

Министерство образования и науки Российской Федерации
ФГАОУ ВПО «УрФУ имени первого Президента России Б. Н. Ельцина»
Институт радиоэлектроники и информационных технологий - РтФ
Департамент информационных технологий и автоматики

Имитационное моделирование в системе
Bizagi Process Modeler

ОТЧЕТ
по лабораторной работе

Преподаватель: Клебанов Борис Исаевич
Студент: Сухоплюев Илья Владимирович
Группа: РИ-440001

Екатеринбург
2017

Содержание

| | |
|---|----|
| Введение | 3 |
| 1 Аналитический раздел | 4 |
| 1.1 Анализ того и сего | 4 |
| 1.2 Существующие подходы к созданию всячины | 5 |
| 2 Конструкторский раздел | 8 |
| 2.1 Архитектура всячины | 8 |
| 2.2 Подсистема всякой ерунды | 8 |
| 2.2.1 Блок-схема всякой ерунды | 8 |
| 3 Технологический раздел | 10 |
| 4 Экспериментальный раздел | 12 |
| Заключение | 13 |
| Список использованных источников | 14 |
| А Картинки | 15 |
| Б Еще картинки | 16 |

Введение

В лабораторной работе проводится знакомство с моделью оказания государственной услуги *«Подготовка предложений о представлении к награждению знаком отличия Свердловской области «Совет да любовь» (Дфлее, услуга Совет да любовь)*. На основе этой услуги изучаются вопросы имитационного моделирования с использованием промышленного стандарта BPMN.

В лабораторной работе студенты знакомятся с моделью бизнес-процесса оказания услуг скорой помощи. На его основе изучаются важные вопросы имитационного моделирования: составление модели с использованием промышленного стандарта BPMN 2.0, использование статистической информации для моделирования, анализ результатов моделирования и оптимизация бизнес-процесса. Работа выполняется с использованием программы Bizagi Process Modeler. Данное методическое пособие является адаптированным переводом официального руководства, доступного на сайте компании Bizagi <http://help.bizagi.com/>

Целью работы является создание всякой всячины. Для достижения поставленной цели необходимо решить следующие задачи:

- проанализировать существующую всячину;
- спроектировать свою, новую всячину;
- изготовить всякую всячину;
- проверить её работоспособность.

Вот так-то. А этот абзац вставлен для визуальной оценки отступа от перечня до следующего абзаца.

1 Аналитический раздел

В данном разделе анализируется и классифицируется существующая всячина и пути создания новой всячины. А вот отступ справа в 1 см. — это хоть и по ГОСТ, но ведь диагноз же...

