

**Сивков С.С., Емельянов Д.М., Петрова Н.К., Хамитов Р.М.**  
Кафедра Информатики и информационных управляющих систем  
Казанский государственный энергетический университет

## **РАЗРАБОТКА ПРОЕКТА ТЕСТИРОВАНИЯ СТУДЕНТОВ СРЕДСТВАМИ C++ С ПРИМЕНЕНИЕМ БИНАРНЫХ ФАЙЛОВ**

### *Аннотация*

*В статье представлен программный проект по реализации тестирования студентов по учебному предмету. В качестве предмета выбрана дисциплина «Программирование на C++». Для разработки рассмотрена ограниченная модель вопросов: программа предоставляет и обрабатывает вопросы только двух типов – открытые и закрытые. Особенностью построения проекта является разработка базы вопросов в виде бинарных файлов, что, во-первых, экономит место на дисковом пространстве и, во-вторых, обеспечивает быстрый доступ к вопросу и вариантам ответов на него. В курсовой работе разработан алгоритм и соответствующая ему программа, которая позволяет создавать и открывать базы тестирования студентов на любые темы. Программа написана на языке C++ в среде разработки CLion и работает на версиях Windows 7-10.*

**Ключевые слова:** программирование, тестирование, бинарные файлы  
**Keywords:** programming, c++, testing, binary files

### **Введение**

Безусловно, в любом современном университете функционирует электронная образовательная среда [1,15], включающая средства тестирования и анализа ответов студентов. Однако бывают случаи, когда облачные хранилища или сайт учебного заведения по каким-либо причинам недоступны в нужный момент. Для таких ситуаций полезно иметь на собственном компьютере программу-тестировщик, позволяющую быстро и эффективно проверить знания студентов.

Задачей данной исследовательской работы является разработка программы тестирования студентов по курсу «Программирование на языке C++». Это актуальная для университета программа позволяет преподавателю автономно проводить оценку знаний студента по дисциплине. Пользователь может дополнять базу вопросов или убирать ненужные задания. Программа автоматически оценивает ответы студентов, экономя время преподавателя.

Приложение должно реализовывать следующие задачи.

Создавать тестовый материал для проверки знаний студентов;

Подразумевать два вида вопросов (открытый и закрытый, см. пункт «Контрольный пример»);

Иметь возможность запускать уже существующие тесты, созданные в этой же программе;

Выводить статистику ответов по завершении теста.

Поскольку размещение программы с базой вопросов и ответов предусмотрено на компьютере пользователя, очень важно, чтобы эта информация занимала как можно меньше места на дисках. С этой целью в работе рассмотрена и реализована возможность упаковки информации в бинарные файлы [4,115].

### **Контрольный пример**

Разрабатываемое приложение позволит создавать свои собственные тесты и запускать уже существующие.

Для формирования алгоритма и тестирования создаваемой программы построим контрольный пример, в котором рассмотрим детали процесса тестирования, формализуем задачу для перевода её на язык компьютера.

Тесты хранятся на жёстком диске компьютера в двоичном формате.

При чтении теста программа:

1. Формирует номер вопроса (например, вопрос №2);
2. Выбирает из базы вопрос №2;

Для примера предлагается рассмотреть следующую базу тестирования (базу вопросов теста и ответов к ним) [5, 56]:

№	Текст вопроса	Тип	Варианты ответа ( <b>жирным</b> выделен правильный)
1	Что такое компилятор?	Закр.	<p>1. Компилятор - программа, которая выполняет построчный анализ, обработку и выполнение исходного кода программы или запроса</p> <p><b>2. Компилятор - программа, которая переводит текст программы, написанный на языке программирования, в набор машинных кодов</b></p> <p>3. Компилятор - программа, которая производит компоновку: принимает на вход один или несколько объектных модулей и собирает по ним исполнимый модуль</p> <p>4. Компилятор - спецификация языка C++</p>
2	Кто является создателем языка C++?	Закр.	<p>1. Чарльз Бэббидж</p> <p>2. Деннис Ритчи</p> <p>3. Брайан Керниган</p> <p><b>4. Бьярне Страуструп</b></p>
3	Назовите любую среду разработки на C++	Откр.	<ul style="list-style-type: none"> <li>• Visual Studio</li> <li>• CLion</li> <li>• Dev-C++</li> <li>• Code::Blocks</li> </ul> <p><i>Примечание: все четыре варианта ответа являются правильными, правильный ответ будет засчитан при вводе любого из них.</i></p>

3. Определяет тип вопроса с помощью поля-перечисления «type» объекта вопроса, либо Type::CLOSE (в памяти имеет значение 0), либо Type::OPEN (в памяти имеет значение 1).

