

Министерство науки и высшего образования Российской Федерации
Санкт-Петербургский политехнический университет Петра Великого

Название института

Работа допущена к защите
Должность руководителя М
_____ И.О. Фамилия
«_____» _____ 202X г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

РАБОТА БАКАЛАВРА

ТЕМА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

по направлению подготовки XX.XX.XX Наименование направления подготовки
Направленность (профиль) XX.XX.XX_YY Наименование направленности (про-
филя) образовательной программы

Выполнил

студент гр. N

И.О. Фамилия

Руководитель

должность,

степень, звание¹

И.О. Фамилия

Консультант²

должность, степень

И.О. Фамилия

Консультант

по нормоконтролю³

И.О. Фамилия

Санкт-Петербург

202X

¹ Должность указывают сокращенно, учёную степень и звание — при наличии, а подразделения — аббревиатурами. «СПбПУ» и аббревиатуры институтов не добавляют.

² Оформляется по решению руководителя ОП или подразделения. Только 1 категория: «Консультант». В исключительных случаях можно указать «Научный консультант» (должен иметь степень). Без печати и заверения подписи.

³ Обязателен, из числа ППС по решению руководителя ОП или подразделения. Должность и степень не указываются. Сведения помещаются в последнюю строчку по порядку. Рецензенты не указываются.

**САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ПЕТРА ВЕЛИКОГО**
Название института

УТВЕРЖДАЮ

Должность руководителя М

_____ И.О. Фамилия

« _____ » _____ 202Хг.

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

студенту Фамилия Имя Отчество гр. N

1. Тема работы: Тема выпускной квалификационной работы.
2. Срок сдачи студентом законченной работы⁴: дд.мм.202Х.
3. Исходные данные по работе⁵: статистические данные с сайта [3.1], а также из репозитория [3.3]; основным источником литературы является монография [3.2] и статья [Article].
 - 3.1. Сайт Федеральной службы государственной статистики. — URL: <http://www.gks.ru/> (дата обращения: 06.03.2019).
 - 3.2. *Babington P.* The title of the work. Vol. 4. — 3rd ed. — The address: The name of the publisher, 1993. — 255 p. — (Ser.: 10).
 - 3.3. The UC Irvine Machine Learning Repository. — URL: <http://archive.ics.uci.edu/ml> (visited on 06.03.2019).
4. Содержание работы (перечень подлежащих разработке вопросов):
 - 4.1. Обзор литературы по теме ВКР.
 - 4.2. Исследование программных продуктов.
 - 4.3. Разработка метода/алгоритма/программы.
 - 4.4. Апробация разработанного метода/алгоритма/программы.

⁴Определяется руководителем ОП, но не позднее последнего числа преддипломной практики и/или не позднее, чем за 20 дней до защиты в силу п. 6.1. «Порядка обеспечения самостоятельности выполнения письменных работ и проверки письменных работ на объем заимствований».

⁵Текст, который подчеркнут и/или выделен в отдельные элементы нумерационного списка, приведён в качестве примера.

5. Перечень графического материала (с указанием обязательных чертежей):
- 5.1. Схема работы метода/алгоритма.
 - 5.2. Архитектура разработанной программы/библиотеки.
6. Консультанты по работе⁶:
- 6.1. Должность, степень, И.О. Фамилия.
 - 6.2. Должность, степень, И.О. Фамилия (нормоконтроль).
7. Дата выдачи задания⁷: дд.мм.202Х.

Руководитель ВКР _____ И.О. Фамилия

Консультант⁸ _____ И.О. Фамилия

Задание принял к исполнению дд.мм.202Х

Студент _____ И.О. Фамилия

⁶Подпись консультанта по нормоконтролю пока не требуется. Назначается всем по умолчанию.

⁷Не позднее 3 месяцев до защиты (утверждение тем ВКР по университету) или первого числа преддипломной практики или по решению руководителя ОП или подразделения (открытый вопрос).

⁸В случае, если есть консультант, отличный от консультанта по нормоконтролю.

РЕФЕРАТ

На 33 с., 6 рисунков, 6 таблиц, 2 приложения.

КЛЮЧЕВЫЕ СЛОВА: СТИЛЕВОЕ ОФОРМЛЕНИЕ САЙТА, УПРАВЛЕНИЕ КОНТЕНТОМ, PHP, MYSQL, АРХИТЕКТУРА СИСТЕМЫ.⁹

Тема выпускной квалификационной работы: «Тема выпускной квалификационной работы»¹⁰.

В данной работе изложена сущность подхода к созданию динамического информационного портала на основе использования открытых технологий Apache, MySQL и PHP. Даны общие понятия и классификация IT-систем такого класса. Проведен анализ систем-прототипов. Изучена технология создания указанного класса информационных систем. Разработана конкретная программная реализация динамического информационного портала на примере портала выбранной тематики...¹¹

В данной работе изложена сущность подхода к созданию динамического информационного портала на основе использования открытых технологий Apache, MySQL и PHP. Даны общие понятия и классификация IT-систем такого класса. Проведен анализ систем-прототипов. Изучена технология создания указанного класса информационных систем. Разработана конкретная программная реализация динамического информационного портала на примере портала выбранной тематики...

ABSTRACT

33 p., 6 figures, 6 tables, 2 appendices.

KEYWORDS: STYLE REGISTRATION, CONTENT MANAGEMENT, PHP, MYSQL, SYSTEM ARCHITECTURE.

⁹Всего **слов**: от 3 до 15. Всего **слов и словосочетаний**: от 3 до 5. Оформляются в именительном падеже множественного числа (или в единственном числе, если нет другой формы), оформленных по правилам русского языка. *Внимание! Размещение сноски после точки является примером как запрещено оформлять сноски.*

¹⁰Реферат **должен содержать**: предмет, тему, цель ВКР; метод или методологию проведения ВКР; результаты ВКР: область применения результатов ВКР; выводы.

¹¹ОТ 1000 ДО 1500 печатных знаков (ГОСТ Р 7.0.99-2018 СИБИД) на русский или английский текст. Текст реферата повторён дважды на русском и английском языке для демонстрации подхода к нумерации страниц.

The subject of the graduate qualification work is «Title of the thesis».

In the given work the essence of the approach to creation of a dynamic information portal on the basis of use of open technologies Apache, MySQL and PHP is stated. The general concepts and classification of IT-systems of such class are given. The analysis of systems-prototypes is lead. The technology of creation of the specified class of information systems is investigated. Concrete program realization of a dynamic information portal on an example of a portal of the chosen subjects is developed...

In the given work the essence of the approach to creation of a dynamic information portal on the basis of use of open technologies Apache, MySQL and PHP is stated. The general concepts and classification of IT-systems of such class are given. The analysis of systems-prototypes is lead. The technology of creation of the specified class of information systems is investigated. Concrete program realization of a dynamic information portal on an example of a portal of the chosen subjects is developed...

