

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Математический факультет

Кафедра теории функций и геометрии

Применение языка JavaScript для генерации ФОС ОГЭ и ЕГЭ

Курсовая работа

Направление 02.03.01 Математика и компьютерные науки

Зав.кафедрой \_\_\_\_\_ д.физ.-мат.н., проф. Е.М. Семёнов

Обучающийся \_\_\_\_\_ В.Д. Китаева

Руководитель \_\_\_\_\_ д.физ.-мат.н., проф. Е.М. Семёнов

Воронеж 2024

# Содержание

<b>Введение</b>	<b>3</b>
<b>1 Реализация алгоритмов на базе проекта Час ЕГЭ</b>	<b>4</b>
<b>2 Вклад автора в расширение каталога</b>	<b>8</b>
2.1 Преимущества программной генерации заданий . . . . .	8
2.2 Вклад автора в расширение каталога . . . . .	8
<b>Заключение</b>	<b>16</b>

# Введение

Единый государственный экзамен (ЕГЭ) — централизованно проводимый в Российской Федерации экзамен в средних учебных заведениях — школах, лицеях и гимназиях, форма проведения ГИА (Государственной Итоговой Аттестации) по образовательным программам среднего общего образования. Служит одновременно выпускным экзаменом из школы и вступительным экзаменом в вузы.

За два года подготовки к ЕГЭ школьники сталкиваются с дефицитом заданий для подготовки. А учителя со списыванием ответов при решении задач экзамена учениками. Также в в конце 2021 года в список заданий ЕГЭ были добавлены новые задания под номером 9, количество которых для прорешивания очень мало. Были добавлены и текстовые задания в ограниченном количестве. Проект «Час ЕГЭ» позволяет решить все эти проблемы. «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. Для пользователей предназначены четыре оболочки (режима работы): «Случайное задание», «Тесты на печать», «Полный тест» и «Мини-интеграция». «Час ЕГЭ» является полностью открытым (код находится под лицензией GNU GPL 3.0) и бесплатным. В настоящее время в проекте полностью реализованы тесты по математике с кратким ответом (бывшая «часть В»). Планируется с течением времени включить в проект тесты по другим предметам школьной программы.

Первая глава этой работы посвящена реализации алгоритмов на базе проекта Час ЕГЭ.

Вторая глава представляет решение проблемы нехватки задач ЕГЭ и ОГЭ; шаблоны задач, написанные автором и примеры генерации.

# 1. Реализация алгоритмов на базе проекта Час ЕГЭ

В этой главе мы приводим вспомогательные функции и алгоритм написания шаблона текстовой задачи.

Функции, используемые в проекте

За 10 лет работы над проектом «Час ЕГЭ» была разработана нестандартная библиотека для упрощения многих задач. Далее представлены наиболее используемые функции из неё. Вспомогательные функции `function sluchch(n,k,s)` Возвращает случайное число от `n` до `k` с шагом `s` (по умолчанию 1). Эта функция используется настолько часто, что для неё была придумана сокращённая форма `sl()`. `function slKrome(a,p1,p2,p3)` Возвращает случайное число, кроме `a`. Если `a` – массив, то не содержащееся в нём; если число или строка, то не равное ему; Если функция, принимающая параметр - то не удовлетворяющее ей.

Работа с числами

1. `Number.prototype.chislitlx=function(p1,p2)` Возвращает строку, состоящую из данного числа и подходящего падежа слова `p1`, при этом полученное словосочетанию стоит в падеже `p2` (если не указан - именительный).
2. `Number.prototype.pow=function(n)` Возвращает число в степени `n`.
3. `Number.prototype.sqrt=function(n)` Возвращает квадратный корень из числа.
4. `Number.prototype.sqr=function()` Возвращает квадрат числа.
5. `Number.prototype.abs=function()` Возвращает модуль числа.
6. `Number.prototype.floor=function()` Возвращает число, округленное до целого в меньшую сторону.
7. `Number.prototype.ceil=function()` Возвращает число, округленное до целого в большую сторону.

8. `Number.prototype.pm=function()` Случайным образом возвращает число или ему противоположное.

