

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
КАФЕДРА ИНФОРМАТИКИ И МАТЕМАТИЧЕСКОГО ОБЕСПЕЧЕНИЯ

09.03.04 - Программная инженерия

Отчет по дисциплине «Верификация программного обеспечения»

Информационная система «СМАЛТ»

Выполнил:

студент 4 курса группы 22407

Е. С. Харичев _____
подпись

Преподаватель:

К. А. Кулаков, к.ф-м.н., доцент

подпись

Содержание

Объект тестирования	3
Описание системы	3
Рамки тестирования	4
Функционал объекта тестирования	4
Стратегия тестирования	5
Структура объекта тестирования	5
Стратегия блочного тестирования	7
Стратегия интеграционного тестирования	8
Стратегия аттестационного тестирования	8
Стратегия нагрузочного тестирования	8
Критерии прохождения тестирования	8
Условия возобновления и приостановки выполнения тестов	9
Косвенные входные данные	10
Детальный план тестов	14
Блочное тестирование	14
Интеграционное тестирование	55
Аттестационное тестирование	57
Аттестационное тестирование	61
Примеры реализации тестов	64
Журнал тестирования	65
Журнал блочного тестирования	65
Журнал интеграционного тестирования	68
Журнал аттестационного тестирования	68
Журнал нагрузочного тестирования	69
Журнал найденных ошибок	70
Результаты	74

Объект тестирования

Описание системы

Объектом тестирования является часть информационной системы «Статистические методы анализа литературного текста» (ИС «СМАЛТ»), которая в своей основе имеет достаточно большую БД литературных произведений с морфологическим и синтаксическим параметрами из публицистики 60-70 гг. 19 века.

Главные цели системы - сбор, централизованное хранение литературных произведений вместе с их грамматической и синтаксической структурами, а также статистическая обработка и анализ этих структур.

Система состоит из двух основных блоков:

1. функциональный блок (морфологический и синтаксический анализ, пополнение БД, внесение исправлений);
2. аналитический блок (модули, реализующие разнообразные методики статистического анализа).

Система имеет клиент-серверную архитектуру. Для верстки клиента используется фреймворк Bootstrap (HTML, CSS, JS). Сервер разрабатывается без использования фреймворков на PHP. В качестве БД используется MySQL.

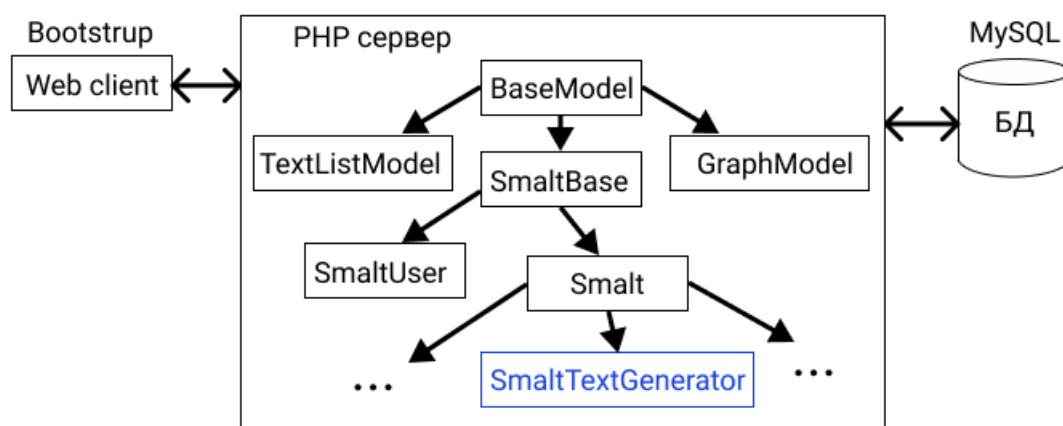


Рис. 1: Архитектура ИС «СМАЛТ»

Ссылка на систему: <http://smalt.karelia.ru/shower/>

Рамки тестирования

Объектом тестирования выступает модуль аналитического блока ИС под названием «Генератор текстов», отвечающий за генерацию тестовых текстов для других исследований. Его основной функционал представлен классом `SmaltTextGenerator`, методы которого и будут участвовать в представленном тестировании. За отображение результатов работы класса отвечает скрипт `text-generator.php`.

Функционал объекта тестирования

Генератор текстов представляет собой страничку с формой (A1), на которой динамически выводится результат генерации. Генератор получает на вход два текста (случайные или выбранные), затем в соответствии с заданными характеристиками (размер фрагмента и доля вставок) формирует выходной текст путем последовательного выбора и вставки фрагментов одного текста в другой. Фрагменты выбираются случайным образом и должны быть распределены по текстам равномерно.

Основные функции генератора:

- генерация кода (A2, A3);
- генерация связки код + текст (A4, A5);
- копирование кода в буфер обмена (A6);
- просмотр сгенерированного текста на отдельной странице (A8);
- экспорт сгенерированного текста (A7).

При вводе значений, выходящих за границы допустимых (например доля вставок должна быть в диапазоне $[0.01; 0.95]$) отображаются ошибки, исправив которые пользователь получает результат генерации (A9, A10, A11).

Стратегия тестирования

Структура объекта тестирования

Генератор текстов представлен одним классом `SmaltTextGenerator`. Функции данного класса описаны в таблице ниже.

Название	Вход	Выход	Назначение
<code>rint()</code> Б1-3	<code>int \$start</code> - начало интервала <code>int \$end</code> - конец интервала	<code>int</code>	Функция возвращает случайное целое число из интервала.
<code>rText()</code> Б4-5	<code>int \$baseTextlistId</code> - id списка основных текстов <code>int \$otherTextlistId</code> - id списка вставляемых текстов	<code>array(int, int)</code>	Функция возвращает id случайных текстов из списков.
<code>getTextLength()</code> Б6-7	<code>int \$textId</code> - id текста	<code>int</code>	Функция считает длину текста в словах.
<code>parseCode()</code> Б8-11, И1-5	<code>string \$code</code> - код для замены фрагментов	<code>array(int, int, array)</code>	Функция читает код и представляет его в удобном для дальнейшей работы виде: id основного текста, id вставляемого текста, массив фрагментов.
<code>getArrayOfShifts()</code> Б12-17	<code>int \$insCount</code> - количество вставок <code>int \$shiftsCount</code> - количество доступных сдвигов	<code>array()</code>	Функция предназначена для получения логического массива сдвигов для случаев, когда количество доступных сдвигов не превышает количество вставок.

Таблица 1: Функции класса `SmaltTextGenerator`

Название	Вход	Выход	Назначение
getTexts- Code() Б18-27, И4-5	int <i>\$baseId</i> - id основного текста int <i>\$otherId</i> - id вставляемого текста int <i>\$baseLength</i> - размер основного текста int <i>\$otherLength</i> - размер вставляемого текста int <i>\$fragmentSize</i> - размер фрагмента float <i>\$percentOfInserts</i> - доля вставок	string	Функция формирует код, по которому будет производится вставка фрагментов.
check- Params() Б28-31	int <i>\$baseLength</i> - размер основного текста int <i>\$otherLength</i> - размер вставляемого текста int <i>\$fragmentSize</i> - размер фрагмента int <i>\$percentOfInserts</i> - доля вставок	bool	Функция проверяет указанные параметры, их зависимости.
validate- Form() Б32-36	array <i>\$params</i> - массив данных формы	bool	Функция проверяет допустимые диапазоны ввода данных.
processText() Б37-38, И1-3, И5	int <i>\$baseId</i> - id основного текста int <i>\$otherId</i> - id вставляемого текста array <i>\$intervals</i> - массив интервалов замен bool <i>\$showText</i> - флаг печати текста на странице	void	Функция обрабатывает тексты в соответствии с массивом интервалов и выводит результат на странице или сохраняет в текстовую переменную.

Таблица 1: Функции класса SmaltTextGenerator

Название	Вход	Выход	Назначение
getNew- Fragment- Positions() Б39-87	<p>int <i>\$startPos</i> - позиция начала фрагмента</p> <p>int <i>\$endPos</i> - позиция конца фрагмента</p> <p>array <i>\$sLR</i> - массив расстояний до ближайших границ предложений влево и вправо для начала фрагмента</p> <p>array <i>\$eLR</i> - массив расстояний до ближайших границ предложений влево и вправо для конца фрагмента</p> <p>int <i>\$lastEndPos</i> - позиция конца предыдущего фрагмента</p> <p>int <i>\$textSize</i> - длина текста в словах</p>	array(int, int)	Функция изменяет границы фрагментов в соответствии с границами предложений

Таблица 1: Функции класса SmaltTextGenerator

Стратегия блочного тестирования

Блочные тесты предполагают тестирование отдельных функций и модулей программы отдельно друг от друга. Для блочного тестирования будет применяться модуль RHPUnit. Для проведения модульных тестов должна быть использована копия структуры базы данных, заполненная тестовыми данными (т. к. многие функции взаимодействуют с БД). Должен быть разработан как минимум один тест на каждый модуль.

Будут протестированы все функции указанные в Таблице 1 (П. «Структура объекта тестирования»)

Стратегия интеграционного тестирования

При интеграционном тестировании отдельные программные модули объединяются и тестируются как группа для оценки соответствия системы или компонента заданным функциональным требованиям. Для интеграционного тестирования будет применяться модуль PHPUnit. Так как объектом тестирования является один класс, то объединять в группы будем функции. Должна быть использована копия структуры базы данных, заполненная тестовыми данными (т. к. многие функции взаимодействуют с БД). Варианты объединений функций (иные не возможны):

1. `getTextsCode()` и `parseCode()`;
2. `parseCode()` и `processText()`;
3. `getTextsCode()`, `parseCode()` и `processText()`.