СОДЕРЖАНИЕ

Введение	6
Глава 1. Обзор теоретической литературы по криптографии и интернету вещей.....	9
1.1. Криптографические алгоритмы: основные понятия.....	9
1.1.1. Информация и криптография.....	9
1.1.2. Шифры: основные понятия и принципы	10
1.1.3. Симметричная криптография.....	12
1.1.4. Асимметричная криптография.....	12
1.1.5. Криптографические хэш-функции	13
1.2. Название параграфа.....	15
1.3. Выводы	16
Глава 2. Название второй главы: разработка метода, алгоритма, модели исследования.....	16
2.1. Название параграфа.....	16
2.2. Название параграфа.....	17
2.2.1. Название подпараграфа	17
2.3. Название параграфа.....	18
2.4. Выводы	25
Глава 3. Название третьей главы: разработка программного обеспечения..	26
3.1. Название параграфа.....	26
3.2. Название параграфа.....	26
3.3. Выводы	26
Глава 4. Название четвёртой главы. Апробация результатов исследования, а именно: метода, алгоритма, модели исследования	26
4.1. Название параграфа.....	26
4.2. Название параграфа.....	26
4.3. Выводы	26
Заключение	27
Список сокращений и условных обозначений.....	28
Словарь терминов.....	29
Список использованных источников.....	30
Приложение 1. Краткие инструкции по настройке издательской системы L ^A T _E X	34
Приложение 2. Некоторые дополнительные примеры	38

ВВЕДЕНИЕ

В последние десятилетия сетевые технологии прочно вошли в жизнь человека. Изначально они были представлены локальными сетями, затем была создана сеть Интернет. В настоящее время одним из бурно развивающихся направлений сетевых технологий является интернет вещей (Internet of Things, IoT). Так, за период 2015-2018 гг. доля IoT-устройств среди всех устройств увеличилась с 27% до 39%, и, согласно прогнозу, достигнет 63% к 2025 году [42].

Интернет вещей – это вычислительная сеть физических предметов (устройств, «вещей»), оснащенных встроенной технологией для взаимодействия друг с другом или с внешней средой [30].

Актуальность исследования. Одной из важных задач при проектировании IoT является обеспечение должного уровня безопасности передаваемых данных. Особенно это важно для медицинских устройств[2]. Уже сейчас правительства развитых стран начинают принимать законы, регламентирующие защиту IoT-устройств[27][7][6]. В то же время фактическая безопасность устройств интернета вещей оставляет желать лучшего. Так, согласно исследованию корпорации HP 2014-го года, 70% устройств IoT передавали данные, в том числе конфиденциального характера, вообще без шифрования[29]! По этой причине изучение и развитие средств защиты IoT-устройств является актуальной задачей.

Безопасность устройств с ограниченными энергетическими ресурсами (а именно такими являются устройства интернета вещей) изучает раздел криптографии, называемый легковесной криптографией (lightweight cryptography, LWC). Также возможны термины «облегченная криптография», «малоресурсная криптография», «низкоэнергетическая криптография». Она рассматривает криптографические алгоритмы в контексте их требовательности к ресурсам устройства и количеству логических элементов, требуемых для реализации алгоритмов. Рассматриваемые ею алгоритмы называются алгоритмами легковесной криптографии (легковесными алгоритмами, LW-алгоритмами). LW-алгоритмы могут иметь программную или аппаратную реализацию.

Объектом исследования являются алгоритмы легковесной криптографии, ориентированные на использование в устройствах интернета вещей.

Предметом исследования являются следующие характеристики легковесных алгоритмов: тип алгоритма, требования к устройству, производительность. Алгоритмы рассматриваются с точки зрения программной реализации.

Целью исследования является сравнение легковесных алгоритмов и определение степени их пригодности к использованию в устройствах IoT. Для достижения выбранной цели поставлены следующие **задачи**:

- А. Обзор теоретических источников по темам легковесной криптографии и интернета вещей.
- В. Обзор существующих стандартов и технических решений.
- С. Анализ различных видов алгоритмов.
- Д. Реализация избранных алгоритмов.
- Е. Ранжирование видов алгоритмов по степени пригодности к использованию в устройствах IoT. Формулировка рекомендаций по их использованию.

Гипотеза исследования. Предполагается, что значительное число классических шифров с определенным ослаблением могут использоваться в качестве легковесных алгоритмов. С другой стороны, многие алгоритмы и классы алгоритмов по тем или иным соображениям однозначно не могут быть использованы в качестве LW-алгоритмов. Ряд алгоритмов может быть использован наилучшим образом при определенных условиях. Может быть создана методика, позволяющая приблизительно оценить производительность реализации легковесного алгоритма с использованием только персонального компьютера, без необходимости взаимодействия с низкоресурсными устройствами.

В данной работе используются такие **методы исследования**, как:

- анализ технической литературы;
- изучение существующих спецификаций алгоритмов, стандартов, технологических систем;
- декомпозиция алгоритмов;
- сравнение алгоритмов, подходов к шифрованию, структурных частей алгоритмов;
- реализация алгоритмов на языке C;
- определение методологии их тестирования;
- реализация конкретной методики тестирования;
- анализ результатов тестирования;
- синтез выводов по результатам тестирования;
- обобщение результатов работы.

Данное исследование имеет высокую **практическую значимость**. Результаты исследования могут быть использованы при выборе алгоритма шифрования и

режима его работы при проектировании системы защиты вычислительной сети IoT. Кроме того, работа в данном направлении может быть продолжена: следующим шагом возможна оптимизация существующих легковесных алгоритмов или создание новых LW-алгоритмов. Предложенная методология тестирования времени работы алгоритмов и их энергопотребления может быть использована при тестировании других алгоритмов. Присутствует и **научная значимость**. Можно исследовать «побочные» вопросы, возникшие в процессе выполнения работы.

Введение раскрывает актуальность, определяет степень научной разработки темы, объект, предмет, цель, задачи и методы исследования, раскрывает теоретическую и практическую значимость работы.

В первой главе содержатся краткое теоретическое введение в тему криптографии и интернета вещей, описываются основные угрозы безопасности систем IoT.

Во второй главе вводится понятие легковесных криптографических алгоритмов. Описываются требования к ним и их применение для нейтрализации угроз безопасности системам IoT, оно сравнивается с нейтрализацией угроз в системах общего назначения). Проводится анализ различных видов алгоритмов на пригодность их для такого применения, рассматриваются наиболее популярные их представители.

В третьей главе описывается методология тестирования производительности и энергопотребления легковесных криптоалгоритмов на персональном компьютере.

В четвертой главе содержится информация о реализации алгоритмов в программном коде, способе и порядке тестирования производительности алгоритмов, а также дальнейшей обработки результатов тестирования. Проводится анализ полученных результатов.

В заключении подводятся итоги работы, приводятся краткие перспективы использования результатов работы.

ГЛАВА 1. ОБЗОР ТЕОРЕТИЧЕСКОЙ ЛИТЕРАТУРЫ ПО КРИПТОГРАФИИ И ИНТЕРНЕТУ ВЕЩЕЙ

1.1. Криптографические алгоритмы: основные понятия

1.1.1. Информация и криптография

Человеческая цивилизация в течение всего времени существования работает с различной информацией. Информация может представлять собой как исторические сведения или описания технологических процессов, так и частные данные человека или группы лиц. Сведения, составляющие информацию и представляющие определенную ценность, должны быть защищены от лиц, не имеющих соответствующих прав доступа. Поэтому задача защиты информации всегда была актуальной.

Защитить информацию можно тремя способами [12].

- А. Создание абсолютно надежного и изолированного от доступа извне хранилища информации и инфраструктуры для него. Это является крайне трудоемкой и дорогостоящей задачей.
- В. Скрытие факта существования или передачи информации. Средства и методы такого скрытия изучает стеганография.
- С. Хранение и передача информации, преобразованной таким образом, чтобы обратное преобразование могли совершить только определенные лица. Средства и методы такого преобразования информации изучает криптография.

Формализуем вышесказанное.

Информация – сведения, передаваемые одними людьми другим людям устным, письменным или каким-либо другим способом, а также сам процесс передачи или получения этих сведений [8].

Защита информации – деятельность, направленная на предотвращение утечки защищаемой информации, несанкционированных и непреднамеренных воздействий на защищаемую информацию [3]. Утечка – неконтролируемое распространение защищаемой информации в результате ее разглашения, несанкционированного доступа к информации и получения защищаемой информации иностранными разведками [4].