1.1 Анализ того и сего

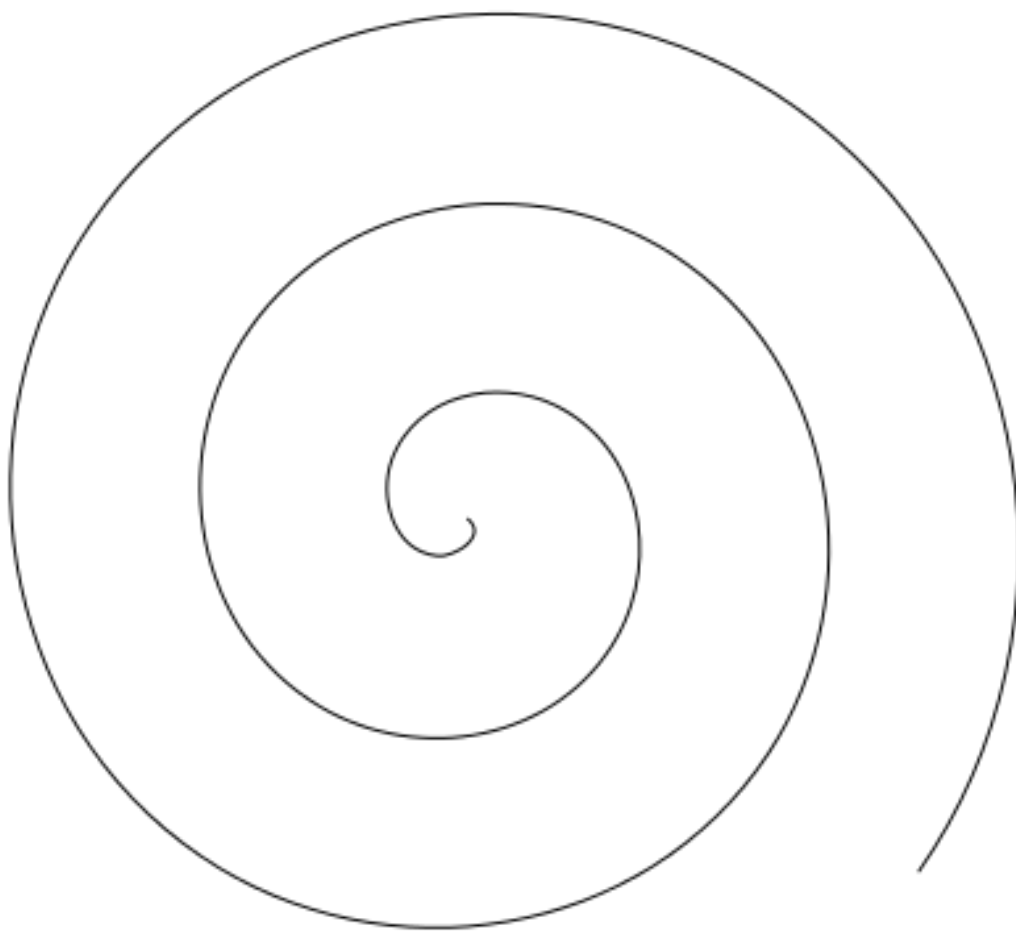


Рисунок 1.1 — Рисунок

В [?] указано, что...

Кстати, про картинки. Во-первых, для фигур следует использовать `[ht]`. Если и после этого картинки вставляются «не по ГОСТ», т.е. слишком далеко от места ссылки, — значит у вас в РПЗ **слишком мало текста!** Хотя и ужасный параметр `!ht` у окружения `figure` тоже

никто не отменял, только при его использовании документ получается страшный, как в ворде, поэтому просьба так не делать по возможности.

1.2 Существующие подходы к созданию всячины

Известны следующие подходы...

- а) Перечисление с номерами.
- б) Номера первого уровня. Да, ГОСТ требует именно так — сначала буквы, на втором уровне — цифры. Чуть ниже будет вариант «нормальной» нумерации и советы по её изменению. Да, мне так нравится: на первом уровне выравнивание элементов как у обычных абзацев. Проверим теперь вложенные списки.
 - 1) Номера второго уровня.
 - 2) Номера второго уровня. Проверяем на длиииииной-предлиииииииинной строке, что получается.... Сойдёт.
- в) По мнению Лукьяненко, человеческий мозг старается подвести любую проблему к выбору из трех вариантов.
- г) Четвёртый (и последний) элемент списка.

Теперь мы покажем, как изменить нумерацию на «нормальную», если вам этого захочется. Пара команд в начале документа поможет нам.

- 1) Изменим нумерацию на более привычную...
- 2) ... нарушим этим гост.
 - а) Но, пожалуй, так лучше.

В заключение покажем произвольные маркеры в списках. Для них нужен пакет **enumerate**.

- 1. Маркер с арабской цифрой и с точкой.
- 2. Маркер с арабской цифрой и с точкой.
 - I. Римская цифра с точкой.
 - II. Римская цифра с точкой.

В отчётах могут быть и таблицы — см. табл. 1.1 и 1.2. Небольшая таблица делается при помощи **tabular** внутри **table** (последний полностью аналогичен **figure**, но добавляет другую подпись).