Стратегия аттестационного тестирования

В ходе аттестационного тестирования будет проверена работоспособность приложения и его возможность осуществлять заявленный функционал. Аттестационное тестирование будет проводиться методом «живого человека». Тестирующий человек, по заранее заданным инструкциям, производит требуемые действия и сверяется с заранее заданными результатами. Тест считается пройденным, если ожидаемый результат совпадает с фактическим результатом. В противном случае тест считается не пройденным.

Стратегия нагрузочного тестирования

Нагрузочное тестирование – это процесс умышленной нагрузки системы, с целью определения показателей производительности, времени отклика, проверки соответствия требованиям, которые были предъявлены к данной системе или отдельному устройству. Умышленная нагрузка в нашем случае будет заключаться в постепенном увеличении количества вставок при средней длине фрагмента (15-20 слов) для текста большого объема (около 42000 слов)

Критерии прохождения тестирования

Тест считается успешно пройденным, если ожидаемый и фактический результаты совпадают. Если тест завершается неудачей, то перед принятием решения целесообразно про-

верить правильность самого теста. Если тест завершился неудачей и тест реализован правильно, то производится заключение о найденной ошибке. Тестирование считается пройденным, если во время его прохождения не выявлено критических ошибок и количество пройденных тестов составляет не менее 85% от общего числа тестов.

Условия возобновления и приостановки выполнения тестов

Тестирование должно быть приостановлено, если количество не пройденных тестов превысит 15% от их общего количества. Тестирование должно быть приостановлено при обнаружении критических ошибок. Тестирование возобновляется после исправления ошибок, выявленных при предыдущем тестировании. Повторное тестирование должно быть выполнено с самого начала, начиная с блочных тестов.

Косвенные входные данные

Для функций работающих с БД были подготовлены фикстуры для следующих таблиц:

- magazine;
- author;
- text;
- word;
- sys_users;
- textlist_description;
- textlist_items;
- entries.

Пример заполнения данных для таблицы **magazine**:

```
array(  
  '20' => array(  
    'ID' => '20', 'TITLE' => 'magazine1', 'ORIGIN_TITLE' => ''  
  )  
);
```

Пример заполнения данных для таблицы **author**:

```
array(  
  '44' => array(  
    'ID' => '44', 'NAME' => 'Иванов_И._И.',  
    'ORIGIN_NAME' => '', 'REAL_NAME' => ''  
  ),  
  '45' => array(  
    'ID' => '45', 'NAME' => 'Петров_А._А.',  
    'ORIGIN_NAME' => '', 'REAL_NAME' => ''  
  )  
);
```

Пример заполнения данных для таблицы **text**:

```

array(
  '329' => array(
    'ID' => '329', 'TITLE' => 'Очень_маленький_текст',
    'AUTHOR_ID' => '44', 'MAGAZINE_ID' => '20', 'MAGAZINE_NO' => '1',
    'PUBLICATION_DATE' => '2022-12-02', 'COMMENT' => NULL,
    'URL' => NULL, 'BACKGROUND' => '../backgrounds/329.txt',
    'INUSE1' => '1', 'INUSE2' => '0', 'SYNTAX' => '0',
    'CATEGORY' => NULL, 'TEXT_TYPE' => NULL, 'AUTHOR_VERIFY' => NULL,
    'AUTHOR_TYPE' => NULL, 'AUTHOR2_ID' => NULL,
    'AUTHOR2_TYPE' => NULL, 'AUTHOR3_ID' => NULL,
    'AUTHOR3_TYPE' => NULL, 'SHORT_TITLE' => 'Очень_маленький_текст',
    'MAGAZINE_VOLUME' => '1', 'MAGAZINE_SECTION' => 'Рассказы',
    'PAGES' => '1-3', 'CENSORSHIP' => NULL, 'ATTRIBUTIONS' => NULL,
    'STATUS' => NULL, 'IDKEY' => 'жрнчнм-2022-1--329',
    'ORIGIN_TITLE' => 'Очень_маленький_текст'
  ),
  ...
);

```

Пример заполнения данных для таблицы **word**:

```

array(
  array(
    'ID_WORD' => '1341871', 'TEXT_ID' => '329', 'WORD_LENGTH' => '2',
    'CHAPTER_INDEX' => '1', 'PARAGRAPH_INDEX' => '1',
    'SENTENCE_INDEX' => '1', 'WORD_INDEX' => '1',
    'CHDATE' => '2022-12-01_00:00:00', 'WORD' => 'A',
    'DICTWORD_ID' => '218994', 'DICTWORD2_ID' => '0',
    'WORDORDER' => '0', 'WORDNO' => '0'
  ),
  ...
);

```

Пример заполнения данных для таблицы **sys_users**:

```

array(
  '1' => array(

```

```

        'ID' => '1', 'NAME' => '123', 'LOGIN' => '345', 'PASSWORD' => '678',
        'LEVEL' => '3', 'RESEARCHER' => '0'
    )
);

```

Пример заполнения данных для таблицы **textlist_description**:

```

array(
    '1' => array(
        'ID' => '1', 'NAME' => 'Список_1', 'OWNER' => '1',
        'PUBLIC' => '1', 'IS_DELETED' => '0'
    ),
    '2' => array(
        'ID' => '2', 'NAME' => 'Список_2',
        'OWNER' => '1', 'PUBLIC' => '1', 'IS_DELETED' => '0'
    ),
);

```

Пример заполнения данных для таблицы **textlist_items**:

```

array(
    '1' => array('ID' => '1', 'LISTID' => '1', 'TEXTID' => '329'),
    '2' => array('ID' => '2', 'LISTID' => '1', 'TEXTID' => '330'),
    '3' => array('ID' => '3', 'LISTID' => '2', 'TEXTID' => '331'),
    '4' => array('ID' => '4', 'LISTID' => '2', 'TEXTID' => '332'),
    '5' => array('ID' => '5', 'LISTID' => '2', 'TEXTID' => '333')
);

```

Пример заполнения данных для таблицы **entries**:

```

array(
    '218994' => array(
        'ID' => '218994', 'WORD' => 'и',
        'INITIAL_FORM' => 'и', 'PARAM_01' => '11', 'PARAM_02' => '0',
        'PARAM_03' => '0', 'PARAM_04' => '0', 'PARAM_05' => '0',
        'PARAM_06' => '0', 'PARAM_07' => '0', 'PARAM_08' => '0',
        'PARAM_09' => '0', 'PARAM_10' => '1', 'PARAM_11' => '1',
        'PARAM_12' => '0', 'PARAM_13' => '0', 'PARAM_14' => '1',
    )
);

```

```
'PARAM_15' => '1', 'PARAM_16' => '0', 'PARAM_17' => '0',  
'PARAM_18' => '0', 'PARAM_19' => '0', 'PARAM_20' => '0',  
'CURRENT_STATUS' => '1', 'PARAMS_COUNT' => '6', 'MODERN' => 'и'  
) ,  
...  
);
```

Детальный план тестов

Блочное тестирование

ID Теста	Б1
Объект тестирования	Функция получения случайного целого числа из интервала rint()
Цель тестирования	Проверка корректной работы функции при стандартных значениях
Входные данные	$\$start = 1$ $\$end = 3$
Ожидаемый результат	Число на выходе лежит в интервале $[1, 3]$ Число на выходе целое
Тип	Позитивный

ID Теста	Б2
Объект тестирования	Функция получения случайного целого числа из интервала rint()
Цель тестирования	Проверка корректной работы функции при сужении интервала до точки
Входные данные	$\$start = 0$ $\$end = 0$
Ожидаемый результат	Число на выходе равно 0
Тип	Краевой

ID Теста	Б3
Объект тестирования	Функция получения случайного целого числа из интервала rint()
Цель тестирования	Проверка корректной работы функции, когда конец отрезка меньше начала
Входные данные	$\$start = 2$ $\$end = 1$
Ожидаемый результат	Исключение «ValueError»

Тип	Негативный
-----	------------

ID Теста	Б4
Объект тестирования	Функция выбора случайных текстов из списков rText()
Цель тестирования	Проверка корректной работы функции при стандартных значениях
Входные данные	$\$baseTextlistId = 1$ $\$otherTextlistId = 2$
Ожидаемый результат	Полученные номера текстов принадлежат соответствующим спискам
Тип	Позитивный

ID Теста	Б5
Объект тестирования	Функция выбора случайных текстов из списков rText()
Цель тестирования	Проверка корректной работы функции, если на вход подано два одинаковых списка
Входные данные	$\$baseTextlistId = 1$ $\$otherTextlistId = 1$
Ожидаемый результат	Полученные номера текстов не совпадают
Тип	Краевой

ID Теста	Б6
Объект тестирования	Функция получения длины текста в словах getTextLength()
Цель тестирования	Проверка корректной работы функции на занесенном в БД тексте
Входные данные	$\$textId = 333$
Ожидаемый результат	Длина текста в словах равна 1620
Тип	Позитивный

ID Теста	Б7
Объект тестирования	Функция получения длины текста в словах getTextLength()
Цель тестирования	Проверка работы функции при указании несуществующего в БД идентификатора текста
Входные данные	$\$textId = 5555$
Ожидаемый результат	NULL
Тип	Негативный