Закрытый вопрос – вопрос, на который существует только один вариант ответа. Таким образом, вопрос «Что такое GCC?» может иметь четыре варианта ответа: «Компилятор языка C++», «Справка по языку C++», «Интерпретатор языка C++» и «Среда разработки для языка C++». В закрытом вопросе может быть только один правильный ответ, в данном случае, «Компилятор языка C++». При его выборе будет засчитан правильный ответ на вопрос, при выборе любого другого будет засчитан неправильный ответ. При этом все четыре варианта ответов будут выведены на экран и от пользователя ожидается ввод с клавиатуры либо номера варианта ответа, либо сам ответ [2,133].

Открытый вопрос – вопрос, на который существует один или несколько правильных ответов, которые не будут выведены пользователю. Он должен ввести

ответ с клавиатуры. Например, в вопросе «Назовите любой знаковый целочисленный тип данных в C++». Правильными ответами в таком случае будут: «int», «long», «short», «char». При вводе любого из этих ответов будет засчитан правильный ответ на вопрос, при вводе другого ответа, не входящего в обозначенный список, будет засчитан неправильный ответ. При этом варианты ответов не выводятся на экран [6, 328].

Вызывает нужную функцию, которая обрабатывает вопрос – выводит текст вопроса и, если вопрос является закрытым, варианты ответа:

Вопрос №2: Кто является создателем языка C++?

Варианты ответа:

1. Чарльз Бэббидж
2. Деннис Ритчи
3. Брайан Керниган
4. Бьярне Страуструп

Ожидается номер ответа пользователем (например, ответ №1).

Если пользователь отвечает правильно, то в переменную «rightAnswers» добавляется единица, если неправильно, то в переменную «wrongAnswers» добавляется ноль.

Ответ: 4

Ответ правильный!

Осуществляется проверка конца процесса теста:

- Если номер\_вопроса < количество\_вопросов, то увеличиваем номер\_вопроса на 1, и переходим к пункту 1.

- Иначе тест завершается.

Вывод результатов тестирования (далее – статистика) – количество правильно отвеченных вопросов, количество неправильно отвеченных вопросов и их соответствующие процентные соотношения:

Тест закончен!

Всего вопросов: 3

Правильно отвеченных вопросов: 3 (100%)

Неправильно отвеченных вопросов: 0 (0%)

Конец программы.

Формат файла представлен на рис. 1.

Первые четыре байта файла занимает «header» (заголовок) файла – комбинация байтов, которые позволяют программе определить, что открывается файл, подходящий для данного приложения. В данном случае это байты 51 55 49 5A, или же «QUIZ» в ASCII. Следующие два байта (по адресам 04 и 05) занимает количество вопросов в тесте. В данном случае в тесте три вопроса, следовательно, там хранятся байты 00 и 03.

Следом до конца файла в памяти располагаются вопросы. Вопрос состоит из количества ответов (следующие два байта), в два старших бита которых также размещены тип вопроса (0 – закрытый или 1 – открытый) и регистрозависимость ответов (0 – ответ принимается строго как он находится в памяти, 1 – разрешены вариации в регистре, например «ответ», «ОТВЕТ», «Ответ», «ОтВеТ» являются вариациями одного ответа).

