

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**  
**МЕХАНИКИ И ОПТИКИ»**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**ПРИМЕР ОФОРМЛЕНИЯ ВКР МАГИСТРА**

Автор: Буздалов Максим Викторович \_\_\_\_\_

Направление подготовки: 01.04.02 Прикладная  
математика и информатика

Квалификация: Магистр

Руководитель: Шалыто А.А., проф., д.т.н. \_\_\_\_\_

**К защите допустить**

Руководитель ОП Парфенов В.Г., проф., д.т.н. \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Санкт-Петербург, 2019 г.

Студент Буздалов М.В.

Группа М4239 Факультет ИТиП

Направленность (профиль), специализация

Технологии разработки программного обеспечения

Консультанты:

а) Белашенков Н.Р., канд. физ.-мат. наук, без звания

\_\_\_\_\_

б) Беззубик В.В., без степени, без звания

\_\_\_\_\_

ВКР принята « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Оригинальность ВКР \_\_\_\_\_ %

ВКР выполнена с оценкой \_\_\_\_\_

Дата защиты « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

Секретарь ГЭК Павлова О.Н.

\_\_\_\_\_

Листов хранения \_\_\_\_\_

Демонстрационных материалов/Чертежей хранения \_\_\_\_\_

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**  
**МЕХАНИКИ И ОПТИКИ»**

**УТВЕРЖДАЮ**

Руководитель ОП  
проф., д.т.н. Парфенов В.Г. \_\_\_\_\_  
« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

**ЗАДАНИЕ**  
**НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ**

**Студент** Буздалов М.В.

**Группа** М4239 **Факультет** ИТиП

**Руководитель** Шалыто А.А., проф., д.т.н., главный научный сотрудник Университета ИТМО

**1 Наименование темы:** Пример оформления ВКР магистра

**Направление подготовки (специальность):** 01.04.02 Прикладная математика и информатика

**Направленность (профиль):** Технологии разработки программного обеспечения

**Квалификация:** Магистр

**2 Срок сдачи студентом законченной работы:** «31» мая 2019 г.

**3 Техническое задание и исходные данные к работе**

Требуется разработать стилевой файл для системы L<sup>A</sup>T<sub>E</sub>X, позволяющий оформлять бакалаврские работы и магистерские диссертации на кафедре компьютерных технологий Университета ИТМО. Стилиевой файл должен генерировать титульную страницу пояснительной записки, задание, аннотацию и содержательную часть пояснительной записки. Первые три документа должны максимально близко соответствовать шаблонам документов, принятым в настоящий момент на кафедре, в то время как содержательная часть должна максимально близко соответствовать ГОСТ 7.0.11-2011 на диссертацию.

**4 Содержание выпускной работы (перечень подлежащих разработке вопросов)**

Пояснительная записка должна демонстрировать использование наиболее типичных конструкций, возникающих при составлении пояснительной записки (перечисления, рисунки, таблицы, листинги, псевдокод), при этом должна быть составлена так, что демонстрируется корректность работы стилового файла. В частности, записка должна содержать не менее двух приложений (для демонстрации нумерации рисунков и таблиц по приложениям согласно ГОСТ) и не менее десяти элементов нумерованного перечисления первого уровня вложенности (для демонстрации корректности используемого при нумерации набора русских букв).

**5 Перечень графического материала (с указанием обязательного материала)**

Графические материалы и чертежи работой не предусмотрены

## **6 Исходные материалы и пособия**

- а) ГОСТ 7.0.11-2011 «Диссертация и автореферат диссертации»;
- б) С.М. Львовский. Набор и верстка в системе  $\text{\LaTeX}$ ;
- в) предыдущий комплект стилевых файлов, использовавшийся на кафедре компьютерных технологий.

## **7 Дата выдачи задания «01» сентября 2017 г.**

Руководитель ВКР \_\_\_\_\_

Задание принял к исполнению \_\_\_\_\_

«01» сентября 2017 г.