### Работа со строками

1. `Number.prototype.toZagl=function()` Возвращает исходную строку с первой заглавной буквой.
2. `setEquationTask`-отвечает за составление задач с уравнениями.
3. `roots[]` задаёт корни уравнения

Для примера возьмём задание №27482.

**Задача №27482.** Причалы  $K$  и  $B$  расположены на водохранилище, расстояние между ними равно 255 км. Моторная лодка отправилась с постоянной скоростью из  $K$  в  $B$ . Через неделю после прибытия она отправилась тем же путём обратно со скоростью на 4 км/ч больше прежней, сделав по пути остановку на 1 час. В результате она затратила на обратный путь столько же времени, сколько на путь из  $K$  в  $B$ . Определите скорость моторной лодки на пути из  $K$  в  $B$ . Ответ дайте в км/ч.

1. Выбираем задание из Открытого Банка Заданий ЕГЭ и копируем его текст.
2. Добавляем ответ в поле `answers` (по умолчанию 0).
3. Инициализируем всех необходимые переменные для задачи (вес, проценты и так далее). Присваиваем им значения при помощи функции `sluchch()` или `slKrome()` (см. главу 2). Для хранения ответа создаём отдельную переменную.
4. Заменяем все числа в тексте на переменные (при помощи `+'.'+`).
5. Обособляем слова, которые не влияют на условия задачи. Это могут быть имена, профессии, транспорт и т.п.

6. Создаём переменные, которые будут отвечать за выбранные в прошлом пункте слова, и заменяем слова на переменные в тексте задачи. Выбираем их значения из массивов при помощи `iz()`
7. Иногда в задании выбранные слова используются в разных падежах. Для этого в проекте существует лексический модуль. Используем на склоняемых словах функцию `sklonlxkand()`. Теперь необходимо указать падеж слов в задании. Также при необходимости заглавной буквы в слове используем `toZagl()`. Если в тексте задачи присутствуют слова, зависящие от числительных, к ним применяется функция `chislitlx()`.
8. Далее составляем общее решение с учетом всех переменных и особенностей задачи.
9. Заготовка шаблона имеет следующий вид.

```
1 (function() { 'use strict'; retryWhileError(function() {
2   let s=slKrome(10, 300, 1);
3   let a=sl(1, 20);
4   let b=sl(1, 7);
5   let v=Math.sqrt(b**2*a**2+4*b*s*a);
6     let x=(-b*a+v)/(2*b);
7   genAssert(x>0, 'Скорость не может быть отрицательной');
8   genAssertZ1000(x, 'Скорость не может быть слишком дробной');
9     let the_berthForFloatingVehicle =
      sklonlxkand(decor.berthForFloatingVehicle.iz());
10  let the_waterbodyWithoutCurrent =
      sklonlxkand(decor.waterbodyWithoutCurrent.iz());
11  let the_activeFloatingVehicle =
      sklonlxkand(decor.activeFloatingVehicleF.iz());
12  let the_afterAWhile = decor.afterAWhile.iz();
13  let the_orderToFind = decor.orderToFind.iz();
14    NATask.setTask({
15    text:
```

```

16      '' + the_berthForFloatingVehicle.im.toZagl() +' А и В
расположены на ' + the_waterbodyWithoutCurrent.pe +', '+'
17      'расстояние между ними равно ' + s + ' км. ' +
the_activeFloatingVehicle.ie.toZagl() +' отправилась с постоянной
скоростью из А в В. ' +
18      the_afterAWhile.toZagl() +' после прибытия она отправилась тем же
путём обратно со скоростью на ' + a + ' кмч/ больше прежней, '+'
19      'сделав по пути остановку на ' + chislitlx(b, 'час') + '. В
результате она затратила на обратный путь столько же времени, '+'
20      'сколько на путь из А в В. ' + the_orderToFind.toZagl() +'
скорость ' + the_activeFloatingVehicle.re +' на пути из А в В. Ответ
дайте в кмч/.',
21      answers: x,
22  });
23  NATask.modifiers.allDecimalsToStandard(/*true*/);
24  NATask.modifiers.variativeABC();
25 }, 2000);})();

```

Примеры генерации задний приведены во второй главе.