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	B1	55	49	5A	00	03	00	04	D7	F2	EE	20	F2	E0	EA	EE	QUIZ...Что такое
00000010	E5	20	EA	EE	EC	EF	E8	EB	FF	F2	EE	F0	3F	00	CA	EE	е компилятор?.Ко
00000020	EC	EF	E8	EB	FF	F2	EE	F0	20	2D	20	EF	F0	EE	E3	F0	мпиллятор - прогр
00000030	E0	EC	EC	E0	2C	20	EA	EE	F2	EE	F0	E0	FF	20	E2	FB	ама, которая вы
00000040	EF	EE	EB	ED	FF	E5	F2	20	EF	EE	F1	F2	F0	EE	F7	ED	полняет построчи
00000050	FB	E9	20	E0	ED	E0	EB	E8	E7	2C	20	EE	E1	F0	E0	E1	ый анализ, обраб
00000060	EE	F2	EA	F3	20	E8	20	E2	FB	EF	EE	EB	ED	E5	ED	E8	отку и выполнени
00000070	E5	20	E8	F1	F5	EE	E4	ED	EE	E3	EE	20	EA	EE	E4	E0	е исходного кода
00000080	20	EF	F0	EE	E3	F0	E0	EC	EC	FB	20	E8	EB	E8	20	E7	программы или з
00000090	E0	EF	F0	EE	F1	E0	00	CA	EE	EC	EF	E8	EB	FF	F2	EE	апроса.Компилято
000000A0	F0	20	2D	20	EF	F0	EE	E3	F0	E0	EC	EC	E0	2C	20	EA	р - программа, к
000000B0	EE	F2	EE	F0	E0	FF	20	EF	E5	F0	E5	E2	EE	E4	E8	F2	оторая переводит
000000C0	20	F2	E5	EA	F1	F2	20	EF	F0	EE	E3	F0	E0	EC	EC	FB	текст программы
000000D0	2C	20	ED	E0	EF	E8	F1	E0	ED	ED	FB	E9	20	ED	E0	20	, написанный на
000000E0	FF	E7	FB	EA	E5	20	EF	F0	EE	E3	F0	E0	EC	EC	E8	F0	языке программир
000000F0	EE	E2	E0	ED	E8	FF	2C	20	E2	20	ED	E0	E1	EE	F0	20	ования, в набор
00000100	EC	E0	F8	E8	ED	ED	FB	F5	20	EA	EE	E4	EE	E2	00	CA	машинных кодов.К
00000110	EE	EC	EF	E8	EB	FF	F2	EE	F0	20	2D	20	EF	F0	EE	E3	омпилятор - прог
00000120	F0	E0	EC	EC	E0	2C	20	EA	EE	F2	EE	F0	E0	FF	20	EF	рамма, которая п
00000130	F0	EE	E8	E7	E2	EE	E4	E8	F2	20	EA	EE	EC	EF	EE	ED	роизводит компо
00000140	EE	E2	EA	F3	3A	20	EF	F0	E8	ED	E8	EC	E0	E5	F2	20	овку: принимает
00000150	ED	E0	20	E2	F5	EE	E4	20	EE	E4	E8	ED	20	E8	EB	E8	на вход один или
00000160	20	ED	E5	F1	EA	EE	EB	FC	EA	EE	20	EE	E1	FA	E5	EA	несколько объек
00000170	F2	ED	FB	F5	20	EC	EE	E4	F3	EB	E5	E9	20	E8	20	F1	тных модулей и с
00000180	EE	E1	E8	F0	E0	E5	F2	20	EF	EE	20	ED	E8	EC	20	E8	обирает по ним и
00000190	F1	EF	EE	EB	ED	E8	EC	FB	E9	20	EC	EE	E4	F3	EB	FC	сполняемый модуль
000001A0	00	CA	EE	EC	EF	E8	EB	FF	F2	EE	F0	20	2D	20	F1	EF	.Компилятор - сп
000001B0	E5	F6	E8	F4	E8	EA	E0	F6	E8	FF	20	FF	E7	FB	EA	E0	ецификация языка
000001C0	20	43	2B	2B	00	00	01	00	04	CA	F2	EE	20	FF	E2	EB	C++....Кто явл
000001D0	FF	E5	F2	F1	FF	20	F1	EE	E7	E4	E0	F2	E5	EB	E5	EC	яется создателем
000001E0	20	FF	E7	FB	EA	E0	20	D1	2B	2B	3F	00	D7	E0	F0	EB	языка C++?.Чарл
000001F0	FC	E7	20	C1	FD	E1	E1	E8	E4	E6	00	C4	E5	ED	ED	E8	ьз Вэббидж.Денни
00000200	F1	20	D0	E8	F2	F7	E8	00	C1	F0	E0	E9	E0	ED	20	CA	с Ритчи.Брайан К
00000210	E5	F0	ED	E8	E3	E0	ED	00	C1	FC	FF	F0	ED	E5	20	D1	ерниган.Бьярне С
00000220	F2	F0	E0	F3	F1	F2	F0	F3	EF	00	00	03	80	04	CD	E0	трауstrup...Ъ.На
00000230	E7	EE	E2	E8	F2	E5	20	EB	FE	E1	F3	FE	20	F1	F0	E5	зовите любую сре
00000240	E4	F3	20	F0	E0	E7	F0	E0	E1	EE	F2	EA	E8	20	ED	E0	ду разработки на
00000250	20	D1	2B	2B	00	56	69	73	75	61	6C	20	53	74	75	64	C++..Visual Stud
00000260	69	6F	00	43	4C	69	6F	6E	00	44	65	76	2D	43	2B	2B	io..CLion.Dev-C++
00000270	00	43	6F	64	65	3A	3A	42	6C	6F	63	6B	73	00			.Code::Blocks.

