

«

»

“ ”

“ ” . . . . .  
\_\_\_\_\_ .

# РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ Логическое программирование

: 09.03.04

, :

: 4, : 7

,

		<b>7</b>
<b>1</b>	( )	3
<b>2</b>		108
<b>3</b>	, .	65
<b>4</b>	, .	18
<b>5</b>	, .	0
<b>6</b>	, .	36
<b>7</b>	, .	8
<b>8</b>	, .	2
<b>9</b>	, .	9
<b>10</b>	, .	43
<b>11</b>	( , , )	
<b>12</b>		

( ): 09.03.04

229 12.03.2015 ., : 01.04.2015 .

: 1, ,

( ): 09.03.04

, 6 20.06.2017

, 6 21.06.2017

:

, . .

:

, . . . . . . . .

:

. . .

# 1.

1.1

<b>Компетенция ФГОС: ОПК.3 готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов; в части следующих результатов обучения:</b>	
8.	
<b>Компетенция ФГОС: ПК.1 готовность применять основные методы и инструменты разработки программного обеспечения; в части следующих результатов обучения:</b>	
13.	
4.	
9.	

# 2.

2.1

(	
---	--

<b>.1. 4</b>	
1.знать языки программирования и среды разработки	;
<b>.1. 13</b>	
2.знать логическую парадигму программирования	;
<b>.1. 9</b>	
3.уметь применять выбранные языки программирования для написания программного кода	;
<b>.3. 8</b>	
4.владеть декларативным подходом к программированию	;

# 3.

3.1

	,	.		
: 7				
:				
1.	0	2	1, 2	,
2.	0	2	1, 2	,
:				
3.	0	2	1, 2	,

4.	.	0	2	1, 2	, .
5.	: .	0	2	1, 2	, .
6.	. .	0	2	1, 2	, .
7.	PROLOG .	0	2	1, 2, 3, 4	, .
:					
8.	.	0	2	1, 2	, .
: .					
9.	(Constraint Logic Programming).	0	2	1, 2	, .

3.2

	, .			
:7				
: , , , .				
1.	.	1	4	3,4 PROLOG; , , ; ; .
2.	.	1	4	3,4 PROLOG; ; ; ; .

3. ( )	1	4	3, 4	PROLOG; ; ; ; ;
: .				
4. : .	1	4	3, 4	PROLOG; ; : ; : ; ; ;
5. : .	1	4	3, 4	PROLOG; ; ( ); ; .

6.	:	1	4	3, 4	PROLOG; ; ( ); ; .
7.	.	1	4	3, 4	; PROLOG; , , ; ; ;
: .					
8.	.	1	4	3, 4	; PROLOG; , , ; ; ;
: .					

9.	(Constraint Logic Programming).	0	4	3, 4	PROLOG; , , ; ; ;
----	---------------------------------	---	---	------	----------------------------------

3.3

		, .		
: 7				
: , , ,				
1.	: , , .	0	10	1, 2, 3, 4 - ,
: .				
2.	.	0	13	1, 2, 3, 4 - .
: .				
3.	.	0	5	1, 2, 3, 4 - .
: .				
4.	.	0	5	1, 2, 3, 4 - .

4.

--	--	--	--	--

: 7				
1		1, 2, 3, 4	10	5
<p>[ ]: - / . . ; . . . - . - , [2011]. - : http://elibrary.nstu.ru/source?bib_id=vtls000162637. - . . . [ ]: - / . . . ; . . . - . - , [2015]. - : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - . . .</p>				
2		1, 2, 3, 4	37	4
<p>3.3 : [ ]: - / . . ; . . . - . - , [2011]. - : http://elibrary.nstu.ru/source?bib_id=vtls000162637. - . . . [ ]: - / . . ; . . . - . - , [2015]. - : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - . . .</p>				

## 5.

, ( . 5.1).

5.1

	-
	e-mail; ;
	e-mail
	e-mail; ;

5.2

1	
<b>Краткое описание применения:</b> Обсуждение задач, решаемых на лабораторных работах.	

## 6.

( ),

- 15- ECTS.

. 6.1.