## 2. Вклад автора в расширение каталога

В этой главе решается проблема нехватки заданий для подготовки к ЕГЭ, а также приводятся аргументы в пользу программного написания шаблонов для подготовки к ЕГЭ.

### 2.1. Преимущества программной генерации заданий

На примере предыдущей задачи было явно показано превосходство шаблонов над заданиями из Открытого Банка Заданий, а именно:

1. Большое количество разнообразных задач одного типа.
2. Простота и скорость написания шаблонов.
3. Невозможность нахождения учащимися ответов на задачи.

### 2.2. Вклад автора в расширение каталога

```
1 (function() { 'use strict'; retryWhileError(function() {  
2   let s=sl(1, 200, 1);  
3   let n=sl(1, 10, 1);  
4   let a=sl(1, 15, 1);  
5   let v=(a**2*n**2+2*s*a*n).sqrt();  
6     let x=v/n;  
7   genAssert(x>0, 'Скорость не может быть отрицательной');  
8   genAssertZ1000(x, 'Скорость не должна быть слишком дробной');  
9     let the_activeFloatingVehicle =  
       sklonljkand(["лодка", "байдарка", "баржа", "яхта", "моторная лодка"].iz());  
10  let the_humanSettlementDestination = sklonljkand(["пункт", "город"].iz());  
11  let the_orderToFind = decor.orderToFind.iz();  
12    NATask.setTask({  
13    text:
```



```

14      '' + the_activeFloatingVehicle.ie.toZagl() +' прошла против течения
реки ' + s +' км и вернулась в ' + the_humanSettlementDestination.ie +
15      ' отправления, затратив на обратный путь на ' + chislitlx(n, 'час') +'
меньше. ' + the_orderToFind.toZagl() +' скорость '+'
16      the_activeFloatingVehicle.re + ' в неподвижной воде, если скорость
течения равна ' + a + ' кмч/. Ответ дайте в кмч/.',
17      answers: x,
18  });
19  NATask.modifiers.allDecimalsToStandard(/*true*/);
20 }, 2000);})();

```

Листинг 1: 26586.js

## Примеры генерируемых задач 26586.js

Моторная лодка прошла против течения реки 160 км и вернулась в пункт отправления, затратив на обратный путь на 1 час меньше. Вычислите скорость моторной лодки в неподвижной воде, если скорость течения равна 4 км/ч. Ответ дайте в км/ч.3626586.js

Байдарка прошла против течения реки 96 км и вернулась в пункт отправления, затратив на обратный путь на 4 часа меньше. Определите скорость байдарки в неподвижной воде, если скорость течения равна 1 км/ч. Ответ дайте в км/ч.726586.js

Яхта прошла против течения реки 180 км и вернулась в город отправления, затратив на обратный путь на 1 час меньше. Найдите скорость яхты в неподвижной воде, если скорость течения равна 1 км/ч. Ответ дайте в км/ч.1926586.js

```

1  (function() { 'use strict'; retryWhileError(function() {
2      let n=s1(1, 3, 1);
3      let s=s1(10, 400, 1);
4      let a=s1(5,50,1);
5      let x=s1(1, 30, 1);
6      let k=n+s/(a-x)+s/(a+x);
7      genAssert(Number.isInteger(k),'Время не должно быть дробным');