Рис.1. Формат файла.

После количества ответов на данный вопрос в памяти располагаются: сам вопрос (то, на что надо ответить), и сами ответы (или варианты ответа). Строки отделяются друг от друга и от других данных нулевым символом. То есть если программа начинает считывать строковые данные, она считывает их в цикле, пока не встретит нулевой символ [7,67]. Это позволяет не хранить лишние данные в виде длины строки. Завершающими двумя байтами в вопросе являются номер правильного ответа.

Ограничения на файл теста:

- Всего можно хранить 65535 вопросов, потому что это наибольшее число, которое может поместиться в 2 байта, отведённые под количество вопросов.
- Всего можно хранить 16383 ответа на каждый вопрос, потому что под количество ответов отведено 14 бит (два бита из двухбайтного числа отведены под тип вопроса и регистрозависимость ответов).
- Длина строк не ограничена в размерах (условно, ограничения не накладываются со стороны приложения, но ограничения накладываются со стороны C++) и разделяются нулевым байтом.

Словесное описание алгоритма

На основании анализа примерного процесса тестирования, описанного в предыдущем разделе, можно сформировать алгоритм решения, на данном этапе – в словесной форме.

1. Считать файл с вопросами, записать вопросы в односвязный список.
2. Итерировать по списку, для каждого вопроса вызвать функцию, обрабатывающую вопрос.
3. Функция, обрабатывающая вопрос, в зависимости от типа вопроса – открытый/закрытый соответственно либо выводит вопрос и варианты ответа (для

закрытого), либо выводит только вопрос и приглашение для ввода ответа (для открытого) путём передачи объекта вопроса в специальную функцию по константной ссылке во избежание копирования объекта, что негативно скажется на производительности программы. Затем эта функция ожидает ввод ответа (для закрытого либо номер варианта ответа, либо сам ответ, а для открытого только ответ), и вызывает метод сравнения ответа, находящийся в объекте вопроса. Этот метод возвращает булево значение, является ли ответ на вопрос правильным.

4. В зависимости от исхода этого вопроса выводится ответ и подсчитывается статистика.

5. После того, как вопросы закончились, осуществляется вывод статистики.

### Предлагаемые методы алгоритмизации

Для реализации описанного алгоритма были задействован ряд алгоритмических методов, описанных в деталях ниже.

Приложение считывает тесты из файла с помощью класса `TestFileReader`. В этом классе есть следующие закрытые атрибуты:

- `questions` – односвязный список объектов класса `Question`, представляющих собой вопросы к тесту;

- `file` – входной поток файла с тестом. Из этого потока считывается информация о тесте и заносится в список `questions`.