Криптография — инженерно-техническая дисциплина, изучающая математические методы защиты информации (шифры). Криптография включает в себя криптосинтез и криптоанализ. Криптосинтез изучает подходы к разработке шифров. Криптоанализ изучает подходы к вскрытию шифров [12].

1.1.2. Шифры: основные понятия и принципы

Итак, шифр (криптографический алгоритм, шифрующая функция) – преобразование данных, обеспечивающее их защищённость (зашифрованность). Это преобразование вида $f: x \rightarrow y$, где x – исходные, незашифрованные данные (представленные в виде целого числа), y – полученный зашифрованный текст (также целое число), f – шифрующая функция.

Дешифрующая функция – функция, обратная к шифрующей: $f^{-1}: y \rightarrow x$. Результатом применения дешифрующей функции к зашифрованным данным являются исходные данные.

Для того, чтобы шифрование было эффективным, необходимо, чтобы шифрующая функция f была легко вычислимой, для затруднения дешифровки дешифрующая функция f^{-1} должна быть достаточно сложна для вычисления [10]. Функции такого вида называются односторонними.

Функция f называется *односторонней*, если выполняются условия [10]:

- А. для любого x из некоторого множества существует эффективный алгоритм вычисления $y=f(x)$;
- В. не существует эффективного алгоритма обращения функции f .

Однако шифр должен быть не только недоступным для взлома злоумышленником. Законный получатель должен иметь возможность эффективно получить исходные данные. Это требование формализуется как «функция с секретом» или «односторонняя функция с секретом».

Функция f называется *односторонней функцией с секретом*, если выполняются условия [10]:

- А. для любого x из некоторого множества существует эффективный алгоритм вычисления $y=f(x)$;
- В. функция f обладает «секретным свойством» k , таким что:.

1. при использовании свойства k можно построить эффективный алгоритм построения обратной функции f^{-1} ;

2. если свойство k неизвестно, то не существует эффективного алгоритма обращения функции f .

Итак, при использовании шифра на основе функции с секретом, шифровать (вычислять функцию f) могут все, а эффективно расшифровывать – только лица, которым известно секретное свойство k .

Секретное свойство также называют *ключом шифра*. Он представляет собой число или набор чисел (параметров алгоритма).

Главная часть шифра, его сердце – некоторая *трудная* задана, которая используется для гарантии того, что узнать секретное свойство (т. е. обратить шифрующую функцию и получить доступ к зашифрованным данным) очень трудно. Пара примеров трудных задач:

- Задача факторизации – разложение большого числа на простые множители (используется, например, в алгоритме RSA). В этом случае секретное свойство – факторизуемое число.
- Задача дискретного логарифмирования – обращение функции вида $f(x)=a^x \bmod p$, где p – большое простое число, a – параметр, подобранный для конкретного p (используется, например, в схеме Эль-Гамала).

Однако если будет представлен алгоритм, эффективно решающий трудную задачу (т. е. задача перестанет быть трудной), шифры на ее основе мгновенно устареют. Поэтому поиск трудных задач (и, наоборот, поиски быстро решающих их алгоритмов) является одним из важных направлений криптографии.

С другой стороны, в секрете хранится только ключ алгоритма, сам алгоритм должен быть открытым. То есть вся криптографическая стойкость алгоритма должна содержаться в ключе, а злоумышленник может знать о криптографической схеме все, кроме ключа. Это правило называется *принципом Керкгоффса*. Его выполнение обеспечивает надежность системы: даже в случае подбора ключа (методом перебора или, например, шантажа) злоумышленником достаточно его сменить, и начинать взлом нужно с самого начала. Похожий принцип «враг знает систему» был (вероятно, независимо) сформулирован Шенноном и называется *максимой Шеннона* [16].

Это, однако, является только рекомендацией. В правительственных, военных и ряде других областей применяется противоположный принцип: безопасность через неясность. Его суть состоит в скрывании внутренней структуры системы безопасности. Для шифров общего назначения он не рекомендуется, их, напротив,

публикуют и обсуждают, чтобы коллективно найти уязвимости и возможности улучшения [40].

1.1.3. Симметричная криптография

Изначально существовали только симметричные схемы шифрования. Они предполагают наличие одного ключа, используемого как для шифрования, так и для расшифровки сообщений. Он хранится в тайне, поэтому такие схемы также называются схемами с закрытым ключом. Классический пример – алгоритм DES.

Симметричные шифры в основном состоят из двух главных компонентов: шифры подстановки (перестановка символов/битов сообщения) и шифры замены (замена символов/битов по отдельности на другие по некоторому правилу) [12].

Симметричные шифры делятся на две категории: блочные и потоковые. Потоковые криптоалгоритмы обрабатывают данные побитово или побайтово. Блочные алгоритмы обрабатывают данные целыми группами битов (блоками), обычно размер блока кратен 64 и составляет от 64 до 256 бит.

Симметричные шифры обладают рядом недостатков:

- Ключ должен храниться в тайне обеими сторонами и передаваться только по *защищенному* каналу (либо в зашифрованном виде). Это требует дополнительных ресурсов.
- Шифровать могут только те, кто знает ключ. Это означает либо жесткие ограничения на количество шифрующих лиц, либо значительный риск компрометации ключа в случае ослабления этих ограничений (компрометация ключа – его раскрытие не криптографическим способом [39]).

1.1.4. Асимметричная криптография

После появления понятия односторонней функции появился другой вид алгоритмов – асимметричные. Такой способ шифрования предполагает наличие известного всем участникам схемы шифрования *открытого* ключа и известного только законным участникам *закрытого* ключа. Открытый ключ используется для шифрования данных, а закрытый – для расшифровки. Таким образом, шифровать сообщения может кто угодно, а вот расшифровывать – только законные пользователи.

Такой способ работы снимает основные проблемы симметричного шифрования: хранить в секрете шифрующий ключ нет необходимости, также он может

передаваться по открытым каналам в незашифрованном виде и транслироваться на любую аудиторию.

По причине наличия открытого ключа такие алгоритмы также называются алгоритмами с открытым ключом. Классический пример – алгоритм RSA.

В некоторых случаях, например при использовании цифровой подписи, сообщения шифруются закрытым ключом, а расшифровываются открытым.

1.1.5. Криптографические хэш-функции

Другим направлением криптографии, основанным на понятии односторонней функции, являются криптографические (односторонние, однонаправленные) хэш-функции.

Хэш-функция – функция, принимающая строку произвольной (или почти произвольной) длины, и преобразующую ее в строку фиксированной, обычно меньшей, длины. Полученная строка называется отпечатком (дайджестом) входной строки или ее хэш-кодом.

Однонаправленная хэш-функция вычисляется только в одном направлении: легко вычислить значение дайджеста по входной строке, но крайне трудно создать прообраз, дайджест которого соответствует заданной строке [39].

Криптографическая хэш-функция является открытой. Безопасность обеспечивается именно однонаправленностью функции. Одним из необходимых условий является изменение значения половины битов дайджеста при изменении в одном бите входа: невозможно путем сравнения близких входов обратить хэширующую функцию.

Хэш-функции применяются при вычислении контрольных сумм для проверки подлинности файлов и транзакций: почти невозможно подобрать поддельный файл, отличный от настоящего, но с таким же дайджестом. При таком применении от функции требуется высокое быстродействие, т. к. файлы могут иметь значительный размер, а транзакции часто должны обрабатываться в реальном времени.

Еще одно использование контрольных сумм – при использовании цифровой подписи можно подписывать не весь документ целиком, а только его контрольную сумму. Это значительно ускоряет работу с подписью и уменьшает потребление памяти при хранении подписей [39].