```

```

8     genAssert(k>n,'Общее затраченное время больше стоянки');
9     genAssert(k<20,'Время стоянки не должно быть слишком большим');
10    let the_activeFloatingVehicle =
sklonlxxkand(decor.activeFloatingVehicle.iz());
11    let the_humanSettlementDestination =
sklonlxxkand(decor.humanSettlementDestination.iz());
12    let the_orderToFind = decor.orderToFind.iz();
13    NATask.setTask({
14        text:
15            '' + the_activeFloatingVehicle.ie.toZagl() + ' проходит по течению
реки до ' + the_humanSettlementDestination.re + ' назначения ' + s +
16            ' км и после стоянки возвращается в ' +
the_humanSettlementDestination.ie + ' отправления. ' +
17            the_orderToFind.toZagl() + ' скорость течения, если скорость ' +
the_activeFloatingVehicle.re + ' в неподвижной воде равна ' + a +
18            ' кмч/, стоянка длится ' + chislitlx(n, 'час') + ', ' + 'а в ' +
the_humanSettlementDestination.ie + ' отправления ' +
19            the_activeFloatingVehicle.ie + ' возвращается через ' + chislitlx(k,
'час') + '. ' +
20            'Ответ дайте в кмч/.',
21        answers: x,
22        authors: ['VeronikaKit'],
23    });
24    NATask.modifiers.allDecimalsToStandard(/*true*/);
25    }, 2000);})();

```

Листинг 2: 26588.js

## Примеры генерируемых задач 26588.js

Байдарка проходит по течению реки до деревни назначения 18 км и после стоянки возвращается в деревня отправления. Определите скорость течения, если скорость байдарки в неподвижной воде равна 24 км/ч, стоянка длится 1 час, а в деревня отправления байдарка возвращается через 3 часа. Ответ дайте в

км/ч.1226588.js

Баржа проходит по течению реки до деревни назначения 160 км и после стоянки возвращается в деревня отправления. Определите скорость течения, если скорость баржи в неподвижной воде равна 26 км/ч, стоянка длится 2 часа, а в деревня отправления баржа возвращается через 15 часов. Ответ дайте в км/ч.6326588.js

Корабль проходит по течению реки до города назначения 63 км и после стоянки возвращается в город отправления. Определите скорость течения, если скорость корабля в неподвижной воде равна 16 км/ч, стоянка длится 2 часа, а в город отправления корабль возвращается через 10 часов. Ответ дайте в км/ч.226588.js

```
1 (function() { 'use strict'; retryWhileError(function() {
2   let s=slKrome(10, 300, 1);
3   let a=sl(1, 20);
4   let b=sl(1, 7);
5   let v=Math.sqrt(b**2*a**2+4*b*s*a);
6   let x=(-b*a+v)/(2*b);
7   genAssert(x>0, 'Скорость не может быть отрицательной');
8   genAssertZ1000(x, 'Скорость не может быть слишком дробной');
9   let the_berthForFloatingVehicle =
    sklonlXkand(decor.berthForFloatingVehicle.iz());
10  let the_waterbodyWithoutCurrent =
    sklonlXkand(decor.waterbodyWithoutCurrent.iz());
11  let the_activeFloatingVehicle =
    sklonlXkand(decor.activeFloatingVehicleF.iz());
12  let the_afterAWhile = decor.afterAWhile.iz();
13  let the_orderToFind = decor.orderToFind.iz();
14  NATask.setTask({
15    text:
16      '' + the_berthForFloatingVehicle.im.toZagl() + ' А и В расположены
на ' + the_waterbodyWithoutCurrent.pe + ', ' +
17      'расстояние между ними равно ' + s + ' км. ' +
the_activeFloatingVehicle.ie.toZagl() + ' отправилась с постоянной скоростью
из А в В. ' +
18      the_afterAWhile.toZagl() + ' после прибытия она отправилась тем же путём
```

```

    обратно со скоростью на ' + a + ' кмч/ больше прежней, '+'
19     'сделав по пути остановку на ' + chislitlx(b, 'час') + '. В результате
    она затратила на обратный путь столько же времени, '+'
20     'сколько на путь из A в B. ' + the_orderToFind.toZagl() + ' скорость ' +
    the_activeFloatingVehicle.re + ' на пути из A в B. Ответ дайте в кмч/.',
21     answers: x,
22 });
23 NATask.modifiers.allDecimalsToStandard(/*true*/);
24 NATask.modifiers.variativeABC();
25 }, 2000);})();