Также класс `TestFileReader` предлагает следующие закрытые методы:

- `rev` – данный метод принимает на вход число и возвращает его с перевернутыми байтами. Например, при входе числа 1234 (0x04D2) данный метод вернёт число 53764 (0xD204). Этот метод перегружен для работы с двухбайтными беззнаковыми (`uint16_t`) и четырёхбайтными беззнаковыми (`uint32_t`) числами;

- `readByte` – данный метод не принимает параметров, считывает из входного потока `file` один байт информации и возвращает её в виде однобайтного беззнакового (`uint8_t`) числа;

- `read2Bytes` – данный метод не принимает параметров, считывает из входного потока `file` два байта информации и возвращает их в виде двухбайтного беззнакового (`uint16_t`) числа;

- `read4Bytes` – данный метод не принимает параметров, считывает из входного потока `file` четыре байта информации и возвращает их в виде четырёхбайтного беззнакового (`uint32_t`) числа;

- `readNullterminatedString` – данный метод не принимает параметров и является обёрткой над `readByte` для циклического чтения строки до встречи нулевого символа, после чего данный метод прекращает чтение строки и возвращает её в виде объекта `std::string`.

Данный класс также имеет конструктор, принимающий имя файла, и открытый метод `readFile`, который ничего не принимает, считывает файл теста по специальному образцу и возвращает список вопросов.

Для записи своих тестов существует аналогичный по своей структуре класс `TestFileWriter`.

Также был создан класс `List`, представляющий собой реализацию односвязного списка для хранения вопросов теста. Этот класс шаблонный, параметр шаблона – `Telem` – существует для того, чтобы можно было создавать списки, содержащие разные типы данных.

Этот класс имеет:

- конструктор с `std::initializer_list<T>`, созданный для удобного создания списка с элементами в одну строку;

- закрытое целочисленное поле `_length`, хранящее размер списка;

- открытую структуру `ListNode`, которая содержит один элемент списка с указателем на следующий (далее объекты этой структуры будут называться Узлом списка).

Определены следующие открытые атрибуты:

- указатель на тип `ListNode` под названием `first`, который содержит адрес первого Узла;

- метод `add`, который принимает константную ссылку на элемент списка и добавляет этот элемент в Узел, который помещается в конец списка;

- метод `elementAt`, принимающий индекс элемента и возвращающий ссылку на элемент под этим индексом. В случае пустого списка (если `_length == 0`), неправильного индекса (если индекс `<0` или `= _length`), то бросается исключение `std::invalid_argument` с соответствующим сообщением;

- метод `length`, ничего не принимающий и возвращающий длину списка;

- метод `iterate`, который принимает функцию от элемента списка и индекса и вызывает эту функцию для каждого элемента списка и соответствующего ему индекса;

- метод `filter`, который принимает булеву функцию (предикат) от элемента списка и индекса и возвращает новый список, который состоит из элементов, которые удовлетворяют предикату;

- метод `map`, который принимает функцию, возвращающую `Telem`, от элемента списка и индекса (трансформер) и возвращает новый список, состоящий из элементов-результатов вызова трансформера над каждым элементом списка;

- метод `toArray`, который принимает целочисленную ссылку на количество элементов в результирующем динамическом массиве, копирует все элементы списка в этот массив и возвращает указатель на первый элемент этого массива [3,51].

Интерфейс программы. Процедура создания новой базы тестирования

Интерфейс приложения ориентирован на работу с консолью. Как только пользователь запускает приложение, он видит предложение создать новый тест или открыть уже существующий и задать количество вопросов для теста (рис. 2).

```
Выберите действие:
1 - создать - создать новый тест
2 - открыть - открыть уже существующий тест
Введите или число (1 или 2), или название действия (создать или открыть).
> 1

Создать тест

Введите имя файла: quiz.txt
Введите количество вопросов: █
```

Рис.2. начало работы с программой.

Для создания новой базы тестирования (база вопросов и ответов к ним) необходимо выбрать первый пункт меню в главном окне программы (либо ввести цифру «1», либо слово «создать»).

При создании нового теста программа запрашивает имя файла нового теста. Расширение может быть любым, рекомендуется использовать «.txt» или «.quiz».

После ввода количества вопросов запрашиваются сами вопросы.

Для каждого вопроса запрашивается его текстовое содержание, то есть сам вопрос, тип вопроса (открытый или закрытый, пояснение по различию между данными типами дано в разделе «Контрольный пример»), количество ответов (или вариантов ответа для закрытого вопроса), и после этого запрашиваются сами ответы (или варианты ответа) на заданный вопрос.