ID Теста	Б8
Объект тестирования	Функция разбора кода parseCode()
Цель тестирования	Проверка корректной работы функции для кода с одной заменой
Входные данные	$\$code = "A325S112E211B326S102E201"$
Ожидаемый результат	Ожидаемый массив равен фактическому
Тип	Позитивный

ID Теста	Б9
Объект тестирования	Функция разбора кода parseCode()
Цель тестирования	Проверка корректной работы функции для кода с 3 заменами
Входные данные	$\$code = "A325S112E211B326S102E201A325S222E321B326S212E311A325S332E431B326S322E421"$
Ожидаемый результат	Ожидаемый массив равен фактическому
Тип	Позитивный

ID Теста	Б10
Объект тестирования	Функция разбора кода parseCode()
Цель тестирования	Проверка работы функции для пустого кода
Входные данные	$\$code = " "$
Ожидаемый результат	NULL
Тип	Негативный

ID Теста	Б11
Объект тестирования	Функция разбора кода parseCode()
Цель тестирования	Проверка работы функции для некорректного кода
Входные данные	$\$code = "12Aghf"$
Ожидаемый результат	NULL
Тип	Негативный

ID Теста	Б12
Объект тестирования	Функция получения логического массива сдвигов getArrayOfShifts()
Цель тестирования	Проверка корректной работы функции для стандартных значений
Входные данные	$\$insCount = 7$ $\$shiftsCount = 5$
Ожидаемый результат	Непустой массив который содержит 5 истинных значений
Тип	Позитивный

ID Теста	Б13
Объект тестирования	Функция получения логического массива сдвигов getArrayOfShifts()
Цель тестирования	Проверка корректной работы функции для равных значений
Входные данные	$\$insCount = 7$ $\$shiftsCount = 7$
Ожидаемый результат	Непустой массив который содержит 7 истинных значений
Тип	Краевой

ID Теста	Б14
----------	-----

Объект тестирования	Функция получения логического массива последовательности сдвигов getArrayOfShifts()
Цель тестирования	Проверка корректной работы функции, если количество сдвигов равно 0
Входные данные	$\$insCount = 7$ $\$shiftsCount = 0$
Ожидаемый результат	Непустой массив который содержит 7 ложных значений
Тип	Краевой

ID Теста	Б15
Объект тестирования	Функция получения логического массива последовательности сдвигов getArrayOfShifts()
Цель тестирования	Проверка работы функции, если количество сдвигов равно меньше 0
Входные данные	$\$insCount = 7$ $\$shiftsCount = -2$
Ожидаемый результат	Пустой массив
Тип	Негативный

ID Теста	Б16
Объект тестирования	Функция получения логического массива последовательности сдвигов getArrayOfShifts()
Цель тестирования	Проверка работы функции, если количество сдвигов больше количества вставок
Входные данные	$\$insCount = 5$ $\$shiftsCount = 7$
Ожидаемый результат	Пустой массив
Тип	Негативный

ID Теста	Б17
----------	-----

Объект тестирования	Функция получения логического массива последовательности сдвигов getArrayOfShifts()
Цель тестирования	Проверка работы функции, если количество вставок равно 0
Входные данные	$\$insCount = 0$ $\$shiftsCount = 7$
Ожидаемый результат	Пустой массив
Тип	Негативный

ID Теста	Б18
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, и объем вставок меньше вставляемого текста
Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 8$ $\$percentOfInserts = 0.15$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Позитивный

ID Теста	Б19
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, и объем вставок больше вставляемого текста

Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 15$ $\$percentOfInserts = 0.35$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Тип	Позитивный

ID Теста	Б20
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, и размер фрагмента максимален
Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 47$ $\$percentOfInserts = 0.6$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Тип	Краевой

ID Теста	Б21
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, размер фрагмента и процент вставок максимальны

Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 47$ $\$percentOfInserts = 0.95$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Тип	Краевой

ID Теста	Б22
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, размер фрагмента минимален
Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 5$ $\$percentOfInserts = 0.6$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Тип	Краевой

ID Теста	Б23
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст больше вставляемого текста, размер фрагмента минимален, процент вставок максимален

Входные данные	$\$bId = 331$ $\$oId = 330$ $\$baseLength = 236$ $\$otherLength = 47$ $\$fragmentSize = 5$ $\$percentOfInserts = 0.95$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Тип	Краевой

ID Теста	Б24
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст меньше вставляемого текста
Входные данные	$\$bId = 330$ $\$oId = 331$ $\$baseLength = 47$ $\$otherLength = 236$ $\$fragmentSize = 10$ $\$percentOfInserts = 0.4$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Позитивный

ID Теста	Б25
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст меньше вставляемого текста, размер фрагмента и доля вставок максимальна

Входные данные	$\$bId = 330$ $\$oId = 331$ $\$baseLength = 47$ $\$otherLength = 236$ $\$fragmentSize = 45$ $\$percentOfInserts = 0.95$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Краевой

ID Теста	Б26
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()
Цель тестирования	Проверка корректной работы функции, когда исходный текст меньше вставляемого текста, размер фрагмента минимален
Входные данные	$\$bId = 330$ $\$oId = 331$ $\$baseLength = 47$ $\$otherLength = 236$ $\$fragmentSize = 5$ $\$percentOfInserts = 0.34$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Краевой

ID Теста	Б27
Объект тестирования	Функция формирования кода для вставки фрагментов getTextsCode()

Цель тестирования	Проверка корректной работы функции, когда исходный текст меньше вставляемого текста, размер фрагмента минимален, доля вставок максимальна
Входные данные	$\$bId = 330$ $\$oId = 331$ $\$baseLength = 47$ $\$otherLength = 236$ $\$fragmentSize = 5$ $\$percentOfInserts = 0.95$
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Краевой

ID Теста	Б28
Объект тестирования	Функция проверки параметров генерации checkParams()
Цель тестирования	Проверка корректной работы функции для стандартных значений
Входные данные	$\$baseLength = 100$ $\$otherLength = 200$ $\$percentOfInserts = 0.2$ $\$fragmentSize = 5$
Ожидаемый результат	true
Тип	Позитивный

ID Теста	Б29
Объект тестирования	Функция проверки параметров генерации checkParams()
Цель тестирования	Проверка корректной работы функции, когда указанная доля вставок меньше фактической

Входные данные	$\$baseLength = 100$ $\$otherLength = 200$ $\$percentOfInserts = 0.2$ $\$fragmentSize = 40$
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б30
Объект тестирования	Функция проверки параметров генерации check-Params()
Цель тестирования	Проверка корректной работы функции, когда размер фрагмента превышает максимальную долю вставок
Входные данные	$\$baseLength = 100$ $\$otherLength = 200$ $\$percentOfInserts = 0.2$ $\$fragmentSize = 100$
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б31
Объект тестирования	Функция проверки параметров генерации check-Params()
Цель тестирования	Проверка корректной работы функции, когда размер фрагмента превышает размер вставляемого текста
Входные данные	$\$baseLength = 100$ $\$otherLength = 200$ $\$percentOfInserts = 0.2$ $\$fragmentSize = 300$
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б32
----------	-----

Объект тестирования	Функция проверки корректности введенных данных validateForm()
Цель тестирования	Проверка корректной работы функции при стандартных значениях
Входные данные	<i>\$params</i> с корректными значениями
Ожидаемый результат	true
Тип	Позитивный

ID Теста	Б33
Объект тестирования	Функция проверки корректности введенных данных validateForm()
Цель тестирования	Проверка корректной работы функции, когда указан пустой список
Входные данные	<i>\$params</i> с <i>baseTextlistId = ""</i>
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б34
Объект тестирования	Функция проверки корректности введенных данных validateForm()
Цель тестирования	Проверка корректной работы функции, когда не указан номер текста
Входные данные	<i>\$params</i> с <i>baseTextId = ""</i>
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б35
Объект тестирования	Функция проверки корректности введенных данных validateForm()
Цель тестирования	Проверка корректной работы функции, когда не указан процент вставки или он не соответствует интервалу [0.01; 0.95]

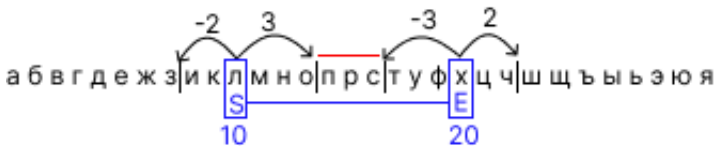
Входные данные	$\$params$ с $percentOfInserts = ""$ или $percentOfInserts = -0.2$ или $percentOfInserts = 1.2$
Ожидаемый результат	false
Тип	Негативный

ID Теста	Б36
Объект тестирования	Функция проверки корректности введенных данных validateForm()
Цель тестирования	Проверка корректной работы функции, когда не указан размер фрагмента или он меньше установленного минимума
Входные данные	$\$params$ с $fragmentSize = ""$ или $fragmentSize = 0$
Ожидаемый результат	false
Тип	Негативный

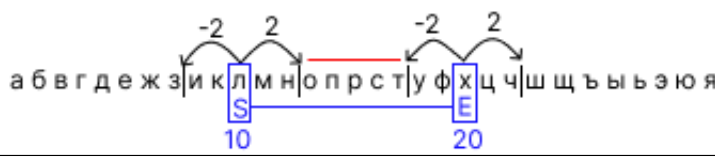
ID Теста	Б37
Объект тестирования	Функция обработки текстов processText()
Цель тестирования	Проверка корректной работы функции при замене 1 фрагмента
Входные данные	$baseId = 329$ $otherId = 329$ $intervals$ (A5-10, B15-20) $showText = false$
Ожидаемый результат	Текст полученный в результате - не пустой, проверка соответствия слов пройдена
Тип	Позитивный