6.1

: 7		
Лабораторная №2:	3	6
<p>" . . . , [2015]. - [ ]: - : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - . . .</p>		



Лабораторная №3:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №4:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №5:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №6:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №7:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №8:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
Лабораторная №9:	3	6
"/ ; " , [2015]. - [ ] : http://elibrary.nstu.ru/source?bib_id=vtls000221771. - .		
РГЗ:	6	12
Экзамен:	20	40

6.2

6.2

		/		
3	8.	+	+	+
1	13.		+	+
	4.		+	+
	9.	+	+	+

1

7.

1. Братко И. Алгоритмы искусственного интеллекта на языке Prolog. пер. с англ. / Иван Братко. - М. [и др.], 2004. - 637 с. : ил.
2. Шрайнер П. А. Основы программирования на языке Пролог : курс лекций : учебное пособие для вузов по специальностям информационных технологий / П. А. Шрайнер. - М., 2005. - 172, [1] с. : ил.
3. Сергиевский Г. М. Функциональное и логическое программирование : учебное пособие [для вузов по направлению "Информатика и вычислительная техника"] / Г. М. Сергиевский, Н. Г. Волочёнков. - М., 2010. - 317, [1] с. : ил, табл.

4. Авдеенко Т. В. Введение в искусственный интеллект и логическое программирование : учебное пособие / Т. В. Авдеенко ; Новосиб. гос. техн. ун-т. - Новосибирск, 2007. - 62, [2] с. : ил. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000068970](http://elibrary.nstu.ru/source?bib_id=vtls000068970)

1. Ходашинский И. А. Язык ПРОЛОГ в примерах и задачах : учебное пособие / И. А. Ходашинский ; Федер. агентство по образованию ; Том. гос. ун-т систем упр. и радиоэлектроники. - Томск, 2006. - 279 с. : ил.

2. Стерлинг Л. Искусство программирования на языке Пролог / Л. Стерлинг, Э. Шапиро ; пер. с англ. С. Ф. Сопрунова и Л. В. Шабанова ; под ред. Дадаева Ю. Г. - М., 1990. - 333 с. : ил.

3. Ин Ц. М. Использование Турбо-Пролога : Пер. с англ.. - М., 1993. - 606 с. : ил.

4. Клоксин У. Программирование на языке Пролог : пер. с англ. / У. Клоксин, К. Меллиш. - М., 1987. - 336 с. : ил.

5. Программирование на языке Пролог : учебное пособие / [Г. В. Разумовский и др.] ; Ленингр. электротехн. ин-т им. В. И. Ульянова. - Л., 1991. - 59, [3] с.

6. Новицкая Ю. В. Функциональное и логическое программирование [Электронный ресурс] : учебно-методический комплекс / Новицкая Ю. В. ; Новосиб. гос. техн. ун-т. - Новосибирск, 2011. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000159514](http://elibrary.nstu.ru/source?bib_id=vtls000159514). - Загл. с этикетки диска.

-

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>

2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>

3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>

4. ЭБС "Znaniy.com" : <http://znaniy.com/>

5. :

## 8.

### 8.1

1. Авдеенко Т. В. Логическое программирование [Электронный ресурс] : учебно-методическое пособие / Т. В. Авдеенко ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000162637](http://elibrary.nstu.ru/source?bib_id=vtls000162637). - Загл. с экрана.

2. Новицкая Ю. В. Логическое программирование [Электронный ресурс] : электронный учебно-методический комплекс / Ю. В. Новицкая ; Новосиб. гос. техн. ун-т. - Новосибирск, [2015]. - Режим доступа: [http://elibrary.nstu.ru/source?bib\\_id=vtls000221771](http://elibrary.nstu.ru/source?bib_id=vtls000221771). - Загл. с экрана.

### 8.2

#### 1 SWI Prolog

## 9.

-

1	( Internet )	Internet



Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Новосибирский государственный технический университет»

Кафедра вычислительной техники

“УТВЕРЖДАЮ”  
ДЕКАН АВТФ  
к.т.н., доцент И.Л. Рева  
“    ”    \_\_\_\_\_    Г.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### УЧЕБНОЙ ДИСЦИПЛИНЫ

#### Логическое программирование

Образовательная программа: 09.03.04 Программная инженерия, профиль: Технологии разработки программного обеспечения