Пример создания нового теста, состоящего из трёх вопросов представлен на рис. 3:

```

Выберите действие:
1 - создать - создать новый тест
2 - открыть - открыть уже существующий тест
Введите или число (1 или 2), или название действия (создать или открыть).
> 1

Создать тест

Введите имя файла: quiz.txt
Введите количество вопросов: 3

Введите вопрос: Что такое компилятор?
Введите тип вопроса (0 = закрытый, 1 = открытый): 0
Введите количество ответов: 4
Введите ответ №1: Компилятор - программа, которая выполняет построчный анализ, обработку и выполнение исходного кода программы или запроса
Введите ответ №2: Компилятор - программа, которая переводит текст программы, написанный на языке программирования, в набор машинных кодов
Введите ответ №3: Компилятор - программа, которая производит компоновку: принимает на вход один или несколько объектных модулей и собирает по ним исполнимый модуль
Введите ответ №4: Компилятор - спецификация языка C++
Введите номер правильного ответа (начиная с единицы): 2

Введите вопрос: Кто является создателем языка C++?
Введите тип вопроса (0 = закрытый, 1 = открытый): 0
Введите количество ответов: 4
Введите ответ №1: Чарльз Бэббидж
Введите ответ №2: Деннис Ритчи
Введите ответ №3: Брайан Керниган
Введите ответ №4: Бьярне Страуструп
Введите номер правильного ответа (начиная с единицы): 4

Введите вопрос: Назовите любую среду разработки на C++
Введите тип вопроса (0 = закрытый, 1 = открытый): 1
Введите количество ответов: 4
Введите ответ №1: Visual Studio
Введите ответ №2: CLion
Введите ответ №3: Dev-C++
Введите ответ №4: Code::Blocks

Press any key to continue . . .

```

Рис.3. Процедура формирования базы вопросов.

### Реализация алгоритма с помощью разработанных модульных структур

Помимо классов, описанных в разделе «Предлагаемые методы алгоритмизации», были разработаны следующие функции:

– **void createDatabaseRoutine();**

Данная функция позволяет вызвать процесс разработки новой базы тестирования. Эта функция вызывается, когда пользователь выбирает первый пункт в главном меню.

– **void readDatabaseRoutine();**

Данная функция позволяет вызвать процесс использования имеющейся базы тестирования. Эта функция вызывается, когда пользователь выбирает второй пункт в главном меню.

– **void processQuestion(const Question& q);**

Данная функция принимает в себя вопрос q и в зависимости от его типа вызывает либо processOpenQuestion, либо processCloseQuestion и передаёт в вызываемую функцию вопрос q.

– **void processOpenQuestion(const Question& q);**

Данная функция принимает в себя открытый вопрос q, выводит его и ожидает ввод ответа, сравнивает ответ пользователя с имеющимся ответом в базе, выводит соответствующий результат и увеличивает соответственно счётчик правильных или неправильных ответов.

– **void processCloseQuestion(const Question& q);**

Данная функция принимает в себя закрытый вопрос q, выводит его, выводит варианты ответа на данный вопрос и ожидает ввод ответа либо номера ответа, сравнивает ответ пользователя с имеющимся ответом в базе, выводит соответствующий результат и увеличивает соответственно счётчик правильных или неправильных ответов.

– **void** clearInp();

Данная функция очищает буфер потока ввода, что необходимо для вызова функции `getline` для чтения всей вводимой строки без ограничения по пробелам.

– **bool** isNumber(string s);

Данная функция возвращает булево значение, отвечающее на вопрос, состоит ли переданная строка `s` полностью из чисел.

### Тестирование программы

Для проверки правильности разработанной программы, а также полноты и массовости алгоритма, программу необходимо протестировать.

Для этого был произведён цикл операций по созданию новой базы тестирования и ее запуск. В ходе этой процедуры был проверен и приведённый контрольный пример контрольный пример на правильность выполнения.