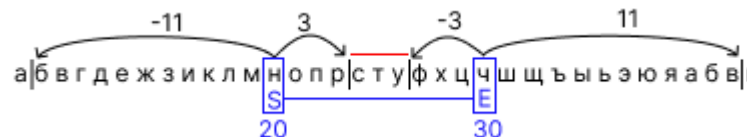
ID Теста	Б38
Объект тестирования	Функция обработки текстов processText()
Цель тестирования	Проверка корректной работы функции при замене 3 фрагментов

Входные данные	$baseId = 329$ $otherId = 329$ $intervals$ (A1-3, 7-9, 13-15, B4-6, 10-12, 16-18) $showText = false$
Ожидаемый результат	Текст полученный в результате - не пустой, проверка соответствия слов пройдена
Тип	Позитивный

ID Теста	Б39
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала вправо, а конца влево
Входные данные	$startPos = 10$ $endPos = 20$ $sLR : 'L' \Rightarrow -2, 'R' \Rightarrow 3$ $eLR : 'L' \Rightarrow -3, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(14, 17) 
Тип	Позитивный

ID Теста	Б40
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала вправо, а конца влево (одинаковые расстояния)

Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 2$ $eLR: 'L' \Rightarrow -2, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(13, 18) 
Тип	Краевой


ID Теста	Б41
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала вправо, а конца влево (другие сдвиги не возможны)
Входные данные	$startPos = 20$ $endPos = 30$ $sLR: 'L' \Rightarrow -11, 'R' \Rightarrow 3$ $eLR: 'L' \Rightarrow -3, 'R' \Rightarrow 11$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(24, 27) 
Тип	Позитивный


ID Теста	Б42
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>

Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала влево, а конца вправо
Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 5$ $eLR: 'L' \Rightarrow -4, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(8, 23)
Тип	Позитивный

ID Теста	Б43
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала влево, а конца вправо (другие сдвиги не возможны)
Входные данные	$startPos = 5$ $endPos = 30$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 11$ $eLR: 'L' \Rightarrow -11, 'R' \Rightarrow 2$ $lEndPos = 1$ $textSize = 100$
Ожидаемый результат	(3, 33)
Тип	Позитивный

ID Теста	Б44
----------	-----

Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала влево, а конца вправо (фрагмент в пределах одного предложения)
Входные данные	$startPos = 10$ $endPos = 12$ $sLR: 'L' \Rightarrow -1, 'R' \Rightarrow 3$ $eLR: 'L' \Rightarrow -3, 'R' \Rightarrow 1$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(9, 14) 
Тип	Позитивный

ID Теста	Б45
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на разные значения)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 11$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(8, 25) 
Тип	Позитивный

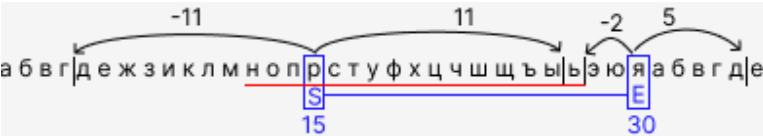
ID Теста	Б46
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на разные значения, другие сдвиги не возможны)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 11$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 11$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(8, 25)
Тип	Позитивный

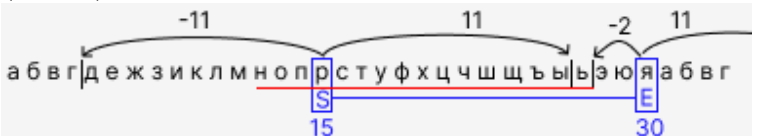
ID Теста	Б47
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на одно значение - по массиву для начала)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow -2, 'R' \Rightarrow 5$ $eLR: 'L' \Rightarrow -11, 'R' \Rightarrow 11$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(8, 29)

Тип	Позитивный
-----	------------

ID Теста	Б48
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на одно значение - по массиву для начала, другие сдвиги не возможны)
Входные данные	$startPos = 5$ $endPos = 30$ $sLR : 'L' \Rightarrow -2, 'R' \Rightarrow 11$ $eLR : 'L' \Rightarrow -11, 'R' \Rightarrow 11$ $lEndPos = -1$ $textSize = 100$
Ожидаемый результат	(3, 29)
Тип	Позитивный

ID Теста	Б49
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на одно значение - по массиву для конца)
Входные данные	$startPos = 15$ $endPos = 30$ $sLR : 'L' \Rightarrow -11, 'R' \Rightarrow 11$ $eLR : 'L' \Rightarrow -2, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$

Ожидаемый результат	(12, 28) 
Тип	Позитивный

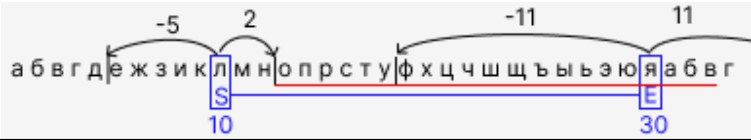
ID Теста	Б50
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца влево (на одно значение - по массиву для конца, другие сдвиги не возможны)
Входные данные	<i>startPos</i> = 15 <i>endPos</i> = 30 <i>sLR</i> : 'L' => -11, 'R' => 11 <i>eLR</i> : 'L' => -2, 'R' => 11 <i>lEndPos</i> = 2 <i>textSize</i> = 100
Ожидаемый результат	(12, 28) 
Тип	Позитивный

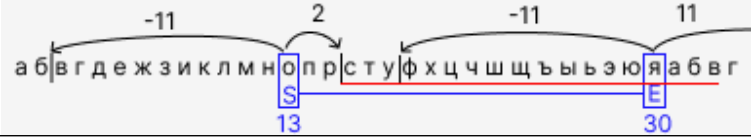
ID Теста	Б51
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на разные значения)

Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow -3, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow -11, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	<p>(15, 33)</p>
Тип	Позитивный


ID Теста	Б52
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на разные значения, другие сдвиги не возможны)
Входные данные	$startPos = 13$ $endPos = 30$ $sLR: 'L' \Rightarrow -11, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow -11, 'R' \Rightarrow 2$ $lEndPos = -1$ $textSize = 100$
Ожидаемый результат	<p>(18, 33)</p>
Тип	Позитивный

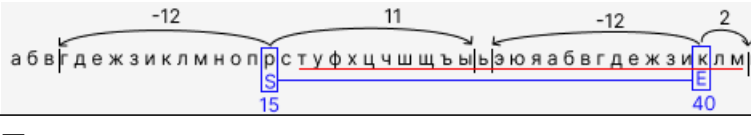
ID Теста	Б53
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>

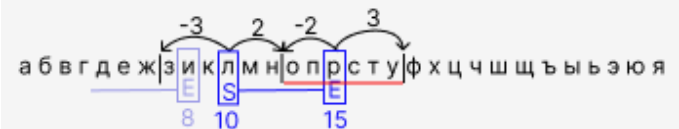
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на одно значение - по массиву для начала)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR : 'L' \Rightarrow -5, 'R' \Rightarrow 2$ $eLR : 'L' \Rightarrow -11, 'R' \Rightarrow 11$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(13, 34) 
Тип	Позитивный

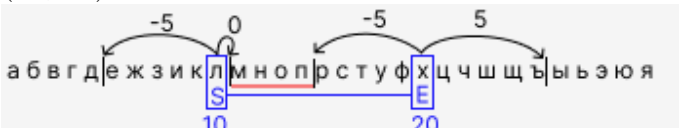
ID Теста	Б54
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на одно значение - по массиву для начала, другие сдвиги не возможны)
Входные данные	$startPos = 13$ $endPos = 30$ $sLR : 'L' \Rightarrow -12, 'R' \Rightarrow 2$ $eLR : 'L' \Rightarrow -11, 'R' \Rightarrow 12$ $lEndPos = 1$ $textSize = 100$
Ожидаемый результат	(16, 34) 
Тип	Позитивный

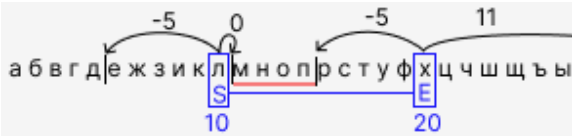
ID Теста	Б55
----------	-----

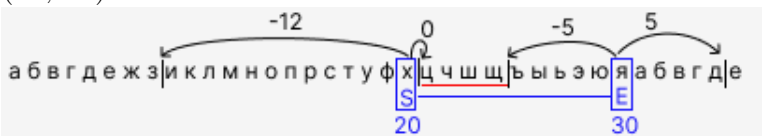
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на одно значение - по массиву для конца)
Входные данные	$startPos = 15$ $endPos = 30$ $sLR: 'L' \Rightarrow -11, 'R' \Rightarrow 11$ $eLR: 'L' \Rightarrow -2, 'R' \Rightarrow 1$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(16, 32) 
Тип	Позитивный