Таблица 1.1 — Пример короткой таблицы с длинным названием на много длинных-длинных строк

| Тело | F | V | E | $F + V - E - 2$ |
|-----------|-----|------|-----|-----------------|
| Тетраэдр | 4 | 4 | 6 | 0 |
| Куб | 6 | 8 | 12 | 0 |
| Октаэдр | 8 | 6 | 12 | 0 |
| Додекаэдр | 20 | 12 | 30 | 0 |
| Икосаэдр | 12 | 20 | 30 | 0 |
| Эйлер | 666 | 9000 | 42 | $+\infty$ |

Для больших таблиц следует использовать пакет **longtable**, позволяющий создавать таблицы на несколько страниц по ГОСТ.

Для того, чтобы длинный текст разбивался на много строк в пределах одной ячейки, надо в качестве ее формата задавать **p** и указывать явно ширину: в мм/дюймах (**110mm**), относительно ширины страницы (**0.22\textwidth**) и т.п.

Можно также использовать уменьшенный шрифт — но, пожалуйста, тогда уж во **всей** таблице сразу.

Таблица 1.2 — Пример длинной таблицы с длинным названием на много длинных-длинных строк

| Вид шума | Громкость, дБ | Комментарий |
|--------------------------|---------------|---|
| Порог слышимости | 0 | |
| Шепот в тихой библиотеке | 30 | |
| Обычный разговор | 60-70 | |
| Звонок телефона | 80 | Конечно, это было до эпохи мобильных (внутри машины) |
| Уличный шум | 85 | |
| Гудок поезда | 90 | |

Продолжение на след. стр.

Продолжение таблицы 1.2

| | | |
|------------------------------|------------|---|
| Шум электрички | 95 | |
| Порог здоровой нормы | 90-95 | Длительное пребывание на более громком шуме может привести к ухудшению слуха |
| Мотоцикл | 100 | (модель бензокосилки) (Doom в целом вреден для здоровья) |
| Power Mower | 107 | |
| Бензопила | 110 | |
| Рок-концерт | 115 | |
| Порог боли | 125 | feel the pain |
| Клепальный молоток | 125 | (автор сам не знает, что это) |
| Порог опасности | 140 | Даже кратковременное пребывание на шуме большего уровня может привести к необратимым последствиям |
| Реактивный двигатель | 140 180 | Необратимое полное повреждение слуховых органов Интересно, почему?.. |
| Самый громкий возможный звук | 194 | |

2 Конструкторский раздел

В данном разделе проектируется новая всячина.

2.1 Архитектура всячины

Проверка параграфа. Вроде работает.

Вторая проверка параграфа. Опять работает.

Вот.

- Это список с «палочками».
- Хотя он и не по ГОСТ, кажется.

1) Поэтому для списка, начинающегося с заглавной буквы, лучше список с цифрами.

Формула 2.1 совершенно бессмысленна.

$$a = cb \tag{2.1}$$

Окружение `cases` опять работает (см. 2.2), спасибо И. Короткову за исправления..

$$a = \begin{cases} 3x + 5y + z, & \text{если хорошо} \\ 7x - 2y + 4z, & \text{если плохо} \\ -6x + 3y + 2z, & \text{если совсем плохо} \end{cases} \tag{2.2}$$

2.2 Подсистема всякой ерунды

Культурная вставка dot-файлов через утилиту `dot2tex` (рис. 2.1).

2.2.1 Блок-схема всякой ерунды

Кстати о заголовках

У нас есть и **subsubsection**. Только лучше её не нумеровать.