```

### Листинг 3: 27482.js

#### Примеры генерируемых задач 27482.js

Пристани  $R$  и  $C$  расположены на озере, расстояние между ними равно 120 км. Баржа отправилась с постоянной скоростью из  $R$  в  $C$ . Через два дня после прибытия она отправилась тем же путём обратно со скоростью на 4 км/ч больше прежней, сделав по пути остановку на 1 час. В результате она затратила на обратный путь столько же времени, сколько на путь из  $R$  в  $C$ . Найдите скорость баржи на пути из  $R$  в  $C$ . Ответ дайте в км/ч.2027482.js

Пристани  $J$  и  $L$  расположены на водохранилище, расстояние между ними равно 168 км. Моторная лодка отправилась с постоянной скоростью из  $J$  в  $L$ . На следующий день после прибытия она отправилась тем же путём обратно со скоростью на 12 км/ч больше прежней, сделав по пути остановку на 7 часов. В результате она затратила на обратный путь столько же времени, сколько на путь из  $J$  в  $L$ . Найдите скорость моторной лодки на пути из  $J$  в  $L$ . Ответ дайте в км/ч.1227482.js

Пристани  $M$  и  $E$  расположены на водохранилище, расстояние между ними равно 136 км. Моторная лодка отправилась с постоянной скоростью из  $M$  в  $E$ . Через два дня после прибытия она отправилась тем же путём обратно со скоростью на 17 км/ч больше прежней, сделав по пути остановку на 4 часа. В результате она затратила на обратный путь столько же времени, сколько на путь из  $M$  в  $E$ . Вычислите

скорость моторной лодки на пути из  $M$  в  $E$ . Ответ дайте в км/ч.1727482.js

```
1 (function() { 'use strict'; retryWhileError(function() {
2   let s=sl(1, 200);
3   let n=sl(1, 10, 1);
4   let a=sl(1, 15, 1);
5   let v=(a**2*n**2+2*s*a*n).sqrt();
6   let x=v/n;
7   genAssert(x>0, 'Скорость не может быть отрицательной');
8   genAssertZ1000(x, 'Скорость не должна быть слишком дробной');
9   let the_activeFloatingVehicle =
    sklonlхkand(decor.activeFloatingVehicleF.iz());
10  let the_humanSettlementDestination = sklonlхkand(["пункт", "город"].iz());
11  let the_orderToFind = decor.orderToFind.iz();
12    NATask.setTask({
13      text:
14        the_activeFloatingVehicle.ie.toZagl() +' прошла против течения реки '
15      + s +' км и вернулась в ' + the_humanSettlementDestination.ie +
16      ' отправления, затратив на обратный путь на ' + chislitlх(n, 'час') +'
17      меньше. ' + the_orderToFind.toZagl() +' скорость '+
18      the_activeFloatingVehicle.re + ' в неподвижной воде, если скорость
19      течения равна ' + a + ' кмч/. Ответ дайте в кмч/.',
20      answers: x,
21    });
22    NATask.modifiers.allDecimalsToStandard(/*true*/);
23  }, 2000);})();
```

Листинг 4: 111557.js

## Примеры генерируемых задач 111557.js

Яхта прошла против течения реки 140 км и вернулась в пункт отправления, затратив на обратный путь на 8 часов меньше. Найдите скорость яхты в неподвижной воде, если скорость течения равна 1 км/ч. Ответ дайте в км/ч.6111557.js

Лодка прошла против течения реки 21 км и вернулась в пункт отправления,

затратив на обратный путь на 9 часов меньше. Вычислите скорость лодки в неподвижной воде, если скорость течения равна 6 км/ч. Ответ дайте в км/ч.8111557.js