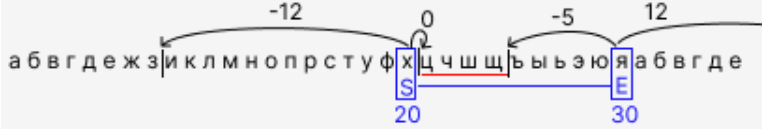
ID Теста	Б56
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (на одно значение - по массиву для конца, другие сдвиги не возможны)
Входные данные	$startPos = 15$ $endPos = 30$ $sLR: 'L' \Rightarrow -12, 'R' \Rightarrow 11$ $eLR: 'L' \Rightarrow -12, 'R' \Rightarrow 2$ $lEndPos = 5$ $textSize = 100$
Ожидаемый результат	(17, 43) 
Тип	Позитивный

ID Теста	Б57
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда требуется сдвиг начала и конца вправо (пересечение с предыдущим интервалом, схождение в 1 точку)
Входные данные	$startPos = 10$ $endPos = 15$ $sLR: 'L' \Rightarrow -3, 'R' \Rightarrow 2$ $eLR: 'L' \Rightarrow -2, 'R' \Rightarrow 3$ $lEndPos = 8$ $textSize = 100$
Ожидаемый результат	(13, 19) 
Тип	Краевой

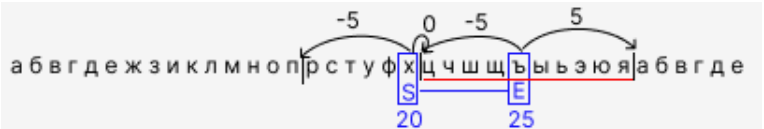
ID Теста	Б58
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(11, 15) 
Тип	Краевой

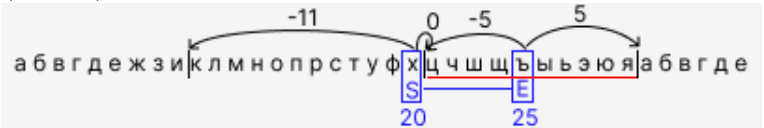
ID Теста	Б59
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 11$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(11, 15) 
Тип	Краевой

ID Теста	Б60
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения
Входные данные	$startPos = 20$ $endPos = 30$ $sLR: 'L' \Rightarrow -12, 'R' \Rightarrow 0$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(21, 25) 
Тип	Краевой

ID Теста	Б61
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения (другие сдвиги не возможны)
Входные данные	$startPos = 20$ $endPos = 30$ $sLR : 'L' \Rightarrow -12, 'R' \Rightarrow 0$ $eLR : 'L' \Rightarrow -5, 'R' \Rightarrow 12$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(21, 25) 
Тип	Краевой

ID Теста	Б62
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения (схождение в одну точку)
Входные данные	$startPos = 20$ $endPos = 25$ $sLR : 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $eLR : 'L' \Rightarrow -5, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$

Ожидаемый результат	(21, 31) 
Тип	Краевой

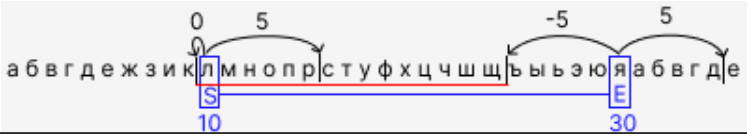
ID Теста	Б63
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения (схождение в одну точку, другие сдвиги не возможны)
Входные данные	<i>startPos</i> = 20 <i>endPos</i> = 25 <i>sLR</i> : 'L' => -11, 'R' => 0 <i>eLR</i> : 'L' => -5, 'R' => 5 <i>lEndPos</i> = 2 <i>textSize</i> = 100
Ожидаемый результат	(21, 31) 
Тип	Краевой

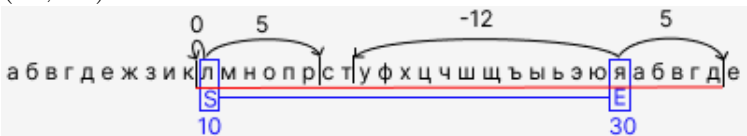
ID Теста	Б64
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения (схождение в одну точку, без сдвига по массивам)

Входные данные	$startPos = 20$ $endPos = 25$ $sLR : 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $eLR : 'L' \Rightarrow -5, 'R' \Rightarrow 12$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	<p>(21, 27)</p>
Тип	Краевой

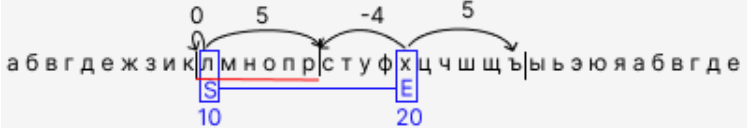
ID Теста	Б65
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения (схождение в одну точку, без сдвига по массивам)
Входные данные	$startPos = 20$ $endPos = 25$ $sLR : 'L' \Rightarrow -11, 'R' \Rightarrow 0$ $eLR : 'L' \Rightarrow -5, 'R' \Rightarrow 12$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	<p>(21, 27)</p>
Тип	Краевой

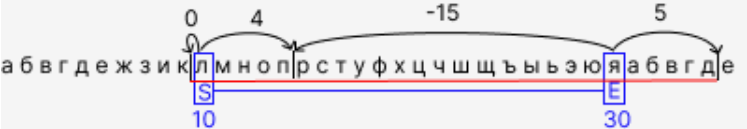
ID Теста	Б66
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()

Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 25) 
Тип	Краевой

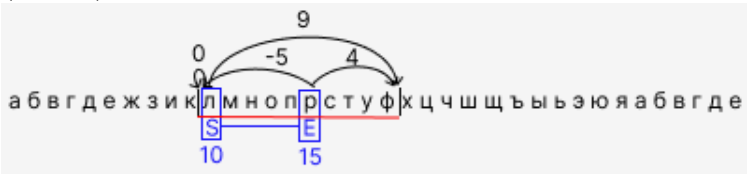
ID Теста	Б67
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $eLR: 'L' \Rightarrow -12, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 36) 
Тип	Краевой

ID Теста	Б68
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()

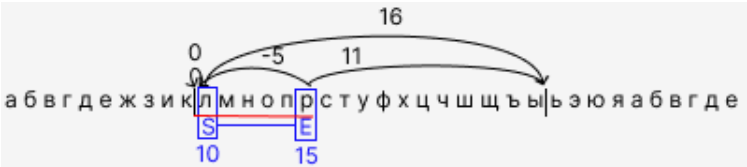
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения (сдвиг в 1 точку - сужение)
Входные данные	$startPos = 10$ $endPos = 20$ $sLR : 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $eLR : 'L' \Rightarrow -4, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 16) 
Тип	Краевой

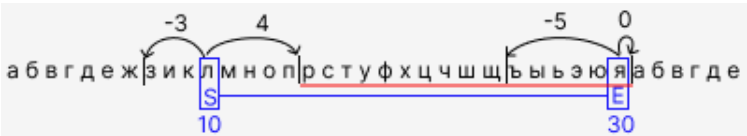
ID Теста	Б69
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения (сдвиг в 1 точку - расширение)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR : 'L' \Rightarrow 0, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow -15, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 36) 
Тип	Краевой

ID Теста	Б70
----------	-----

Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения (фрагмент в пределах одного предложения)
Входные данные	$startPos = 10$ $endPos = 15$ $sLR: 'L' \Rightarrow 0, 'R' \Rightarrow 9$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 4$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 20) 
Тип	Краевой

ID Теста	Б71
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения (фрагмент в пределах одного предложения, сдвиг не возможен)
Входные данные	$startPos = 10$ $endPos = 15$ $sLR: 'L' \Rightarrow 0, 'R' \Rightarrow 16$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 11$ $lEndPos = 2$ $textSize = 100$

Ожидаемый результат	(10, 16) 
Тип	Краевой

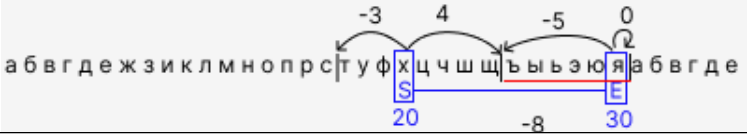
ID Теста	Б72
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения
Входные данные	$startPos = 10$ $endPos = 30$ $sLR: 'L' \Rightarrow -3, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(15, 31) 
Тип	Краевой


ID Теста	Б73
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения (другие сдвиги не возможны)


Входные данные	$startPos = 15$ $endPos = 30$ $sLR: 'L' \Rightarrow -11, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	
Тип	Краевой

ID Теста	Б74
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения (сдвиг в одну точку, другие сдвиги не возможны)
Входные данные	$startPos = 20$ $endPos = 30$ $sLR: 'L' \Rightarrow -11, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	
Тип	Краевой


ID Теста	Б75
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>

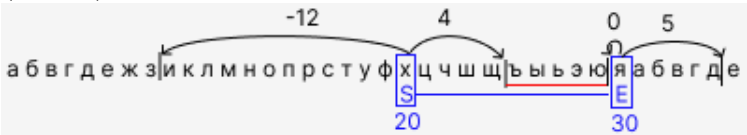
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения (сдвиг в одну точку)
Входные данные	$startPos = 20$ $endPos = 30$ $sLR : 'L' \Rightarrow -3, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(25, 31) 
Тип	Краевой

ID Теста	Б76
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения (фрагмент в пределах одного предложения)
Входные данные	$startPos = 25$ $endPos = 30$ $sLR : 'L' \Rightarrow -3, 'R' \Rightarrow 5$ $eLR : 'L' \Rightarrow -8, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(22, 31) 
Тип	Краевой