Также криптографические хэш-функции используются для хранения паролей. Такие хэш-функции называются Key Derivation Functions, KDF. В этом случае

на носителе пароль не сохраняется, а хранится только его хэш, и каждый раз при вводе строки вычисляется ее хэш-код и сравнивается с хэшем пароля. Если произошло совпадение, значит, считаем, пароль введен верно. Для минимизации вероятности коллизии можно хранить два дайджеста от разных хэш-функций.

Такой способ хранения паролей является в настоящее время наиболее распространенным, так как сам по себе обеспечивает дополнительный уровень защищенности: даже если злоумышленник получил доступ к хэшам, для восстановления по ним паролей ему потребуется много времени (или большие вычислительные мощности). При этом от хэширующей пароли функции требуется, чтобы она вычислялась долго: время вычисления порядка 100 мс незаметно при авторизации, однако значительно затрудняет подбор пароля методом грубой силы. Кроме того, по тем же соображениям желательно большое потребление оперативной памяти, это также усложняет перебор, особенно в многопоточном режиме. К их стойкости также предъявляются повышенные требования.

Для дополнительного повышения защищенности хэшированных паролей используется *соль*. Это случайная строка, добавляемая к шифруемым данным, она должна храниться вместе с хэшем. Она не позволит понять, что захэшированы одинаковые строки, так как они будут иметь разную соль. Благодаря этому, брутфорс одного пароля из базы не позволяет найти другие такие же пароли в базе [39].

Итак, есть два вида криптографических хэш-функций:

- Быстрые, используются для вычисления контрольных сумм и должны потреблять как можно меньше ресурсов. Примеры: MD5, семейство SHA2, из новых – SHA3 и BLAKE2.
- Медленные, используются для хэширования паролей. Должны потреблять много ресурсов и быть очень стойкими. Примеры: bcrypt и scrypt, новая – Argon2.

Одиночные формулы оформляют в окружении *equation*, например, как указано в следующей одиночной нумерованной формуле:

$$\pi \approx 3,141. \quad (1.1)$$

На рис.1.1 изображена гидробашня СПбПУ, а в табл.2.2 приведены данные, на примере которых коротко и наглядно будет изложена суть ВКР.



Рис.1.1. Вид на гидробашню СПбПУ [41]

1.2. Название параграфа

Формулы могут быть размещены в несколько строк. Чтобы выставить номер формулы напротив средней строки, используйте окружение `multlined` из пакета `mathtools` следующим образом [26]:

$$\begin{aligned}
 (A_1, B_1) &\leqslant (A_2, B_2) \Leftrightarrow \\
 &\Leftrightarrow A_1 \subseteq A_2 \Leftrightarrow \\
 &\Leftrightarrow B_2 \subseteq B_1.
 \end{aligned}
 \tag{1.2}$$

Используя команду `\labelcref` из пакета `cleveref`, допустимо следующим образом оформлять ссылку на несколько формул: (1.1 и 1.2). На рис.1.2 приведены три картинki под общим номером и названием, но с отдельной нумерацией подрисунков посредством пакета `subcaption`.

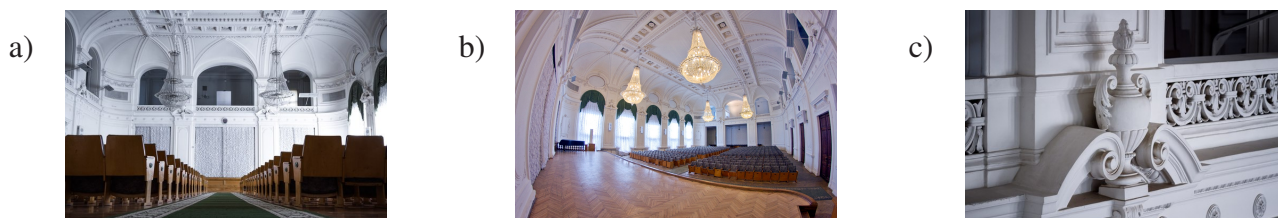


Рис.1.2. Фотографии Белого зала СПбПУ [41], в том числе: *a* — со стороны зрителей; *b* — со стороны сцены; *c* — барельеф

Далее можно ссылаться на три отдельных рисунка: рис.1.2а, рис.1.2b и рис.1.2с.

Пример ссылок [**Article**; 14; 15; 17; 20—22; 25; 28; 31; 32; 34; 35; 44], а также ссылок с указанием страниц, на котором отображены номера страниц [37, с. 96] или в виде мультицитаты на несколько источников [37, с. 96; 26, с. 46]. Часть библиографических записей носит иллюстративный характер и не имеет отношения к реальной литературе.

1.3. Выводы

Текст выводов по главе 1.

Кроме названия параграфа «выводы» можно использовать (единообразно по всем главам) следующие подходы к именованию последних разделов с результатами по главам:

- «выводы по главе N», где N — номер соответствующей главы;
- «резюме»;
- «резюме по главе N», где N — номер соответствующей главы.

Параграф с изложением выводов по главе *является обязательным*.

ГЛАВА 2. НАЗВАНИЕ ВТОРОЙ ГЛАВЫ: РАЗРАБОТКА МЕТОДА, АЛГОРИТМА, МОДЕЛИ ИССЛЕДОВАНИЯ

Глава посвящена более подробным примерам оформления текстово-графических объектов.

В параграфе 2.1 приведены примеры оформления многострочной формулы и одиночного рисунка. Параграф 2.2 раскрывает правила оформления перечислений и псевдокода. В параграфе 2.3 приведены примеры оформления сложносоставных рисунков, длинных таблиц, а также теоремоподобных окружений.

2.1. Название параграфа

Все формулы, размещенные в отдельных строках, подлежат нумерации, например, как формулы (2.1) и (2.2) из [26].

$$A^\uparrow = \{m \in M \mid gIm \ \forall g \in A\}; \quad (2.1)$$

$$B^\downarrow = \{g \in G \mid gIm \ \forall m \in B\}. \quad (2.2)$$

Обратим внимание, что формулы содержат знаки препинания и что они выровнены по левому краю (с помощью знака & окружения align).

На рис.2.1 приведёна фотография Нового научно-исследовательского корпуса СПбПУ.



Рис.2.1. Новый научно-исследовательский корпус СПбПУ [41]

2.2. Название параграфа

Название параграфа оформляется с помощью команды `\section{...}`, название главы — `\chapter{...}`.

2.2.1. Название подпараграфа

Название подпараграфа оформляется с помощью команды `\subsection{...}`.

Использование подподпараграфов в основной части крайне не рекомендуется. В случае использования, необходимо вынести данный номер в содержание. Название подпараграфа оформляется с помощью команды `\subsubsection{...}`.

Вместо подподпараграфов рекомендовано использовать перечисления.

Перечисления могут быть с нумерационной частью и без неё и использоваться с иерархией и без иерархии. Нумерационная часть при этом формируется следующим способом:

1. в перечислениях *без иерархии* оформляется арабскими цифрами с точкой (или длинным тире).
2. В перечислениях *с иерархией* — в последовательности сначала прописных латинских букв с точкой, затем арабских цифр с точкой и далее — строчных латинских букв со скобкой.

Далее приведён пример перечислений с иерархией.

А. Первый пункт.

В. Второй пункт.

С. Третий пункт.

Д. По ГОСТ 2.105–95 [5] первый уровень нумерации идёт буквами русского или латинского алфавитов (*для определенности выбираем английский алфавит*), а второй — цифрами.