Яхта прошла против течения реки 120 км и вернулась в пункт отправления, затратив на обратный путь на 2 часа меньше. Определите скорость яхты в неподвижной воде, если скорость течения равна 8 км/ч. Ответ дайте в км/ч.32111557.js

```
1 (function() { 'use strict'; retryWhileError(function() {
2   let s=sl(1, 200, 1);
3   let k=sl(1, 15, 1);
4   let b=sl(1, 24, 1);
5   let n=sl(1, 5, 1);
6   let x=slKrome(k, 5, 50);
7   let a=b-n-s/(x+k)-s/(x-k);
8   genAssert(Number.isInteger(a),'Время не может быть дробным');
9   genAssert(a>0,'Время отправления не может быть отрицательным');
10  genAssert(a<15,'Время отправления не может быть слишком большим');
11  genAssert(b>a,'Время отправления не может быть больше времени прибытия');
12  let the_activeFloatingVehicle =
    sklonlXkand(["пароход","теплоход","каяк","корабль","паром","катер"].iz());
13 let the_orderToFind = decor.orderToFind.iz();
14   NATask.setTask({
15     text:
16       '' + the_activeFloatingVehicle.ie.toZagl() + ' в ' + a + ':00 вышел
по течению реки из пункта А в пункт В, расположенный в ' +
17       s + ' км от А. Пробыв в пункте В ' + chislitlx(n, 'час') + ', ' +
the_activeFloatingVehicle.ie +
18       ' отправился назад и вернулся в пункт А в ' + b + ':00 того же дня. ' +
the_orderToFind.toZagl() + ' собственную скорость ' +
19       the_activeFloatingVehicle.re + 'в( кмч/), если известно, что скорость
течения реки ' + k + ' кмч/. Ответ дайте в кмч/.',
20     answers: x,
21   });
22   NATask.modifiers.allDecimalsToStandard(/*true*/);
23 }, 2000);})();
```

### Примеры генерируемых задач 323375.js

Корабль в 10 : 00 вышел по течению реки из пункта  $A$  в пункт  $B$ , расположенный в 45 км от  $A$ . Пробыв в пункте  $B$  3 часа, корабль отправился назад и вернулся в пункт  $A$  в 22 : 00 того же дня. Определите собственную скорость корабля(в км/ч), если известно, что скорость течения реки 12 км/ч. Ответ дайте в км/ч.18323375.js

Теплоход в 1 : 00 вышел по течению реки из пункта  $A$  в пункт  $B$ , расположенный в 95 км от  $A$ . Пробыв в пункте  $B$  2 часа, теплоход отправился назад и вернулся в пункт  $A$  в 15 : 00 того же дня. Найдите собственную скорость теплохода(в км/ч), если известно, что скорость течения реки 14 км/ч. Ответ дайте в км/ч.24323375.js

Теплоход в 6 : 00 вышел по течению реки из пункта  $A$  в пункт  $B$ , расположенный в 4 км от  $A$ . Пробыв в пункте  $B$  2 часа, теплоход отправился назад и вернулся в пункт  $A$  в 9 : 00 того же дня. Определите собственную скорость теплохода(в км/ч), если известно, что скорость течения реки 3 км/ч. Ответ дайте в км/ч.9323375.js

## Заключение

В ходе выполнения курсовой работы за 3 курс было написано:

- Текстовые задачи ЕГЭ — 11 шаблонов принято.
- Уравнения ОГЭ — 15 шаблонов.

Все добавленные в проект задания можно использовать для составления контрольных работ, проведения текущего контроля знаний учащихся, подготовки к ЕГЭ.

В будущем планируется добавить в проект большее количество заданий ЕГЭ и ОГЭ различных типов.



## Список литературы

- [1] Тренажёр "Час ЕГЭ". – URL: <https://math.vsu.ru/chas-ege/sh/katalog.html>
- [2] Федеральный институт педагогических измерений. – URL: <https://fipi.ru/ege/otkrytyy-bank-zadaniy-ege>
- [3] Момот Е. А., Арахов Н. Д. Разработка и внедрение ПО для сбора статистики результатов подготовки к ЕГЭ по математике профильного уровня // Актуальные проблемы прикладной математики, информатики и механики. – 2021. – С. 1-2.
- [4] Открытый банк задач ЕГЭ по математике. Профильный уровень. – URL: <https://prof.mathege.ru/>
- [5] Единый государственный экзамен. – URL: [https://ru.wikipedia.org/wiki/Единый\\_государственный\\_экзамен](https://ru.wikipedia.org/wiki/Единый_государственный_экзамен)
- [6] Решу ЕГЭ - Сдам ГИА. – URL: <https://ege.sdamgia.ru/problem?id=27074>