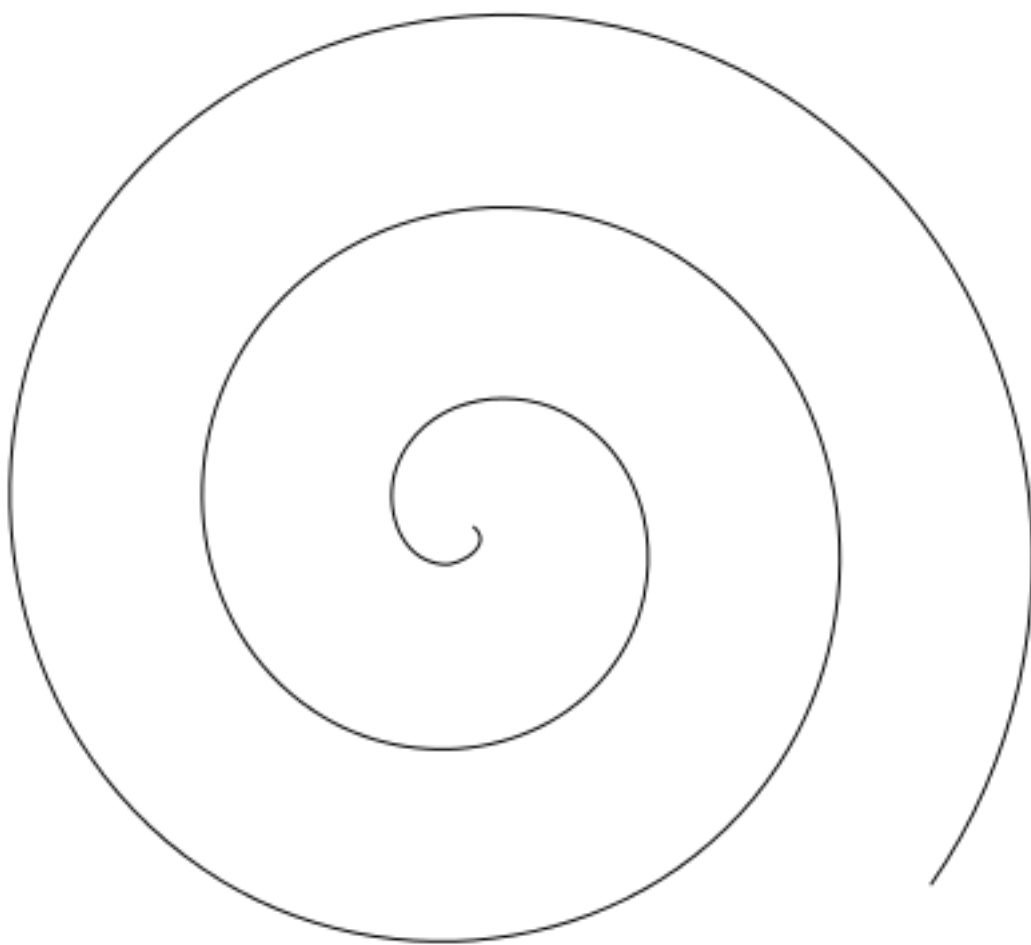


Рисунок 2.1 — Рисунок

3 Технологический раздел

В данном разделе описано изготовление и требование всячины. Кстати, в Latex нужно эскейпить подчёркивание (писать «`some_function`» для `some_function`).

Для вставки кода есть пакет `listings`. К сожалению, пакет `listings` всё ещё работает криво при появлении в листинге русских букв и кодировке исходников utf-8. В данном примере он (увы) на лету конвертируется в koi-8 в ходе сборки pdf.

Есть альтернатива `listingsutf8`, однако она работает лишь с `\lstinputlisting`, но не с окружением `\lstlisting`

Вот так можно вставлять псевдокод (питоноподобный язык определен в `listings.inc.tex`):

Листинг 3.1 — Алгоритм оценки дипломных работ

```
1 def EvaluateDiplomas():
2     for each student in Masters:
3         student.Mark ← 5
4     for each student in Engineers:
5         if Good(student):
6             student.Mark ← 5
7         else:
8             student.Mark ← 4
```

Еще в шаблоне определен псевдоязык для BNF:

Листинг 3.2 — Грамматика

```
1 ifstmt → "if" "(" expression ")" stmt |
2         "if" "(" expression ")" stmt1 "else" stmt2
3 number → digit digit *
```

В листинге 3.3 работают русские буквы. Сильная магия. Однако, работает только во включаемых файлах, прямо в `TeX` нельзя.