1. В данном пункте лежит следующий нумерованный список:

- а) первый пункт;
- б) третий уровень нумерации не нормирован ГОСТ 2.105–95 (*для определенности выбираем английский алфавит*);
- с) обращаем внимание на строчность букв в этом нумерованном и следующем маркированном списке:
 - первый пункт маркированного списка.

Е. Пятый пункт верхнего уровня перечисления.

Маркированный список (без нумерационной части) используется, если нет необходимости ссылки на определенное положение в списке:

- первый пункт с *маленькой буквы* по правилам русского языка;
- второй пункт с *маленькой буквы* по правилам русского языка.

Оформление псевдокода необходимо осуществлять с помощью пакета `algorithm2e` в окружении `algorithm`. Данное окружение интерпретируется в шаблоне как рисунок. Пример оформления псевдокода алгоритма приведён на рис.2.2.

Обратим внимание, что можно сослаться на строчку 1 псевдокода из рис.2.2.

2.3. Название параграфа

Одиночные формулы также, как и отдельные формулы в составе группы, могут быть размещены в несколько строк. Чтобы выставить номер формулы напротив средней строки, используйте окружение `multlined` из пакета `mathtools` следующим образом [26]:

$$\begin{aligned}
 (A_1, B_1) &\leq (A_2, B_2) \Leftrightarrow \\
 &\Leftrightarrow A_1 \subseteq A_2 \Leftrightarrow \\
 &\Leftrightarrow B_2 \subseteq B_1.
 \end{aligned}
 \tag{2.3}$$

Algorithm

Input: the many-valued context $M \stackrel{\text{def}}{=} (G, M, W, J)$, the class membership $\varepsilon : G \rightarrow K$

Output: positive and negative binary contexts $\overline{K}_+ \stackrel{\text{def}}{=} (\overline{G}_+, M, I_+)$, $\overline{K}_- \stackrel{\text{def}}{=} (\overline{G}_-, M, I_-)$ such that i-tests found in \overline{K}_+ are diagnostic tests in M , and objects from \overline{K}_- are counter-examples

```

1. for  $\forall g_i, g_j \in G$  do
2.   if  $i < j$  then
3.      $\overline{G} \leftarrow (g_i, g_j)$ ;
4.   for  $\forall (g_i, g_j) \in \overline{G}$  do
5.     if  $m(g_i) = m(g_j)$  then
6.        $(g_i, g_j)Im$ ;
7.     if  $\varepsilon(g_i) = \varepsilon(g_j)$  then
8.        $\overline{G}_+ \leftarrow (g_i, g_j)$ ;
9.     else  $\overline{G}_- \leftarrow (g_i, g_j)$ ;
10.   $I_+ = I \cap (\overline{G}_+ \times M)$ ,  $I_- = I \cap (\overline{G}_- \times M)$ ;
11.  for  $\forall \overline{g}_+ \in \overline{G}_+$ ,  $\forall \overline{g}_- \in \overline{G}_-$  do
12.    if  $\overline{g}_+^\uparrow \subseteq \overline{g}_-^\uparrow$  then
13.       $\overline{G}_+ \leftarrow \overline{G}_+ \setminus \overline{g}_+$ ;

```

Рис.2.2. Псевдокод алгоритма DiagnosticTestsScalingAndInferring [37]

Используя команду `\labelcref{...}` из пакета `cleveref`, допустимо оформить ссылку на несколько формул, например, (2.1–2.3).

Пример оформления четырёх иллюстраций в одном текстово-графическом объекте приведён на рис.2.3. Это возможно благодаря использованию пакета `subcaption`.

Далее можно сослаться на составные части данного рисунка как на самостоятельные объекты: рис.2.3а, рис.2.3b, рис.2.3с, рис.2.3d или на три из четырёх изображений одновременно: рис.2.3а–2.3с.

Приведём пример табличного представления данных с записью продолжения на следующей странице на табл.2.1.

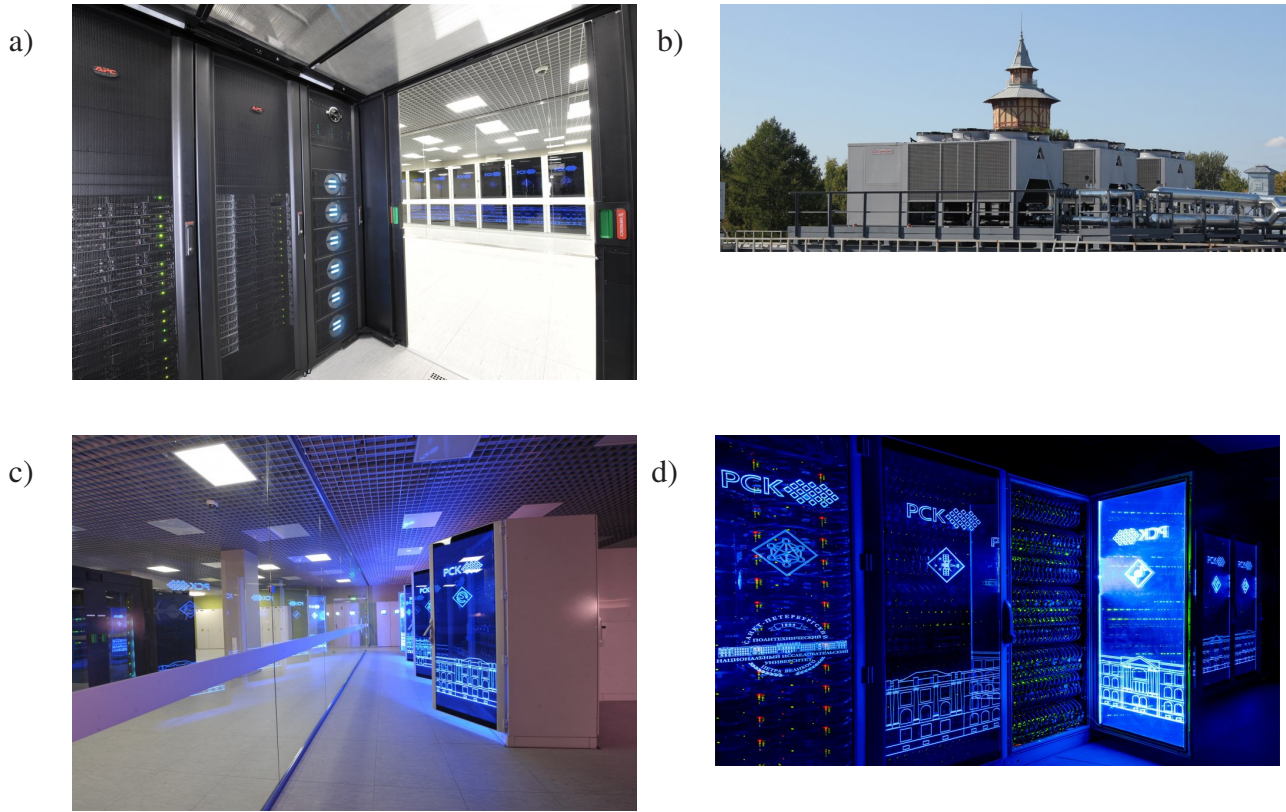


Рис.2.3. Фотографии суперкомпьютерного центра СПбПУ [41]: *a* — система хранения данных и узлы NUMA-вычислителя; *b* — холодильные машины на крыше научно-исследовательского корпуса; *c* — машинный зал; *d* — элементы вычислительных устройств

Таблица 2.1

Пример задания данных из [38] (с повтором для переноса таблицы на новую страницу)

G	m_1	m_2	m_3	m_4	K
1	2	3	4	5	6
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2

Продолжение табл. 2.1

1	2	3	4	5	6
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2

Таблица 2.2

Пример представления данных для сквозного примера по ВКР [38]

G	m_1	m_2	m_3	m_4	K
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2

Таблица 2.3

Пример задания данных в табличном виде из [38] (с помощью окружения `minipage`)

G	m_1	m_2	m_3	m_4	K
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2



Рис.2.4. Новый научно-исследовательский корпус СПбПУ [41] (с помощью окружения `minipage`)

Вопросы форматирования текстово-графических объектов (окружений) не регламентированы в известных нам ГОСТах, поэтому предлагаем придерживаться следующих правил:

- **полужирный текст** рекомендуем использовать только для названий стандартных окружений с нумерационной частью, например, для представления *впервые*: **определение 1.1**, **теорема 2.2**, **пример 2.3**, **лемма 4.5**;
- *курсив* рекомендуем использовать только для выделения переменных в формулах, служебной информации об авторах главы (статьи), важных терминов, представляемых по тексту, а также для всего тела окружений, связанных с получением *новых существенных результатов и их доказательством*: теорема, лемма, следствие, утверждение и другие.

