chislitlx(a, 'рубль') + '.' +
34         ' Каждый следующий год его прибыль увеличивалась на ' + b +
35         ' Определите, сколько было подписано за ' + om.porchesl[n2].i[0] +
36         '% по сравнению с предыдущим годом. Сколько рублей заработал ' + arr[rand]
37         + ' за 200' + (n-1)+ 'год?',
38     answers: progression.member(n),
39 });
40 //99586

```

Бизнесмен Плюшкин получил в 2000 году прибыль в размере 700000 рублей. Каждый следующий год его прибыль увеличивалась на 24% по сравнению с предыдущим годом. Сколько рублей заработал Плюшкин за 2002 год?

Бизнесмен Баранкин получил в 2000 году прибыль в размере 1000000 рублей. Каждый следующий год его прибыль увеличивалась на 10% по сравнению с предыдущим годом. Сколько рублей заработал Баранкин за 2003 год?

```

1 class GeometricProgression {
2     constructor(first, factor) {
3         this.first = first;
4         this.factor = factor;
5         this.nmember = 0;
6     }
7
8     member(nmember) {
9         this.nmember = nmember;
10        return (this.first * Math.pow(this.factor, this.nmember - 1)).toFixed(0);
11    }
12 }
13
14 let b1 = sluchch(2, 3);
15 let a1 = sluchch(2000, 7000, 500);
16 n1 = sluchch(6, 9);
17
18 let b2 = b1 + sluchch(1, 2);
19 let a2 = a1 + sluchch(500, 3000, 500);
20 n2 = n1 - sluchch(2, 3);

```

```

21
22 let progression1 = new GeometricProgression(a1, b1);
23 let progression2 = new GeometricProgression(a2, b2);
24
25 NAtask.setTask({
26   text: 'Компания "Альфа", начала инвестировать средства в перспективную отрасль в 2001 году, ' +
27   'имея капитал в размере ' + a1 + ' долларов. ' +
28   'Каждый год, начиная с 2002 года, она получала прибыль, которая составляла ' +
29   '+ (b1-1) +
30   '00% от капитала предыдущего года. А компания "Бета" начала инвестировать с
31   'редства в другую отрасль в 200' + n2 +
32   ' году, имея капитал в размере ' + a2 + ' долларов, и, начиная с 200' + (n2 + 1) +
33   ' года, ежегодно получала прибыль, составляющую ' + (b2 - 1) + '00% от капи-
34   'тала предыдущего года.' +
35   ' На сколько долларов капитал одной из компаний был больше капитала другой
36   ' к концу 200' + n1 +
37   ' года, если прибыль из оборота не изымалась?',
38   answers: Math.abs(progression1.member(n1) - progression2.member(n1 - n2 +
39     1)),
40 });
41 //99587

```

Компания "Альфа", начала инвестировать средства в перспективную отрасль в 2001 году, имея капитал в размере 6000 долларов. Каждый год, начиная с 2002 года, она получала прибыль, которая составляла 100% от капитала предыдущего года. А компания "Бета" начала инвестировать средства в другую отрасль в 2004 году, имея капитал в размере 7500 долларов, и, начиная с 2005 года, ежегодно получала прибыль, составляющую 300% от капитала предыдущего года. На сколько долларов капитал одной из компаний был больше капитала другой к концу 2007 года, если прибыль из оборота не изымалась?

Компания "Альфа", начала инвестировать средства в перспективную отрасль в 2001 году, имея капитал в размере 6000 долларов. Каждый год, начиная с 2002 года, она получала прибыль, которая составляла 200% от капитала предыдущего года. А компания "Бета" начала инвестировать средства в другую отрасль в 2005 году, имея капитал в размере 8000 долларов, и, начиная с 2006 года, ежегодно получала прибыль, составляющую 400% от капитала предыдущего года. На сколько долларов капитал одной из компаний был больше капитала другой к концу 2008 года, если прибыль из оборота не изымалась?

```

1 (function() {
2   'use strict';
3
4   var c = sluchch(1, 9).pm();
5   var d = sluchch(2, 99);
6   var o = sluchch(2, 99);
7   var x = sluchch(1, 9).pm();
8   var b = d - c * x;
9
10  chas2.task.setEquationTask({
11    parts: [‘(’ + b + ‘+’ + c + ‘x)’, d],
12    wrapper: [‘\log_{’ + o + ‘}’, ‘’],
13    roots: x,
14    enablePartsSubtraction: 1,
15  }, {
16    tags: {
17      log: 1
18    },
19
20    analys: ‘Решение:’ +
21      ‘$$\log_{’ + o + ‘}{(’ + b + ‘+’ + c + ‘x)}=’ + ‘\log_{’ + o + ‘}{’ + ‘
22      d + ‘}’\Rightarrow ‘ + b + ‘+’ + c +
23      ‘x=’ + d + ‘\Rightarrow ‘ + c + ‘x=’ + (d - b) + ‘\’
24      ‘\Rightarrow ‘x=’ + (d - b) / c + ‘$$’ +
25      ‘$Ответ: x=’ + x + ‘$’,
26  });
27 })();

```