Листинг 3.3 — Пример (`test.c`)

```
1 #include <stdio.h>
2 int main()
3 {
4     return 0;
5 }
```

Можно также использовать окружение **verbatim**, если **listings** чем-то не устраивает. Только следует помнить, что табы в нём «съедаются». Существует так же команда `\verbatiminput` для вставки файла.

```
a_b = a + b; // русский комментарий
if (a_b > 0)
    a_b = 0;
```

4 Экспериментальный раздел

В данном разделе проводятся вычислительные эксперименты. А на рис. 4.1 показана схема мыслительного процесса автора...

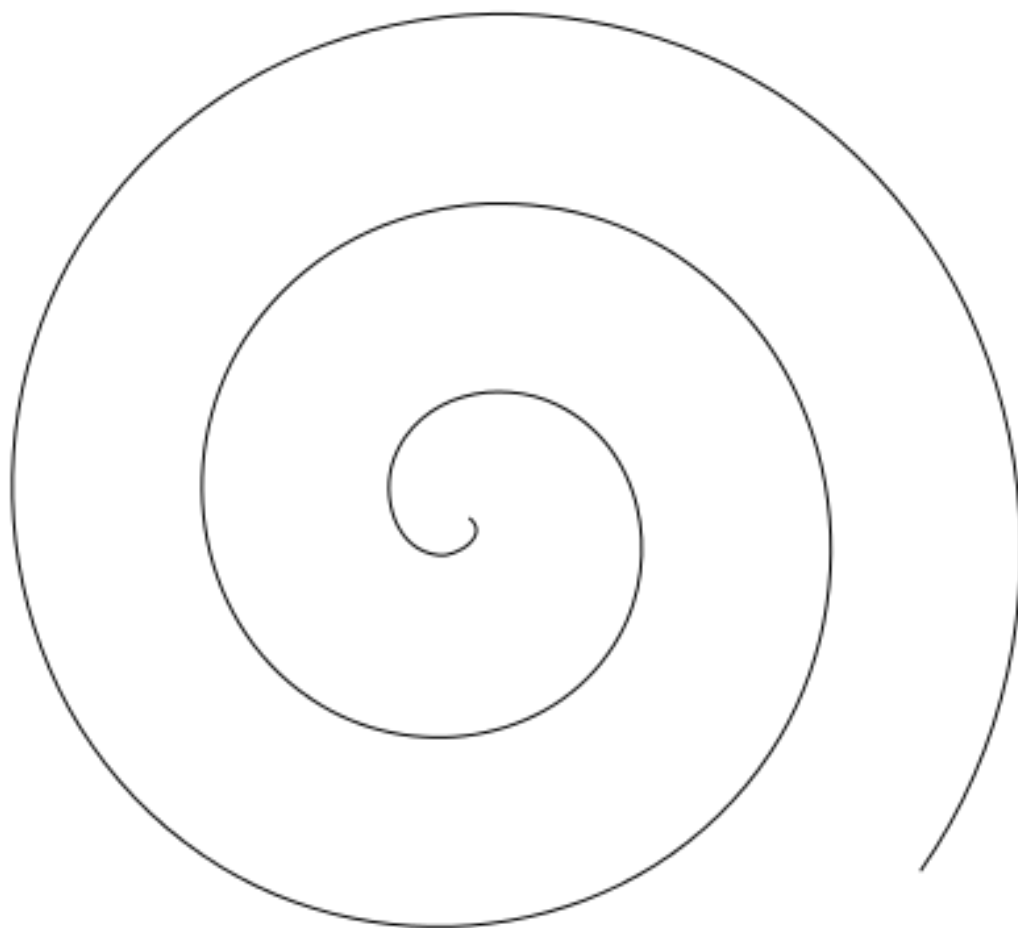


Рисунок 4.1 — Как страшно жить

Заключение

В результате проделанной работы стало ясно, что ничего не ясно...

Список использованных источников

Приложение А Картинки

Рисунок А.1 — Картинка в приложении. Страшная и ужасная.

Приложение Б Еще картинки

Рисунок Б.1 — Еще одна картинка, ничем не лучше предыдущей. Но
надо же как-то заполнить место.