По аналогии с нумерацией формул, рисунков и таблиц нумеруются и иные текстово-графические объекты, то есть включаем в нумерацию номер главы, например: теорема 3.1. для первой теоремы третьей главы монографии. Команды \LaTeX выставляют нумерацию и форматирование автоматически. Полный перечень команд для подготовки текстово-графических и иных объектов находится в подробных методических рекомендациях [13].

Для удобства авторов названия стандартных окружений, рекомендованных к использованию, приведены в табл.2.4, а в табл.2.5 перечислены имена специально разработанных окружений для шаблонов SPbPU.

Таблица 2.4

Стандартные окружения

Название окружения	Назначение
center	центрирование, аналог команды <code>\centering</code> , но с добавлением нежелательного пробела, поэтому лучше избегать применения <code>center</code>
itemize	перечисления, в которых нет необходимости нумеровать пункты (немаркированные списки)
enumerate	перечисления с нумерацией (немаркированные списки)
refsection	создание отдельных библиографических списков для глав
tabular	оформление таблиц
table	автоматическое перемещение по тексту таблиц, оформленных, например, с помощью <code>tabular</code> , для минимизации пустых пространств
longtable	оформление многостраничных таблиц
tikzpicture	создание иллюстраций с помощью пакета <code>tikz</code> [23]
figure	автоматическое перемещение по тексту рисунков, оформленных, например, с помощью <code>tikz</code> или подключенных с помощью команды <code>\includegraphics</code> , для минимизации пустых пространств
subfigure	оформление вложенных рисунков в составе <code>figure</code>
algorithm	оформление псевдокода на основе пакета <code>algorithm2e</code> [24]
minipage	оформление рисунков и таблиц без функций автоматического перемещения по тексту для минимизации пустых пространств
equation	оформление выключенных (не встроенных в текст с помощью <code>\$...\$</code>) одиночных формул на одной строке
multilined	оформление выключенных (не встроенных в текст с помощью <code>\$...\$</code>) одиночных формул в несколько строк
aligned	оформление нескольких формул с выравниванием по символу <code>&</code> .

На базе пакета `tikz` разработано большое количество расширений [23], например, `tikzcd`, которые мы рекомендуем использовать для оформления иллюстраций.

В случае, если авторам потребовалось новое окружение, то создать его можно в файле `my_folder/my_settings.tex` согласно правилам, приведённым ниже.

1. Для перехода в режим создания окружений следует указать:

– `\theoremstyle{myplain}` — окружения с доказательствами или аксиомами

Таблица 2.5

Специальные окружения

Название окружения	Текстово-графический объект
abstr	реферат (abstract)
m-theorem	теорема
m-corollary	следствие
m-proposition	утверждение
m-lemma	лемма
m-axiom	аксиома
m-example	пример
m-definition	определение
m-condition	условие
m-problem	проблема
m-exercise	упражнение
m-question	вопрос
m-hypothesis	гипотеза

– `\theoremstyle{mydefinition}` — окружения, не связанные с доказательствами или аксиомами.

2. В команде создания окружения следует ввести краткий псевдоним (`m-new-env`) и отображаемое в pdf имя окружения (Название_окружения):

– `\newtheorem{m-new-env-second}{Название_окружения} - [chapter]`.

Теорема 2.1 (о чем-то конкретном). *Текст теоремы полностью выделен курсивом. Допустимо математические символы не выделять курсивом, если это искажает их значения. Используется абзацный отступ, так как “Абзацы в тексте начинают отступом” в соответствии с ГОСТ 2.105–95. Название теоремы допустимо убрать. Доказательство окончено.*

Доказательство теоремы 2.1, леммы, утверждений, следствий и других подобных окружений (в последнем абзаце) завершаем предложением в котором сказано, что доказательство окончено. Например, доказательство теоремы 2.1 окончено.

Тело доказательства не выделяется курсивом. Тело следующих окружений также не выделяется сплошным курсивом: определение, условие, проблема, пример, упражнение, вопрос, гипотеза и другие.

Определение 2.1 (термин). В тексте определения только *важные термины* выделяются курсивом. Если определение носит лишь вспомогательный характер, то допустимо не использовать окружение `m-definition`, представляя текст опре-

деления в обычном абзаце. Ключевые термины при этом обязательно выделяются курсивом.

Вместо теоремо-подобных окружений для вставки небольших текстово-графических объектов иногда используются команды. Типичным примером такого подхода является команда `\footnote{text}`¹², где в аргументе `text` указывают текст *подстрочной ссылки (сноски)*. В них *нельзя добавлять веб-ссылки или цитировать литературу*. Для этих целей используется список литературы. Нумерация сносок сквозная по ВКР без точки на конце выставляется в шаблоне автоматически, однако в каждом приложении к ВКР нумерация, зависящая от номера приложения, выставляется префикс «П», например «П1.1» — первая сноска первого приложения.

2.4. Выводы

Текст заключения ко второй главе. Пример ссылок [**Article**; 14; 15; 17; 20—22; 25; 28; 31; 32; 34; 35; 44], а также ссылок с указанием страниц, на котором отображены те или иные текстово-графические объекты [37, с. 96] или в виде мультицитаты на несколько источников [37, с. 96; 26, с. 46]. Часть библиографических записей носит иллюстративный характер и не имеет отношения к реальной литературе.

Короткое имя каждого библиографического источника содержится в специальном файле `my_biblio.bib`, расположенном в папке `my_folder`. Там же находятся исходные данные, которые с помощью программы `Viber` и стилевого файла `Biblatex-GOST` [18] приведены в списке использованных источников согласно ГОСТ 7.0.5-2008. Многообразные реальные примеры исходных библиографических данных можно посмотреть по ссылке [19].

Как правило, ВКР должна состоять из четырех глав. Оставшиеся главы можно создать по образцу первых двух и подключить с помощью команды `\input` к исходному коду ВКР. Далее в приложении 1 приведены краткие инструкции запуска исходного кода ВКР [36; 43].

В приложении 2 приведено подключение некоторых текстово-графических объектов. Они оформляются по приведенным ранее правилам. В качестве номера

¹²Внимание! Команда вставляется непосредственно после слова, куда вставляется сноска (без пробела). Лишние пробелы также не указываются внутри команды перед и после фигурных скобок.

структурного элемента вместо номера главы используется «П» с номером главы. Текстово-графические объекты из приложений не учитываются в реферате.

ГЛАВА 3. НАЗВАНИЕ ТРЕТЬЕЙ ГЛАВЫ: РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Хорошим стилем является наличие введения к главе. Во введении может быть описана цель написания главы, а также приведена краткая структура главы.