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**  
**МЕХАНИКИ И ОПТИКИ»**

**АННОТАЦИЯ**  
**ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

**Студент:** Буздалов Максим Викторович

**Наименование темы ВКР:** Пример оформления ВКР магистра

**Наименование организации, в которой выполнена ВКР:** Университет ИТМО

**ХАРАКТЕРИСТИКА ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**

1 Цель исследования: Разработка удобного стилевого файла  $\text{\LaTeX}$  для бакалавров и магистров кафедры компьютерных технологий.

2 Задачи, решаемые в ВКР:

- а) обеспечение соответствия титульной страницы, задания и аннотации шаблонам, принятым в настоящее время на кафедре;
- б) обеспечение соответствия содержательной части пояснительной записки требованиям ГОСТ 7.0.11-2011 «Диссертация и автореферат диссертации»;
- в) обеспечение относительного удобства в использовании — указание данных об авторе и научном руководителе один раз и в одном месте, автоматический подсчет числа тех или иных источников.

3 Число источников, использованных при составлении обзора: 4

4 Полное число источников, использованных в работе: 5

5 В том числе источников по годам:

Отечественных			Иностранных		
Последние 5 лет	От 5 до 10 лет	Более 10 лет	Последние 5 лет	От 5 до 10 лет	Более 10 лет
0	1	0	2	1	1

6 Использование информационных ресурсов Internet: нет

7 Использование современных пакетов компьютерных программ и технологий:

Пакеты компьютерных программ и технологий	Раздел работы
Пакет <code>tabularx</code> для чуть более продвинутых таблиц	1.1, Приложения А, Б
Пакет <code>biblatex</code> и программное средство <code>biber</code>	Список использованных источников

8 Краткая характеристика полученных результатов

Получился, надо сказать, практически неплохой стилевик. В 2015–2018 годах его уже использовали некоторые бакалавры и магистры. Надеюсь на продолжение.

## 9 Гранты, полученные при выполнении работы

Автор разрабатывал этот стилевик исключительно за свой счет и на добровольных началах. Однако значительная его часть была бы невозможна, если бы автор не написал в свое время кандидатскую диссертацию в  $\text{\LaTeX}$ , а также не отвечал за формирование кучи научно-технических отчетов по гранту, известному как «5-в-100», что происходило при государственной финансовой поддержке ведущих университетов Российской Федерации (субсидия 074-U01).

## 10 Наличие публикаций и выступлений на конференциях по теме работы

По теме этой работы я (к счастью!) ничего не публиковал. Однако покажу, как можно ссылаться на свои публикации из списка литературы:

- 1 Буздалов М. В. Генерация тестов для олимпиадных задач по программированию с использованием генетических алгоритмов // Научно-технический вестник СПбГУ ИТМО. — 2011. — 2(72). — С. 72–77.
- 2 *Buzdalov M., Shalyto A.* Hard Test Generation for Augmenting Path Maximum Flow Algorithms using Genetic Algorithms: Revisited // Proceedings of IEEE Congress on Evolutionary Computation. — 2015. — P. 2121–2128.

Студент            Буздалов М.В. \_\_\_\_\_

Руководитель    Шалыто А.А.        \_\_\_\_\_

« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

## ОГCONTENTSЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	5
ГЛАВА 1. Первая глава.....	6
1.1. Таблицы.....	6
1.2. Рисунки.....	6
1.3. Листинги.....	7
ГЛАВА 2. Проверка сквозной нумерации.....	8
Выводы по главе 2.....	8
ЗАКЛЮЧЕНИЕ .....	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	10
ПРИЛОЖЕНИЕ А. Пример приложения .....	11
ПРИЛОЖЕНИЕ Б. Еще один пример приложения с неимоверно длинным названием для тестирования переносов.	13
ПРИЛОЖЕНИЕ В. Пример огромного листинга .....	14

## **ВВЕДЕНИЕ**

Для некоторых систем эффективное межпроцессное взаимодействие является критически важной частью их работы. Требование к минимизации времени обработки запросов может напрямую следовать из области применения системы, как в случае с финансовыми системами для алгоритмической торговли на финансовых рынках. Обработка запросов множеством логически связанных процессов может быть существенно ускорена при размещении таких процессов на одном физическом узле и при применении наиболее эффективных методов межпроцессного взаимодействия.