ID Теста	Б77
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - последнее слово предложения (фрагмент в пределах одного предложения, сдвиг не возможен)
Входные данные	$startPos = 26$ $endPos = 30$ $sLR : 'L' \Rightarrow -11, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow -15, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(26, 31) 
Тип	Краевой

ID Теста	Б78
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (сужение)
Входные данные	$startPos = 10$ $endPos = 30$ $sLR : 'L' \Rightarrow -5, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$

Ожидаемый результат	(15, 30) 
Тип	Краевой

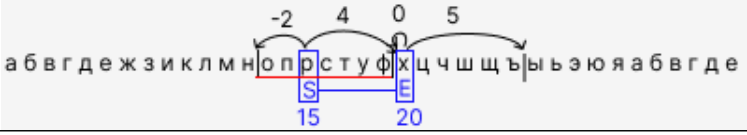
ID Теста	Б79
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (сужение, другие сдвиги не возможны)
Входные данные	<i>startPos</i> = 20 <i>endPos</i> = 30 <i>sLR</i> : 'L' => -12, 'R' => 4 <i>eLR</i> : 'L' => 0, 'R' => 5 <i>lEndPos</i> = 2 <i>textSize</i> = 100
Ожидаемый результат	(25, 30) 
Тип	Краевой

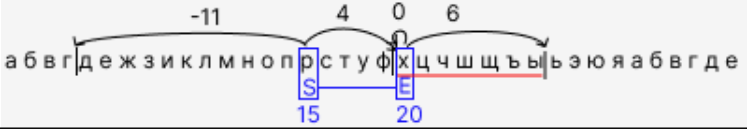
ID Теста	Б80
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (сужение)

Входные данные	$startPos = 20$ $endPos = 30$ $sLR: 'L' \Rightarrow -5, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow 0, 'R' \Rightarrow 12$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	
Тип	Краевой

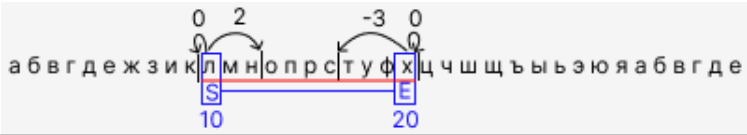
ID Теста	Б81
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (сужение, другие сдвиги не возможны)
Входные данные	$startPos = 20$ $endPos = 30$ $sLR: 'L' \Rightarrow -12, 'R' \Rightarrow 4$ $eLR: 'L' \Rightarrow 0, 'R' \Rightarrow 12$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	
Тип	Краевой

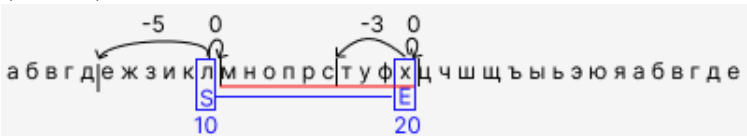
ID Теста	Б82
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений <code>getNewFragmentPositions()</code>

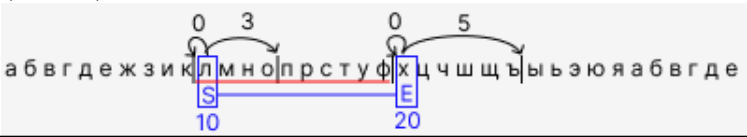
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (фрагмент преобразуется в одно предложение)
Входные данные	$startPos = 15$ $endPos = 20$ $sLR : 'L' \Rightarrow -2, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(13, 20) 
Тип	Краевой

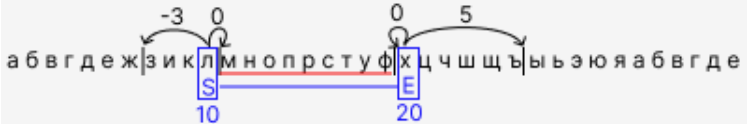
ID Теста	Б83
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда конец фрагмента - первое слово предложения (фрагмент преобразуется в одно предложение, сдвиг вправо)
Входные данные	$startPos = 15$ $endPos = 20$ $sLR : 'L' \Rightarrow -11, 'R' \Rightarrow 4$ $eLR : 'L' \Rightarrow 0, 'R' \Rightarrow 6$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(20, 27) 
Тип	Краевой

ID Теста	Б84
----------	-----

Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения, а конец фрагмента - последнее слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow 0, 'R' \Rightarrow 2$ $eLR: 'L' \Rightarrow -3, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 21) 
Тип	Краевой

ID Теста	Б85
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения, и конец фрагмента - последнее слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR: 'L' \Rightarrow -5, 'R' \Rightarrow 0$ $eLR: 'L' \Rightarrow -3, 'R' \Rightarrow 0$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(11, 21) 
Тип	Краевой

ID Теста	Б86
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - первое слово предложения, и конец фрагмента - первое слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR : 'L' \Rightarrow 0, 'R' \Rightarrow 3$ $eLR : 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(10, 20) 
Тип	Краевой

ID Теста	Б87
Объект тестирования	Функция обработки фрагментов в соответствии с границами предложений getNewFragmentPositions()
Цель тестирования	Проверка корректной работы функции, когда начало фрагмента - последнее слово предложения, а конец фрагмента - первое слово предложения
Входные данные	$startPos = 10$ $endPos = 20$ $sLR : 'L' \Rightarrow -3, 'R' \Rightarrow 0$ $eLR : 'L' \Rightarrow 0, 'R' \Rightarrow 5$ $lEndPos = 2$ $textSize = 100$
Ожидаемый результат	(11, 20) 

Тип	Краевой
-----	---------

Интеграционное тестирование

ID Теста	И1
Объект тестирования	Функция разбора кода parseCode() и функция обработки текстов processText()
Цель тестирования	Проверка совместной работы функций после формирования кода
Входные данные	<i>code = A329S7E12B329S14E19</i> <i>showText = false</i>
Ожидаемый результат	Текст полученный в результате - не пустой, проверка соответствия слов пройдена
Тип	Позитивный

ID Теста	И2
Объект тестирования	Функция разбора кода parseCode() и функция обработки текстов processText()
Цель тестирования	Проверка совместной работы функций после формирования кода при указании неверного кода
Входные данные	<i>code = G329S7E12M329S14E19</i> <i>showText = false</i>
Ожидаемый результат	Текст полученный в результате - пустой, код ответа 400
Тип	Негативный

ID Теста	И3
Объект тестирования	Функция разбора кода parseCode() и функция обработки текстов processText()
Цель тестирования	Проверка совместной работы функций после формирования кода при указании неверного номера текста

Входные данные	<i>code = A444S7E12B444S14E19</i> <i>showText = false</i>
Ожидаемый результат	Текст полученный в результате - пустой
Тип	Негативный

ID Теста	И4
Объект тестирования	Функция формирования кода getTextsCode() и функция разбора кода parseCode()
Цель тестирования	Проверка совместной работы функций перед обработкой текста
Входные данные	<i>\$bId = 329</i> <i>\$oId = 329</i> <i>\$baseLength = 22</i> <i>\$otherLength = 22</i> <i>\$fragmentSize = 3</i> <i>\$percentOfInserts = 0.6</i>
Ожидаемый результат	Проверка для интервалов пройдена (проверка соответствия размера вставок, их количество, отсутствие пересечений)
Тип	Позитивный

ID Теста	И5
Объект тестирования	Функция формирования кода getTextsCode() , функция разбора кода parseCode() и функция обработки текстов processText()
Цель тестирования	Проверка совместной работы функций от начала формирования кода до печати результата генерации

Входные данные	$\$bId = 329$ $\$oId = 329$ $\$baseLength = 22$ $\$otherLength = 22$ $\$fragmentSize = 5$ $\$percentOfInserts = 0.6$ $\$showText = false$
Ожидаемый результат	Выходной текст не пустой, размер текста 22 слова. Проверка вставок по коду и исходному тексту пройдена.
Тип	Позитивный

Аттестационное тестирование

ID Теста	A1
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка доступности страницы генератора с различных браузеров
Входные данные	Ввод url-адреса в адресную строку браузера: http://smalt.karelia.ru/shower/research/text-generator.php
Ожидаемый результат	Отображение страницы генератора во всех популярных браузерах
Тип	Позитивный

ID Теста	A2
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения кода генерации для случайных текстов
Входные данные	Заранее подготовлены тестовые списки текстов. Указаны параметры формы (пункт «Случайные тексты» включен)

Ожидаемый результат	Отображение библиографического описания текстов (с ссылками на полный текст), их соответствие указанным спискам. Отображение кода-ссылки с кнопкой «Копировать код».
Тип	Позитивный

ID Теста	A3
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения кода генерации для конкретных текстов
Входные данные	Заранее подготовлены тестовые списки текстов. Указаны параметры формы (пункт «Случайные тексты» выключен)
Ожидаемый результат	Отображение библиографического описания текстов (с ссылками на полный текст), их соответствие указанным текстам. Отображение кода-ссылки с кнопкой «Копировать код».
Тип	Позитивный

ID Теста	A4
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения код+текст для случайных текстов
Входные данные	Заранее подготовлены тестовые списки текстов. Указаны параметры формы (пункт «Случайные тексты» включен)
Ожидаемый результат	Отображение библиографического описания текстов (с ссылками на полный текст), их соответствие указанным спискам. Отображение кода-ссылки с кнопкой «Копировать код». Отображение полученного текста с цветовым выделением вставок и кнопкой «Экспорт».