3.1. Название параграфа

3.2. Название параграфа

3.3. Выводы

Текст выводов по главе 3.

ГЛАВА 4. НАЗВАНИЕ ЧЕТВЁРТОЙ ГЛАВЫ. АПРОБАЦИЯ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ, А ИМЕННО: МЕТОДА, АЛГОРИТМА, МОДЕЛИ ИССЛЕДОВАНИЯ

Хорошим стилем является наличие введения к главе. Во введении может быть описана цель написания главы, а также приведена краткая структура главы.

4.1. Название параграфа

4.2. Название параграфа

Пример ссылки на литературу [1; 9; 11; 33].

4.3. Выводы

Текст выводов по главе 4.

ЗАКЛЮЧЕНИЕ

Заключение (2 – 5 страниц) обязательно содержит выводы по теме работы, *конкретные предложения и рекомендации* по исследуемым вопросам. Количество общих выводов должно вытекать из количества задач, сформулированных во введении выпускной квалификационной работы.

Предложения и рекомендации должны быть органически увязаны с выводами и направлены на улучшение функционирования исследуемого объекта. При разработке предложений и рекомендаций обращается внимание на их обоснованность, реальность и практическую приемлемость.

Заключение не должно содержать новой информации, положений, выводов и т. д., которые до этого не рассматривались в выпускной квалификационной работе. Рекомендуются писать заключение в виде тезисов.

Последним абзацем в заключении можно выразить благодарность всем людям, которые помогали автору в написании ВКР.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

DOI Digital Object Identifier.

WoS Web of Science.

ВКР Выпускная квалификационная работа.

ТГ-объект Текстово-графический объект.

СЛОВАРЬ ТЕРМИНОВ

TeX — язык вёрстки текста и издательская система, разработанные Дональдом Кнутом.

LaTeX — язык вёрстки текста и издательская система, разработанные Лэсли Лампортом как надстройка над TeX.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Автономова Н. С. Философский язык Жака Деррида. — М.: Российская политическая энциклопедия (РОССПЭН), 2011. — 510 с. — (Сер.: Российские Пропилеи).
2. Беклемищева Н., Жуков В., Михайлов К. О некоторых вопросах информационной безопасности медицинских устройств // Актуальные проблемы авиации и космонавтики. — 2017. — № 13.
3. ГОСТ Р 50922-2006. Защита информации. Основные термины и определения. — URL: <http://docs.cntd.ru/document/gost-r-50922-2006> (дата обращения: 10.06.2020).
4. ГОСТ Р 53114-2008. — URL: <http://docs.cntd.ru/document/gost-r-53114-2008> (дата обращения: 10.06.2020).
5. Единая система конструкторской документации. Общие требования к текстовым документам [текст]: ГОСТ 2.105–95. — Взамен ГОСТ 2.105—79, ГОСТ 2.906—71 ; введ. 1996—07—01. — Минск : Межгос. совет по стандартизации, метрологии и сертификации, 2002. — 31 с. — (Сер.: Межгосударственный стандарт).
6. Защита для чайников. Интернет вещей хотят обезопасить от кибератак. — URL: <https://rg.ru/2016/10/13/evrokomissii-predlozila-zashchitit-internet-veshchej-ot-kiberatak.html> (дата обращения: 10.06.2020).
7. Интернет вещдоков. Власти разработали концепцию развития сетей IoT. — URL: <https://www.kommersant.ru/doc/3924324> (дата обращения: 10.06.2020).
8. Информация – Большая советская энциклопедия. — URL: <https://gufo.me/dict/bse/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D1%8F> (дата обращения: 10.06.2020).
9. Котельников И. А., Чеботаев П. З. LaTeX по-русски. — 3-е изд. — Новосибирск: Сибирский Хронограф, 2004. — 496 с. — URL: <http://www.tex.uniyar.ac.ru/doc/kotelnikovchebotaev2004b.pdf> (дата обращения: 06.03.2019).
10. Музыкантский А., Фурин В. Лекции по криптографии. — 2-е изд. — Москва: МЦНМО, 2013. — 68 с.
11. Песков Н. В. Поиск информативных фрагментов описаний объектов в задачах распознавания: дис. . . . канд. физ.-мат. наук: 05.13.17 / Песков Николай Владимирович. — М., 2004. — 102 с.

12. *Яценко В. В.* Введение в криптографию. — 4-е изд. — Москва: МЦНМО, 2012. — 348 с.
13. Author and editor guide to prepare and submit the academic SPbPU editions to Clarivate Analytics: Book Citation Index Web of Science / V. Parkhomenko [et al.]. — 2018. — URL: https://github.com/ParkhomenkoV/SPbPU-BCI-template/blob/master/Author_guide_SPbPU-BCI.pdf (visited on 06.03.2019).
14. *Babington P.* The title of the work. Vol. 4. — 3rd ed. — The address: The name of the publisher, 1993. — 255 p. — (Ser.: 10).
15. *Badiou A.* Briefings on Existence: A Short Treatise on Transitory Ontology / ed. and trans. from the French, with an introd., by N. Madarasz. — NY: SUNY Press, 2006. — 190 p. — URL: https://books.google.ru/books?id=7HNkAT%5C_NFksC (visited on 05.12.2017).
16. *Bauer F. L.* Decrypted Secrets: Methods and Maxims of Cryptology. — 4th ed. — Berlin: Springer, 2007. — 555 p.
17. *Caxton P.* The title of the work. — The address of the publisher, 1993. — 255 p.
18. *Domanov O.* BibLATEX support for GOST standard bibliographies. — URL: <https://ctan.org/pkg/biblatex-gost> (visited on 06.03.2019).
19. *Domanov O.* Biblatex-GOST examples. — URL: <http://ctan.altspu.ru/macros/latex/contrib/biblatex-contrib/biblatex-gost/doc/biblatex-gost-examples.pdf> (visited on 06.03.2019).
20. *Draper P.* The title of the work // The title of the book. Vol. 4 / ed. by T. editor. — The organization. The address of the publisher: The publisher, 1993. — (Ser.: 5).
21. *Eston P.* The title of the work // Book title. Vol. 4. — 3rd ed. — The address of the publisher: The name of the publisher, 1993. — Chap. 8 — P. 201–213. — (Ser.: 5).
22. *Farindon P.* The title of the work // The title of the book. Vol. 4 / ed. by T. editor. — 3rd ed. — The address of the publisher: The name of the publisher, 1993. — Chap. 8 — P. 201–213. — (Ser.: 5).
23. *Feuersanger C., Tantau T.* The TikZ and PGF packages. — URL: <https://ctan.org/pkg/pgf> (visited on 06.03.2019).
24. *Fiorio C.* The algorithm2e package. — URL: <https://ctan.org/pkg/algorithm2e> (visited on 06.03.2019).
25. *Gainsford P.* The title of the work / The organization. — 3rd ed. — The address of the publisher, 1993. — 255 p.