## 1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине **Логическое программирование** приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.5 способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	у10. уметь использовать специализированные программные средства при решении профессиональных задач	Декларативное программирование. Основы логического программирования. Логическая программа: факты, правила вывода, декларативная семантика, поиск с возвратом. Предложения: факты и правила вывода. Запросы. Поиск с возвратом и управлением им. Поиск всех возможных решений. Управление поиском с возвратом. Отсечение.	РГЗ	Зачет, вопросы 1÷5
ПК.10.В/ПТ готовность к разработке компонентов аппаратно-программных комплексов и баз данных с использованием современных инструментальных средств и технологий программирования	з11. знать логическую парадигму программирования	Рекурсия. Рекурсия: нехвостовая и хвостовая. Составные структуры данных. Рекурсивные структуры данных: списки и деревья. Графы. Представление и работа с графами. Практические примеры применения языка PROLOG для решения рекурсивных задач.	РГЗ	Зачет, вопросы 6÷11
ПК.10.В/ПТ	уб. владеть декларативным подходом к программированию	Динамические базы данных. Применение логического программирования в задачах искусственного интеллекта. Логическое программирование в ограничениях (Constraint Logic Programming).	РГЗ	Зачет, вопросы 12,13

## 2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 7 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОПК.5, ПК.10.В/ПТ.

Экзамен проводится в устной форме, по билетам.

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 7 семестре обязательным этапом текущей аттестации является расчетно-графическое задание

(работа) (РГЗ(Р)). Требования к выполнению РГЗ(Р), состав и правила оценки сформулированы в паспорте РГЗ(Р).

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе учебной дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.5, ПК.10.В/ПТ, за которые отвечает дисциплина, на разных уровнях.

#### **Общая характеристика уровней освоения компетенций.**

**Ниже порогового.** Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

**Пороговый.** Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

**Базовый.** Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

**Продвинутый.** Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

## Паспорт экзамена

по дисциплине «Логическое программирование», 7 семестр

### 1. Методика оценки

Экзамен проводится в устной форме, по билетам. Билет формируется по следующему правилу: вопросы выбираются из диапазона вопросов (список вопросов приведен ниже). В ходе зачет преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

### Форма билета для экзамена

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
Факультет АВТФ

Билет № \_\_\_\_\_

к экзамену по дисциплине «Логическое программирование»

---

Вопрос 1. Предикаты. Предложения: факты и правила. Запросы (цели). Переменные.  
Вопрос 2. Простые объекты данных. Составные объекты данных. Многоуровневые составные объекты данных.  
Задача.

Утверждаю: зав. кафедрой \_\_\_\_\_ должность, ФИО  
(подпись) (дата)

### 2. Критерии оценки

- Ответ на зачетный билет считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, при решении задачи допускает принципиальные ошибки, оценка составляет *менее 50 баллов*.
- Ответ на экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи допускает непринципиальные ошибки, например, вычислительные, оценка составляет *50-72 баллов*.
- Ответ на экзаменационный билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, законы, дает

характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики процессов, не допускает ошибок при решении задачи, оценка составляет 73-86 баллов.

- Ответ на экзаменационный билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, не допускает ошибок и способен обосновать выбор метода решения задачи, оценка составляет 87-100 баллов.

### 3. Шкала оценки

Экзамен считается сданным, если сумма баллов по всем заданиям билета оставляет не менее 20 баллов (из 40 возможных).

В общей оценке по дисциплине баллы за экзамен учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

#### 4. Вопросы к экзамену по дисциплине «Логическое программирование»

1. Предикаты. Предложения: факты и правила. Запросы (цели). Переменные.
2. Сопоставление и унификация. Предикат равенства. Детерминизм.
3. Основные принципы поиска с возвратом.
4. Управление поиском решений (предикаты fail и !).
5. Управление поиском решений (динамическое отсечение).
6. Простые объекты данных. Составные объекты данных. Многоуровневые составные объекты данных.
7. Предикат gereat.
8. Рекурсия. Хвостовая рекурсия.
9. Деревья, примеры работы.
10. Списки, примеры работы. Сортировка списков.
11. Графы: представление и действия над графами.
12. Динамические базы данных.
13. Применение логического программирования в задачах искусственного интеллекта.



## **Паспорт расчетно-графического задания (работы)**

по дисциплине «Логическое программирование», 7 семестр

### **1. Методика оценки**

В рамках расчетно-графического задания (работы) по дисциплине студенты должны показать владение декларативным подходом к программированию.

При выполнении расчетно-графического задания (работы) студенты должны написать и отладить программу в соответствии с заданием на языке логического программирования.