Тип	Позитивный
-----	------------

ID Теста	A5
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения код+текст для конкретных текстов
Входные данные	Заранее подготовлены тестовые списки текстов. Указаны параметры формы (пункт «Случайные тексты» выключен)
Ожидаемый результат	Отображение библиографического описания текстов (с ссылками на полный текст), их соответствие указанным текстам. Отображение кода-ссылки с кнопкой «Копировать код». Отображение полученного текста с цветовым выделением вставок и кнопкой «Экспорт».
Тип	Позитивный

ID Теста	A6
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка копирования кода в буфер
Входные данные	После генерации кода нажатие на кнопку «Копировать код».
Ожидаемый результат	Всплывающее уведомление об успешности копирования. При вставке в текстовый редактор действительно отображается код.
Тип	Позитивный

ID Теста	A7
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка экспорта полученного текста
Входные данные	После генерации текста нажатие на кнопку «Экспорт».

Ожидаемый результат	Загружен текстовый файл, соответствующий отображаемому на странице преобразованному тексту
Тип	Позитивный

ID Теста	A8
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка перехода по коду-ссылке на полученный текст
Входные данные	После генерации кода нажатие на ссылку прикрепленную к коду.
Ожидаемый результат	Открылась страница с сгенерированным текстом.
Тип	Позитивный

ID Теста	A9
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения ошибки при вводе доли вставок не соответствующей интервалу [0.01; 0.95]
Входные данные	Доля вставок <0.01 или >0.95 или не заполнена.
Ожидаемый результат	Вывод ошибки «Укажите долю вставок в диапазоне 0.01 - 0.95»
Тип	Негативный

ID Теста	A10
Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения ошибки при неуказанном размере фрагмента или размере <5
Входные данные	Размер фрагмента <5 или не заполнен.
Ожидаемый результат	Вывод ошибки «Минимальный размер заменяемого фрагмента 5»
Тип	Негативный

ID Теста	A11
----------	-----

Объект тестирования	Модуль генератора текстов в целом
Цель тестирования	Проверка отображения ошибок при указании неверного размера фрагмента (3 случая)
Входные данные	<p>Пример:</p> <ol style="list-style-type: none"> 1. $L1 = 500$ $POI = 0.1$ (т.е 50 слов) $FS = 70$; 2. $L1 = 200$ $L2 = 250$ $FS = 195$; 3. $L1 = 200$ $L2 = 150$ $FS = 195$;
Ожидаемый результат	<ol style="list-style-type: none"> 1. Вывод ошибки «Указанная доля вставок меньше фактической. Укажите $\geq \dots$»; 2. Вывод ошибки «Максимальная доля вставок 0.95, размер фрагмента не должен превышать это значение. (...)»; 3. Вывод ошибки «Размер фрагмента не должен превышать размер вставляемого текста. (...)»;
Тип	Негативный

Нагрузочное тестирование

ID Теста	Н1
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 20.

Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.
---------------------	---

ID Теста	Н2
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 50.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

ID Теста	Н3
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 100.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

ID Теста	Н4
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок

Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 500.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

ID Теста	Н5
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 1000.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

ID Теста	Н6
Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 1500.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

ID Теста	Н7
----------	----

Цель тестирования	Проверка работоспособности генератора на достаточно больших текстах с разным количеством вставок
Входные данные	В БД загружено два больших текста. Размер фрагмента 20 слов. Доля подбирается так, чтобы количество вставок соответствовало 2000.
Ожидаемый результат	Генератор выводит результат генерации. Время ожидания не более 10 сек. Нет никаких ошибок и предупреждений.

Примеры реализации тестов

```

public function testProcessText_Correct1Ins()
{
    $smalt = new SmaltTextGenerator();
    $smalt->switchPDO();
    $intervals = array(
        'A' => array(
            0 => array(
                'S' => 5,
                'E' => 10
            )
        ),
        'B' => array(
            0 => array(
                'S' => 15,
                'E' => 20,
            )
        )
    );
    $smalt->processText( textIdBase: 329, textIdOther: 329, $intervals, showText: false);
    $this->assertNotEmpty($smalt->textData);
    $words = preg_split( pattern: '/\s+/', $smalt->textData, limit: -1, flags: PREG_SPLIT_NO_EMPTY);
    for($i = 5; $i <=10; $i++)
        $this->assertTrue( condition: $words[$i] == $words[$i+10]);
}

```

Рис. 2: Пример блочного теста


```

public function testIntegrationGTCandPC()
{
    $smalt = new SmaltTextGenerator();
    $smalt->switchPDO();
    $code = $smalt->getTextsCode( bld: 329, old: 329, baseTextLength: 22, otherTextLength: 22, fragmentSize: 3, percentOfinserts: 0.6);
    $intervals = $smalt->parseCode($code);
    $this->assertIntervals($intervals, insCount: 4, intervalSize: 3, checkSecondIntShifts: true);
}

public function assertIntervals($fragments, $insCount, $intervalSize, $checkSecondIntShifts){
    $this->assertCount($insCount, $fragments['intervals']['A']);
    $this->assertCount($insCount, $fragments['intervals']['B']);
    $lastEnd = -1;
    foreach($fragments['intervals']['A'] as $intervalsA) {
        $this->assertEquals($intervalSize, actual: $intervalsA['E'] - $intervalsA['S'] + 1);
        $this->assertTrue ( condition: $intervalsA['S'] > $lastEnd);
        $lastEnd = $intervalsA['E'];
    }
    if($checkSecondIntShifts) $lastEnd = -1;
    foreach($fragments['intervals']['B'] as $intervalsB) {
        $this->assertEquals($intervalSize, actual: $intervalsB['E'] - $intervalsB['S'] + 1);
        if($checkSecondIntShifts) {
            $this->assertTrue ( condition: $intervalsB['S'] > $lastEnd);
            $lastEnd = $intervalsB['E'];
        }
    }
}
}

```

Рис. 3: Пример интеграционного теста

Журнал тестирования

Журнал блочного тестирования

ID Теста	Дата	Результат	Отчет
Б1	12.12.2022	Пройден	—
Б2	12.12.2022	Пройден	—
Б3	12.12.2022	Пройден	—
Б4	12.12.2022	Пройден	—
Б5	12.12.2022	Пройден	—
Б6	12.12.2022	Пройден	—
Б7	12.12.2022	Пройден	—
Б8	12.12.2022	Пройден	—
Б9	12.12.2022	Пройден	—
Б10	12.12.2022	Пройден	—
Б11	12.12.2022	Пройден	—
Б12	12.12.2022	Пройден	—

ID Теста	Дата	Результат	Отчет
Б13	12.12.2022	Пройден	—
Б14	12.12.2022	Пройден	—
Б15	12.12.2022	Пройден	—
Б16	12.12.2022	Пройден	—
Б17	12.12.2022	Пройден	—
Б18	12.12.2022	Пройден	—
Б19	12.12.2022	Пройден	—
Б20	07.12.2022	Ошибка	Отчет об ошибке №1
Б21	12.12.2022	Пройден	—
Б22	07.12.2022	Ошибка	Отчет об ошибке №2
Б23	12.12.2022	Пройден	—
Б24	12.12.2022	Пройден	—
Б25	12.12.2022	Пройден	—
Б26	12.12.2022	Пройден	—
Б27	12.12.2022	Пройден	—
Б28	12.12.2022	Пройден	—
Б29	12.12.2022	Пройден	—
Б30	12.12.2022	Пройден	—
Б31	12.12.2022	Пройден	—
Б32	12.12.2022	Пройден	—
Б33	12.12.2022	Пройден	—
Б34	12.12.2022	Пройден	—
Б35	12.12.2022	Пройден	—
Б36	12.12.2022	Пройден	—
Б37	12.12.2022	Пройден	—
Б38	12.12.2022	Пройден	—
Б39	14.12.2022	Ошибка	Отчет об ошибке №3
Б40	14.12.2022	Пройден	—
Б41	14.12.2022	Пройден	—
Б42	14.12.2022	Пройден	—
Б43	14.12.2022	Пройден	—

ID Теста	Дата	Результат	Отчет
Б44	14.12.2022	Пройден	—
Б45	14.12.2022	Ошибка	Отчет об ошибке №4
Б46	14.12.2022	Пройден	—
Б47	14.12.2022	Пройден	—
Б48	14.12.2022	Пройден	—
Б49	14.12.2022	Пройден	—
Б50	14.12.2022	Пройден	—
Б51	14.12.2022	Ошибка	Отчет об ошибке №5
Б52	14.12.2022	Пройден	—
Б53	14.12.2022	Пройден	—
Б54	14.12.2022	Пройден	—
Б55	14.12.2022	Пройден	—
Б56	14.12.2022	Пройден	—
Б57	14.12.2022	Пройден	—
Б58	14.12.2022	Пройден	—
Б59	14.12.2022	Пройден	—
Б60	14.12.2022	Пройден	—
Б61	14.12.2022	Пройден	—
Б62	14.12.2022	Пройден	—
Б63	14.12.2022	Пройден	—
Б64	14.12.2022	Пройден	—
Б65	14.12.2022	Пройден	—
Б66	14.12.2022	Пройден	—
Б67	14.12.2022	Пройден	—
Б68	14.12.2022	Пройден	—
Б69	14.12.2022	Пройден	—
Б70	14.12.2022	Пройден	—
Б71	14.12.2022	Пройден	—
Б72	14.12.2022	Ошибка	Отчет об ошибке №6
Б73	14.12.2022	Пройден	—
Б74	14.12.2022	Пройден	—