26. *Ganter B., Wille R.* Formal concept analysis: mathematical foundations. — Springer, Berlin, 1999. — 284 p.
27. Government response to the Regulatory proposals for consumer Internet of Things (IoT) security consultation. — URL: <https://www.gov.uk/government/consultations/consultation-on-regulatory-proposals-on-consumer-iot-security/outcome/government-response-to-the-regulatory-proposals-for-consumer-internet-of-things-iot-security-consultation> (visited on 10.06.2020).
28. *Harwood P.* The title of the work: Master's thesis / Harwood Peter. — The address of the publisher: The school where the thesis was written, 1993. — 255 p.
29. HP Study Reveals 70 Percent of Internet of Things Devices Vulnerable to Attack. — URL: <https://www8.hp.com/us/en/hp-news/press-release.html?id=1744676> (visited on 10.06.2020).
30. Internet Of Things (iot). — URL: <https://www.gartner.com/en/information-technology/glossary/internet-of-things> (visited on 10.06.2020).
31. *Isley P.* The title of the work. — 1993.
32. *Joslin P.* The title of the work: diss. ... PhD in Engineering / Joslin Peter. — The address of the publisher: The school where the thesis was written, 1993. — 255 p.
33. *Kotelnikov I. A., Chebotaev P. Z.* LaTeX in Russian. — 3rd ed. — Novosibirsk: Sibiskiy Hronograph, 2004. — 496 p. — URL: <http://www.tex.uniyar.ac.ru/doc/kotelnikovchebotaev2004b.pdf> (visited on 06.03.2019); (in Russian).
34. *Lambert P.* The title of the work: tech. rep. / The institution that published. — The address of the publisher, 1993. — 255 p. — No. 2.
35. *Marcheford P.* The title of the work. — 1993.
36. MiKTeX web site. — URL: <https://miktex.org/> (visited on 06.03.2019).
37. Notes on relation between symbolic classifiers / X. Naidenova [et al.] // CEUR Workshop Proceedings / ed. by K. S. Watson B.W. — 2017. — Vol. 1921. — P. 88–103. — URL: <http://ceur-ws.org/Vol-1921/paper9.pdf> (visited on 19.12.2017).
38. *Peskov N. V.* Searching for informative fragments of object descriptions in the recognition tasks: diss. ... cand. phys.-math. sci.: 05.13.17 / Peskov Nickolay Vladimirovich. — M., 2004. — 102 p. — (in Russian).
39. *Schneier B.* Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C (cloth). — 1st ed. — John Wiley, Sons, Inc, 1996. — 1027 p.
40. Secrecy (Obscurity) is a Valid Security Layer. — URL: <https://danielmiessler.com/study/security-by-obscurity/> (visited on 10.06.2020).

41. SPbPU photo gallery. — URL: <http://www.spbstu.ru/media/photo-gallery/> (visited on 06.03.2019).

42. State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating. — URL: <https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/> (visited on 10.06.2020).

43. TeXstudio web site. — URL: <https://www.texstudio.org/> (visited on 06.03.2019).

44. The title of the work. Vol. 4 / ed. by P. Kidwelly. — The organization. The address of the publisher: The name of the publisher, 1993. — 255 p. — (Ser.: 5).

Приложение 1

Краткие инструкции по настройке издательской системы L^AT_EX

В SPbPU-BCI-template автоматически выставляются необходимые настройки и в исходном тексте шаблона приведены примеры оформления текстово-графических объектов, поэтому авторам достаточно заполнить имеющийся шаблон текстом главы (статьи), не вдаваясь в детали оформления, описанные далее. Возможный «быстрый старт» оформления главы (статьи) под Windows следующий^{П1.1}:

- A. Установка полной версии MikTeX [36]. В процессе установки лучше выставить параметр доустановки пакетов «на лету».
- B. Установка TexStudio [43].
- C. Запуск TexStudio и компиляция `my_chapter.tex` с помощью команды «Build&View» (например, с помощью двойной зелёной стрелки в верхней панели). Иногда, для достижения нужного результата необходимо несколько раз скомпилировать документ.
- D. В случае, если не отобразилась библиография, можно
 - воспользоваться командой Tools → Commands → Biber, затем запустив Build&View;
 - настроить автоматическое включение библиографии в настройках Options → Configure TexStudio → Build → Build&View (оставить по умолчанию, если сборка происходит слишком долго): `txs:///pdflatex | txs:///biber | txs:///pdflatex | txs:///pdflatex | txs:///view-pdf`.

В случае возникновения ошибок, попробуйте скомпилировать документ до последних действий или внимательно ознакомьтесь с описанием проблемы в log-файле. Бывает полезным переход (по подсказке TexStudio) в нужную строку в pdf-файле или запрос с текстом ошибки в поисковиках. Наиболее вероятной проблемой при первой компиляции может быть отсутствие какого-либо установленного пакета L^AT_EX.

В случае корректной работы настройки «установка на лету» все дополнительные пакеты будут скачиваться и устанавливаться в автоматическом режиме. Если доустановка пакетов осуществляется медленно (несколько пакетов за один запуск

^{П1.1} Внимание! Пример оформления подстрочной ссылки (сноски).

компилятора), то можно попробовать установить их в ручном режиме следующим образом:

1. Запустите программу: меню → все программы → MikTeX → Maintenance (Admin) → MiKTeX Package Manager (Admin).
2. Пользуясь поиском, убедитесь, что нужный пакет присутствует, но не установлен (если пакет отсутствует воспользуйтесь сначала MiKTeX Update (Admin)).
3. Выделив строку с пакетом (возможно выбрать несколько или вообще все неустановленные пакеты), выполните установку Tools → Install или с помощью контекстного меню.
4. После завершения установки запустите программу MiKTeX Settings (Admin).
5. Обновите базу данных имен файлов Refresh FNDB.

Для проверки текста статьи на русском языке полезно также воспользоваться настройками Options → Configure TexStudio → Language Checking → Default Language. Если русский язык «ru_RU» не будет доступен в меню выбора, то необходимо вначале выполнить Import Dictionary, скачав из интернета любой русскоязычный словарь.

Далее приведены формулы (П1.2), (П1.1), рис.П1.2, рис.П1.1, табл.П1.2, табл.П1.1.

$$\pi \approx 3,141. \quad (\text{П1.1})$$



Рис.П1.1. Вид на гидробашню СПбПУ [41]

Представление данных для сквозного примера по ВКР [38]

G	m_1	m_2	m_3	m_4	K
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2

П1.1. Параграф приложения

П1.1.1. Название подпараграфа

Название подпараграфа оформляется с помощью команды `\subsection{...}`.
Использование подподпараграфов в основной части крайне не рекомендуется.

$$\pi \approx 3,141. \quad (\text{П1.2})$$



Рис.П1.2. Вид на гидробашню СПбПУ [41]

Представление данных для сквозного примера по ВКР [38]

G	m_1	m_2	m_3	m_4	K
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2

Приложение 2

Некоторые дополнительные примеры

В приложении^{П2.1} приведены формулы (П2.2), (П2.1), рис.П2.2, рис.П2.1, табл.П2.2, табл.П2.1

$$\pi \approx 3,141.$$

(П2.1)



Рис.П2.1. Вид на гидробашню СПбПУ [41]

Таблица П2.1

Представление данных для сквозного примера по ВКР [38]

G	m_1	m_2	m_3	m_4	K
g_1	0	1	1	0	1
g_2	1	2	0	1	1
g_3	0	1	0	1	1
g_4	1	2	1	0	2
g_5	1	1	0	1	2
g_6	1	1	1	2	2

^{П2.1}Внимание! Пример оформления подстрочной ссылки (сноски).

П2.1. Подраздел приложения

$$\pi \approx 3,141.$$

(П2.2)



Рис.П2.2. Вид на гидробашню СПбПУ [41]

Таблица П2.2

Представление данных для сквозного примера по ВКР [38]

<i>G</i>	<i>m</i> ₁	<i>m</i> ₂	<i>m</i> ₃	<i>m</i> ₄	<i>K</i>
<i>g</i> ₁	0	1	1	0	1
<i>g</i> ₂	1	2	0	1	1
<i>g</i> ₃	0	1	0	1	1
<i>g</i> ₄	1	2	1	0	2
<i>g</i> ₅	1	1	0	1	2
<i>g</i> ₆	1	1	1	2	2