## ГЛАВА 1. ПЕРВАЯ ГЛАВА

Пример ссылок в рамках обзора: [1, 3–5]. Вне обзора: [2].

### 1.1. Таблицы

В качестве примера таблицы приведена таблица 1.

Таблица 1 – Таблица умножения (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68

Есть еще такое окружение `tabularx`, его можно аккуратно растянуть на всю страницу. Приведем пример (таблица 2).

Таблица 2 – Таблица умножения с помощью `tabularx` (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68

### 1.2. Рисунки

Пример рисунка (с помощью `TikZ`) приведен на рисунке 1. Под `pdflatex` можно также использовать `*.jpg`, `*.png` и даже `*.pdf`, под `latex` можно использовать `Metapost`. Последний можно использовать и под `pdflatex`, для чего в стилевике продекларированы номера картинок от 1 до 20.

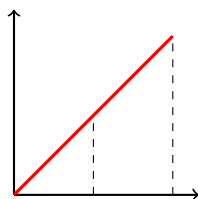


Рисунок 1 – Пример рисунка

### 1.3. Листинги

В работах студентов кафедры «Компьютерные технологии» часто встречаются листинги. Листинги бывают двух основных видов — исходный код и псевдокод. Первый оформляется с помощью окружения `lstlisting` из пакета `listings`, который уже включается в стилевике и немного настроен. Пример Hello World на Java приведен на листинге 1. Пример большого листинга — в приложении (листинг В.1).

Листинг 1 – Пример исходного кода на Java

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Псевдокод можно оформлять с помощью разных пакетов. В данном стилевике включается пакет `algorithmicx`. Сам по себе он не генерирует флагов, поэтому для них используется пакет `algorithm`. Пример их совместного использования приведен на листинге 2.

Листинг 2 – Пример псевдокода

```
function IsPrime( $N$ )
  for  $t \leftarrow [2; \lfloor \sqrt{N} \rfloor]$  do
    if  $N \bmod t = 0$  then
      return false
    end if
  end for
  return true
end function
```

Наконец, листинги из `listings` тоже можно подвешивать с помощью `algorithm`, пример на листинге 3.

Листинг 3 – Исходный код и флаги `algorithm`

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

## ГЛАВА 2. ПРОВЕРКА СКВОЗНОЙ НУМЕРАЦИИ

Листинг 4 должен иметь номер 4.

Листинг 4 – Исходный код и флюид `algorithm`

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Рисунок 2 должен иметь номер 2.

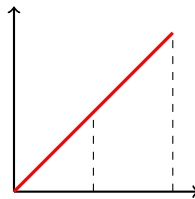


Рисунок 2 – Пример рисунка

Таблица 3 должна иметь номер 3.

Таблица 3 – Таблица умножения с помощью `tabularx` (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68

### Выводы по главе 2

В конце каждой главы желательно делать выводы. Вывод по данной главе — нумерация работает корректно, ура!

## **ЗАКЛЮЧЕНИЕ**

В данном разделе размещается заключение.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 *Буздалов М. В.* Генерация тестов для олимпиадных задач по программированию с использованием генетических алгоритмов // Научно-технический вестник СПбГУ ИТМО. — 2011. — 2(72). — С. 72–77.
- 2 *Bellman R. E.* Dynamic Programming. — Princeton, NJ : Princeton University Press, 1957. — 342 p.
- 3 *Buzdalov M., Doerr B., Kever M.* The Unrestricted Black-Box Complexity of Jump Functions // Evolutionary Computation. — 2016. — Accepted for publication.
- 4 *Buzdalov M., Shalyto A.* Hard Test Generation for Augmenting Path Maximum Flow Algorithms using Genetic Algorithms: Revisited // Proceedings of IEEE Congress on Evolutionary Computation. — 2015. — P. 2121–2128.
- 5 *Nebro A. J., Durillo J. J.* On the Effect of Applying a Steady-State Selection Scheme in the Multi-Objective Genetic Algorithm NSGA-II // Nature-Inspired Algorithms for Optimisation. — Springer Berlin Heidelberg, 2009. — P. 435–456. — (Studies in Computational Intelligence ; 193).