ID Теста	Дата	Результат	Отчет
Б75	14.12.2022	Ошибка	Отчет об ошибке №7
Б76	14.12.2022	Пройден	—
Б77	14.12.2022	Ошибка	Отчет об ошибке №8
Б78	14.12.2022	Пройден	—
Б79	14.12.2022	Пройден	—
Б80	14.12.2022	Пройден	—
Б81	14.12.2022	Пройден	—
Б82	14.12.2022	Пройден	—
Б83	14.12.2022	Пройден	—
Б84	14.12.2022	Пройден	—
Б85	14.12.2022	Пройден	—
Б86	14.12.2022	Пройден	—
Б87	14.12.2022	Пройден	—

Журнал интеграционного тестирования

ID Теста	Дата	Результат	Отчет
И1	12.12.2022	Пройден	—
И2	12.12.2022	Пройден	—
И3	12.12.2022	Пройден	—
И4	12.12.2022	Пройден	—
И5	12.12.2022	Пройден	—

Журнал аттестационного тестирования

ID Теста	Дата	Результат	Отчет
А1	12.12.2022	Пройден	—
А2	12.12.2022	Пройден	—
А3	12.12.2022	Пройден	—
А4	12.12.2022	Пройден	—
А5	12.12.2022	Пройден	—
А6	12.12.2022	Пройден	—

ID Теста	Дата	Результат	Отчет
A7	12.12.2022	Пройден	—
A8	12.12.2022	Пройден	—
A9	12.12.2022	Пройден	—
A10	12.12.2022	Пройден	—
A11	12.12.2022	Пройден	—

Журнал нагрузочного тестирования

ID Теста	Дата	Результат	Отчет
H1	12.12.2022	Пройден	—
H2	12.12.2022	Пройден	—
H3	12.12.2022	Пройден	—
H4	12.12.2022	Пройден	—
H5	12.12.2022	Пройден	—
H6	12.12.2022	Пройден	—
H7	12.12.2022	Пройден	—

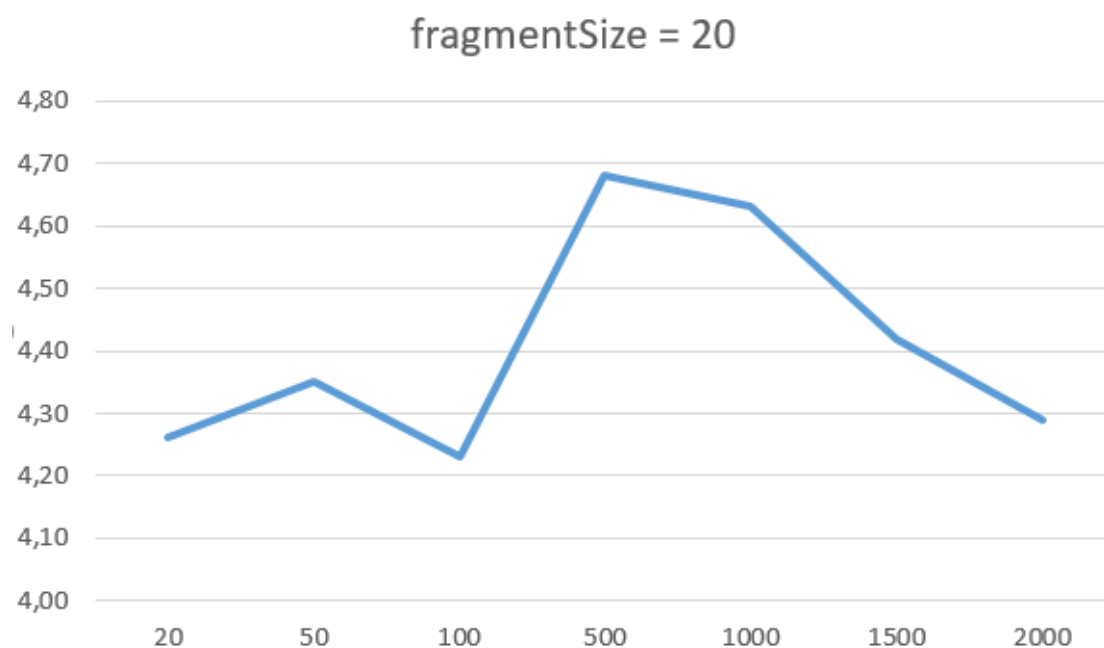


Рис. 4: Результаты нагрузочного тестирования

Журнал найденных ошибок

№ Отчета об ошибке	1
Дата составления отчета	07.12.2022
Номер теста	Б20
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Фактический результат	Несоответствие количества вставок и размеров фрагмента. Первая вставка обозначена правильно, вторая - остаток до конца текста
Воспроизводимость	Всегда
Приоритет	Средний
Статус	Решено

Таблица 116: Отчет об ошибке №1

№ Отчета об ошибке	2
Дата составления отчета	07.12.2022
Номер теста	Б22
Ожидаемый результат	Проверка для кода пройдена успешно (проверка соответствия размера вставок, их количество)
Фактический результат	Размер последнего фрагмента не всегда правильный.
Воспроизводимость	Периодическая
Приоритет	Маленький
Статус	Решено

Таблица 117: Отчет об ошибке №2

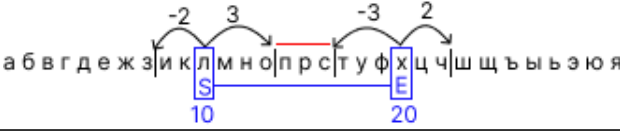
№ Отчета об ошибке	3
Дата составления отчета	14.12.2022
Номер теста	Б39
Ожидаемый результат	(14, 17) 
Фактический результат	(8, 23)
Воспроизводимость	Всегда
Приоритет	Средний
Статус	Не решено

Таблица 118: Отчет об ошибке №3

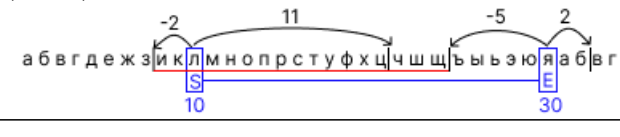
№ Отчета об ошибке	4
Дата составления отчета	14.12.2022
Номер теста	Б45
Ожидаемый результат	(8, 25) 
Фактический результат	(8, 33)
Воспроизводимость	Всегда
Приоритет	Средний
Статус	Не решено

Таблица 119: Отчет об ошибке №4

№ Отчета об ошибке	5
Дата составления отчета	14.12.2022
Номер теста	Б51

Таблица 120: Отчет об ошибке №5

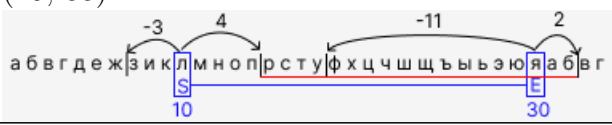
Ожидаемый результат	(15, 33) 
Фактический результат	(7, 33)
Воспроизводимость	Всегда
Приоритет	Средний
Статус	Не решено

Таблица 120: Отчет об ошибке №5

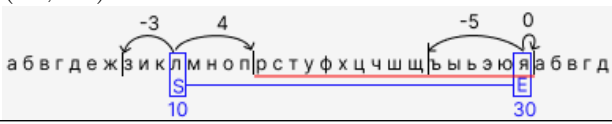
№ Отчета об ошибке	6
Дата составления отчета	14.12.2022
Номер теста	Б72
Ожидаемый результат	(15, 31) 
Фактический результат	(7, 31)
Воспроизводимость	Всегда
Приоритет	Низкий
Статус	Не решено

Таблица 121: Отчет об ошибке №6

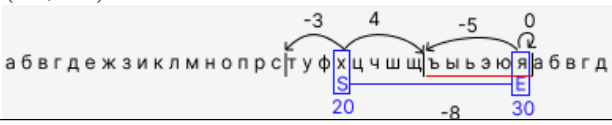
№ Отчета об ошибке	7
Дата составления отчета	14.12.2022
Номер теста	Б75
Ожидаемый результат	(25, 31) 
Фактический результат	(17, 31)
Воспроизводимость	Всегда

Таблица 122: Отчет об ошибке №7

Приоритет	Низкий
Статус	Не решено

Таблица 122: Отчет об ошибке №7


№ Отчета об ошибке	8
Дата составления отчета	14.12.2022
Номер теста	Б77
Ожидаемый результат	(26, 31) 
Фактический результат	(31, 36)
Воспроизводимость	Всегда
Приоритет	Низкий
Статус	Не решено

Таблица 123: Отчет об ошибке №8

Результаты

В рамках дисциплины «Верификация программного обеспечения» были протестированы функциональные возможности модуля информационной системы «СМАЛТ», отвечающего за генерацию текстов для проверки алгоритмов атрибуции.

На этапах модульного и интеграционного тестирования, проводимого посредством автоматизированного тестирования, не было выявлено критических ошибок в функционировании рассмотренных методов системы, часть некритических ошибок устранены (для указанных функциональностей системы), другая часть требует дальнейшей доработки алгоритма для полностью реализованной функциональности (сдвиги границ фрагментов). На этапе аттестационного тестирования, проводимого посредством ручного тестирования, критические ошибки также выявлены не были. На этапе нагрузочного тестирования система показала хорошие результаты по времени: генерация текста занимала менее 5 сек. при больших объемах вставки, хотя ожидалось увеличение времени вплоть до 10 сек.

Таким образом, информационная система «Статистические методы анализа литературного текста» успешно прошла тестирование.