Структура:

- титульный лист;
- задание на расчетно-графическое задание;
- содержание;
- введение;
- описание назначения программного продукта;
- описание данных;
- описание методов решения;
- описание программы;
- описание пользовательского интерфейса;
- список использованных источников;
- приложение (исходные коды с комментариями).

Пример оформления титульного листа.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
Кафедра вычислительной техники

РАСЧЕТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ  
ПО ДИСЦИПЛИНЕ «ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ»

Студент: \_\_\_\_\_

Группа: \_\_\_\_\_

Преподаватель: Новицкая Ю.В.

Срок представления к защите: 17 учебная неделя

Представлено к защите: \_\_\_\_\_

Новосибирск 20\_\_ г.

## 2. Критерии оценки

- работа считается выполненной **на пороговом** уровне, если задание выполнено в минимальном объеме, не использован декларативный подход к написанию программы, оценка составляет 50-72 баллов.
- работа считается выполненной **на базовом** уровне, если задание выполнено в базовом объеме, использован декларативный подход к написанию программы, оценка составляет 73-86 баллов.
- работа считается выполненной **на продвинутом** уровне, если задание выполнено в продвинутом объеме, использован декларативный подход к написанию программы, оценка составляет 87-100 баллов.

## 3. Шкала оценки

В общей оценке по дисциплине баллы за РГЗ(Р) учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

## 4. Примерный перечень тем РГЗ(Р)

Вариант 1. Поиск пути в лабиринте.

Разработать программу для поиска пути в лабиринте. Предусмотреть возможность задания входа и выхода, наглядное представление найденных решений.

Вариант 2. Разбиение текста.

Разработать программу для разбиения текста, находящегося в файле, на строки длиной около 80 символов с переносом слов.

Исходный и обработанный тексты хранятся в файле.

Для разбиения слова на части для переноса использовать следующие правила:

- две идущие подряд гласные можно разделить, первой из них предшествует согласная, а за второй идет хотя бы одна буква (буква й при этом рассматривается вместе с предшествующей гласной как единое целое);
- две идущие подряд согласные можно разделить, если первой из них предшествует гласная, а той части слова, которая идет за второй согласной, имеется хотя бы одна гласная (буквы ь, ъ вместе с предшествующей согласной рассматриваются как единое целое);
- если не удастся применить указанные выше два пункта, то следует попытаться разбить слово так, чтобы первая часть содержала более чем одну букву и оканчивалась на гласную, а вторая содержала хотя бы одну гласную.

Вероятность правильного разбиения увеличивается, если предварительно воспользоваться хотя бы неполным списком приставок, содержащих гласные, и попытаться, прежде всего, выделить из слова такую приставку.

Вариант 3. Калькулятор.

Разработать программу-калькулятор.

Калькулятор позволяет использовать четыре арифметических действия, для выполнения расчетов можно использовать скобки.

Вариант 4. Игра «Угадывание слова».

Разработать программу для игры «Угадывание слова». Слова для угадывания хранятся в файле. Предусмотреть подсчет числа попыток и ограничить их количество.

Вариант 5. Игра «Жизнь».

Разработать программу для игры «Жизнь». Игра моделирует жизнь поколений гипотетической колонии живых клеток, которые выживают, размножаются или погибают в соответствии со следующими правилами. Клетка выживает, если и только если она имеет двух или трех соседей из восьми возможных. Если у клетки только один сосед или вовсе ни одного, она погибает в изоляции. Если клетка имеет четырех или более соседей, она погибает от перенаселения. В любой пустой позиции, у которой ровно три соседа, в следующем поколении появляется новая клетка. Предусмотреть задание размеров поля и случайную или ручную расстановку клеток.

Вариант 6. Игра «Крестики-нолики».

Разработать программу для игры в крестики-нолики. Для игрового поля можно задавать произвольные размеры.

Вариант 7. Справочная система авиакомпании.

Разработать справочно-информационную систему «Справочная система авиакомпании». Система должна содержать базу данных со следующей информацией: номер рейса, пункт отправления, пункт назначения, тип самолета, время отправления, время прибытия, дни выполнения рейсов, цена билета. Система должна подбирать рейсы (с учетом стыковок рейсов в течение одних суток) по минимальной стоимости билетов. Разработать средства для работы с базой данных: загрузка базы, редактирование записей в базе, добавление (с проверкой на дублирование) и удаление записей, сохранение базы.

Вариант 8. Игра «Быки и коровы».