## ПРИЛОЖЕНИЕ А. ПРИМЕР ПРИЛОЖЕНИЯ

В приложениях рисунки, таблицы и другие подобные элементы нумеруются по приложениям с соответствующим префиксом. Проверим это.

Листинг А.1 должен иметь номер А.1.

Листинг А.1 – Исходный код и флюат `algorithm`

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Рисунок А.1 должен иметь номер А.1.

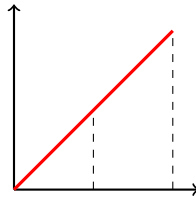


Рисунок А.1 – Пример рисунка

Таблица А.1 должна иметь номер А.1.

Таблица А.1 – Таблица умножения с помощью `tabularx` (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68

Заодно проверим нумерованные и ненумерованные перечисления. Ненумерованные:

- пункт А;
- пункт Б;
- пункт В.

Нумерованные списки нескольких уровней:

- а) первый элемент;
- б) второй элемент с подэлементами:
  - 1) первый подэлемент;

2) второй подэлемент;

3) третий подэлемент.

в) третий элемент;

г) четвертый элемент;

д) пятый элемент;

е) шестой элемент;

ж) седьмой элемент;

и) восьмой элемент;

к) девятый элемент;

л) десятый элемент.

**ПРИЛОЖЕНИЕ Б. ЕЩЕ ОДИН ПРИМЕР ПРИЛОЖЕНИЯ С  
НЕИМОВЕРНО ДЛИННЮЩИМ НАЗВАНИЕМ ДЛЯ  
ТЕСТИРОВАНИЯ ПЕРЕНОСОВ**

Проверим на примере таблиц, что нумерация в приложениях — по приложениям. Таблица Б.1 должна иметь номер Б.1.

Таблица Б.1 – Таблица умножения с помощью `tabularx` (фрагмент)

–	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34
3	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51
4	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68



## ПРИЛОЖЕНИЕ В. ПРИМЕР ОГРОМНОГО ЛИСТИНГА

Листинг В.1 – Пример большого листинга

```
import java.util.*;

public class Example {
    static int[] restoreOutgoing(int[] g, int[] outgoing,
                                int vertex, int mask) {
        int[] rv = new int[1 + Integer.bitCount(mask)];
        int n = g.length;
        int current = rv.length - 1;
        while (true) {
            rv[current] = vertex;
            if (current == 0) {
                if (vertex != 0) {
                    throw new AssertionError();
                }
                return rv;
            }
            mask ^= 1 << (vertex - 1);
            int prevMask = outgoing[mask] & g[vertex];
            if (prevMask == 0) {
                throw new AssertionError();
            }
            vertex = Integer.numberOfTrailingZeros(prevMask);
            --current;
        }
    }

    static int[] restoreIncoming(int[] g, int[] incoming,
                                int vertex, int mask) {
        int[] rv = new int[1 + Integer.bitCount(mask)];
        int n = g.length;
        int current = 0;
        while (true) {
            rv[current] = vertex;
            if (current == rv.length - 1) {
                if (vertex != 0) {
                    throw new AssertionError();
                }
                return rv;
            }
            mask ^= 1 << (vertex - 1);
            int prevMask = incoming[mask] & g[vertex];
            if (prevMask == 0) {
                throw new AssertionError();
            }
            vertex = Integer.numberOfTrailingZeros(prevMask);
            ++current;
        }
    }
}
```

```
mask ^= 1 << (vertex - 1);
int nextMask = incoming[mask] & g[vertex];
if (nextMask == 0) {
    throw new AssertionError();
}
vertex = Integer.numberOfTrailingZeros(nextMask);
++current;
}
}
}
```