```

1 (function() {
2   retryWhileError(function() {
3     'use strict';
4
5     var b = sluchch(1, 10).pm();
6     var c = sluchch(2, 5);
7     var k = sluchch(2, 3);
8     var a=b.abs()*sluchch(1,10)+k;
9     var pm = " ";
10       var bome = b>0 ? '>' : '<';


```

Найдите корень уравнения

$$\log_{35} 81 = \log_{35}(36 + 9x)$$

Решение:

$$\log_{35} (36 + 9x) = \log_{35} 81 \Rightarrow 36 + 9x = 81 \Rightarrow 9x = 45 \Rightarrow x = 5$$

Ответ : $x = 5$

Найдите корень уравнения

$$\log_{98}(-43 - 8x) = \log_{98} 13$$

Решение:

$$\log_{98} (-43 - 8x) = \log_{98} 13 \Rightarrow -43 - 8x = 13 \Rightarrow -8x = 56 \Rightarrow x = -7$$

Ответ : $x = -7$

```
11
12
13     var x = [];
14
15     var x1 = (k - a) / b;
16     genAssertZ1000(x1, "Один из корней очень нецелый");
17     x.pushIf(x1, a + b * x1 !== 1 && a + b * x1 > 0);
18
19     if (c % 2 === 0) {
20         var x2 = (-k - a) / b;
21         genAssertZ1000(x2, "Один из корней очень нецелый");
22         x.pushIf(x2, a + b * x2 !== 1 && a + b * x2 > 0 && x1 !== x2);
23         pm = "\pm";
24     }
25     genAssertNonempty(x, "А корней-то и нет!");
26
27     chas2.task.setEquationTask({
28         parts: ['\log_{' + [a, b + 'x'].slag() + '}{' + k.pow(c) + '}'],
29         roots: x,
```

```

30     enablePartsSubtraction: 1,
31     handleMultipleRoots: 'randomExceptList',
32     forceMultipleRoots: sl(4),
33
34 }, {
35   tags: {
36     log: 1,
37   },
38   analys: 'Решение: ' +
39     '$$\\log_{' + a+'+' + b + 'x' + '}{' + k.pow(c) + '}=' + c+', \\'
40     '\\Rightarrow' +
41     '\\begin{cases} (' + b + 'x + ' + a + ')^{'} + c + ' = ' + k.pow(c) + ', \\\\
42     '\\forall x \\exists ' + b + 'x + ' + a + ' > 0 \\Rightarrow ' +
43     '\\begin{cases} ' + b + 'x + ' + a + ' = ' + pm + k + ', \\\\
44     '\\forall x \\exists ' + bome + ', \\frac{' + a + '-' + a + '}{' + b + '} \\leq ' +
45     '\\forall x \\exists ' + bome + ', \\frac{' + a + '-' + a + '}{' + b + '} \\geq ' +
46     '\\forall x \\exists ' + bome + ', \\frac{' + a + '-' + a + '}{' + b + '} = ' +
47     '\\forall x \\exists ' + bome + ', ' + '$', \\
48   });

```

Найдите корень уравнения

$$\log_{-2x+11} 27 = 3$$

Если корней несколько, в ответе укажите их произведение.

Решение:

$$\log_{11-2x} 27 = 3 \Rightarrow \begin{cases} (-2x+11)^3 = 27 \\ -2x+11 > 0 \\ -2x+11 \neq 1 \end{cases} \Rightarrow \begin{cases} -2x+11 = 3 \\ x < \frac{-11}{-2} \\ x \neq \frac{-10}{-2} \end{cases} \Rightarrow -2x+11 = 3 \Rightarrow x = 4.$$

Ответ: $x = 4$.

Найдите корень уравнения

$$\log_{-5x+38} 27 - 3 = 0$$

Если корней несколько, в ответе укажите их произведение.

Решение:

$$\log_{38-5x} 27 = 3 \Rightarrow \begin{cases} (-5x+38)^3 = 27 \\ -5x+38 > 0 \\ -5x+38 \neq 1 \end{cases} \Rightarrow \begin{cases} -5x+38 = 3 \\ x < \frac{-38}{-5} \\ x \neq \frac{-37}{-5} \end{cases} \Rightarrow -5x+38 = 3 \Rightarrow x = 7.$$

Ответ: $x = 7$.