На основании созданной базы вопросов и ответов студента производится автоматическая оценка ответа. Так, для полноты возможных исходов программы были введены два неправильных ответа (рис. 4). Скриншот процедуры тестирования показывает, что все этапы моделируемого процесса выполняются и программа реализует корректную оценку результатов.

### Заключение

Подводя итоги, необходимо отметить, что разработка этого проекта была достаточно интересной. Разработанная программа имеет большой потенциал и, надеемся, будет востребована в университетской среде как преподавателями, так и студентами для самопроверки.

В ходе разработки базы тестирования были включены вопросы из широкого спектра тем по программированию на языке C++, начиная от указателей, ссылок, циклов, условных операторов и заканчивая такими сложными вещами, как макросы, односвязный список, объектно-ориентированное программирование и работа с двоичными файлами и шаблонами.

Весь программный код, включая классы `Question`, `List`, `TestFileReader` и `TestFileWriter` был написан на основах знаний, полученных из курсов «Алгоритмизация и программирование», «Информационные технологии», а также с использованием дополнительных Интернет-источников [3,53].



```
Открыть тест
Введите имя файла: Kurs1.txt
Вопрос №1:
Что такое Visual Studio?
Варианты ответа:
1. Среда разработки
2. Компилятор
3. Разработчик C++
4. Операционная система
Ответ: 1
Ответ правильный!

Вопрос №2:
Назовите один из компиляторов для C++.
Ответ: clang
Ответ правильный!

Вопрос №3:
Назовите три области видимости, существующих в классах и структурах C++.
Варианты ответа:
1. open, closed, internal
2. listed, unlisted, static
3. published, privated, static
4. public, private, protected
5. visible, hidden, deleted
Ответ: 3
Ответ неправильный!

Вопрос №4:
Выберите правильный синтаксис открытого наследования класса в C++.
Варианты ответа:
1. class Derived extends public Base {};
2. class Derived : public Base {};
3. class Derived public extends Base {};
4. class Derived(public Base) {};
Ответ: 1
Ответ неправильный!

Вопрос №5:
Что такое указатель на void (void*)?
Варианты ответа:
1. Указатель на void способен представлять значения любых указателей на объектные и неполные типы, то есть адреса любых объектов.
2. Указатель на void является указателем на функцию, возвращающую void
3. Указатель на void является указателем на несуществующую или невыделенную область памяти.
4. Такого не существует, это синтаксическая ошибка.
Ответ: 1
Ответ правильный!

Тест закончен!
Всего вопросов: 5
Правильно отвеченных вопросов: 3 (60%)
Неправильно отвеченных вопросов: 2 (40%)
Press any key to continue . . .
```

Рис.4. Процедура тестирования и оценка полученных ответов.

## Литература

1. Гизатуллин Р.М., Бикеева Н.Г., Петрова Н.К.. Значение официального сайта университета для эффективной организации учебного процесса. Modern Science. 2021. (в печати).
2. Петрова Н.К., Мухачёв А.П., Загидуллин А.А. Разработка мобильного приложения по обучению основам алгоритмизации и программирования на базе языка Python. Математические методы в технике и технологиях - ММТТ. 2020. Т. 3. С. 131-134.
3. Курсовая работа по дисциплине «Алгоритмизация и программирование»: методические указания // сост. Петрова Н.К., С.М. Куценко. – Казань: Казан. гос. энерг. ун-т, 2019. – 51 с.
4. Ишмуратов Р.А., Ситников С.Ю. Применение визуальных сред разработки приложений для создания обучающих программ // Ученые записки ИСГЗ, Казань. - 2018. - Т.16. - №2, С.111-117.
5. Щелканов А.В., Чередов А.И. Пакет программ для тестирования студентов // Навигатор в мире науки и образования – 2012. – №4-7 – С.57.
6. Гладких И. Ю., Якушин А.В. Системы автоматизированного тестирования по программированию в образовательном пространстве // Современные проблемы науки и образования – 2016. – №3 – С.326.

7. Петрова Н.К., Бикеева Н.Г., Каюров Г.Д., Сапаров Ы.Д., Валиуллин К.И. Разработка программы по реализации тестирования студентов с вопросами открытого и закрытого типа // Образование и наука в России и за рубежом – 2019. – №10. – С.64-68.