Разработать программу для игры «Быки и коровы». Требуется угадать случайное

четырёхзначное число. Называются пробные цифры. Если цифра по значению и позиции совпала с цифрой в исходном числе, то эта цифра – «корова». Если же цифра совпала по значению, но не совпала по позиции с цифрой в исходном числе, то эта цифра – «бык». Например, если загадано число 1294, а названо число 1429, то это одна «корова» и три «быка».

Вариант 9. Справочная система железнодорожной компании.

Разработать справочно-информационную систему «Справочная система железнодорожной компании». Система должна содержать базу данных со следующей информацией: номер рейса, пункт отправления, пункт назначения, тип вагона, время отправления, время прибытия, дни выполнения рейсов, цена билета. Система должна подбирать рейсы (с учетом пересадок в течение одних суток) по минимальному времени в пути. Разработать средства для работы с базой данных: загрузка базы, редактирование записей в базе, добавление (с проверкой на дублирование) и удаление записей, сохранение базы.

Вариант 10. Разработать программу для работы с двоичными деревьями.

Реализовать следующие функции: балансировку дерева, вывод дерева в форме, представленной на рисунке, преобразование дерева из представления ( $\text{tree}(\text{Root}, \text{Left}, \text{Right})$ ) в представление  $1(2(4(7,8(9,10)),5),3(6,(11)))$ .

Вариант 11. Условия задачи.

1. Есть 5 домов, каждый разного цвета.
2. В каждом доме живет один человек, отличающийся от соседнего по национальности: немец, англичанин, швед, датчанин, норвежец.
3. Каждый пьет только один определенный напиток, курит определенную марку сигарет и держит определенное животное.
4. Никто из 5 человек не пьет одинаковые с другими напитки, не курит одинаковые сигареты и не держит одинаковое животное.

Вопрос: кому принадлежит рыба?

Подсказки:

1. Англичанин живет в красном доме.
2. Швед держит собаку.
3. Датчанин пьет чай.
4. Зеленый дом стоит слева от белого.
5. Жилец зеленого дома пьет кофе.
6. Человек, который курит Pall Mall, держит птицу.
7. Жилец из среднего дома пьет молоко.
8. Жилец из желтого дома курит Dunhill.
9. Норвежец живет в первом доме.
10. Курильщик Marlboro живет около того, кто держит кошку.
11. Человек, который держит лошадь, живет около того, кто курит Dunhill.
12. Курильщик сигарет Winfield пьет пиво.
13. Норвежец живет около голубого дома.
14. Немец курит Rothmans.

15. Курильщик Marlboro живет по соседству с человеком, который пьет воду.

Вариант 12. Разработать программу для работы с графами.

Реализовать следующие функции: нахождение остовного дерева связного графа с минимальной стоимостью, вывод графа на экран в удобной форме, взаимное преобразование трех различных представлений графа.

Вариант 13. Разработать программу для раскрашивания плоской карты.

Раскрасить плоскую карту четырьмя цветами так, чтобы никакие две смежные области на ней не были окрашены в одинаковые цвета. Предусмотреть возможность задания карты.

Вариант 14. Три миссионера и три каннибала.

Три миссионера и три каннибала. находятся на берегу реки. Здесь же небольшая лодка, вмещающая двух человек. Все хотят перебраться на другой берег. Если на каком-либо берегу миссионеров окажется больше, чем каннибалов, то миссионеры обратят каннибалов в свою веру, в обратном случае миссионерам грозит беда. Найти последовательность поездок, гарантирующую безопасность миссионерам и свободу вероисповедания каннибалам.

Пять ревнивых мужей.

Во время наводнения пять супружеских пар оказались отрезанными от суши водой. В их распоряжении была одна лодка, которая одновременно могла вместить только трех человек. Каждый супруг был настолько ревнив, что не мог позволить находиться своей супруге в лодке или на любом берегу с другим мужчиной или мужчинами в его отсутствие. Найти способ переправить на сушу мужчин и их жен.

Вариант 15. Разработать предикаты для работы со списками.

Проверить, является ли список палиндромом.

Преобразовать многоуровневый список в одноуровневый.

Удалить повторяющиеся элементы списка.  $[a,a,a,b,c,c,a,a,d,e,e,e,e] \Rightarrow [a,b,c,a,d,e]$

Поместить повторяющиеся элементы списка в подсписки.  $[a,a,a,b,c,c,a,a,d,e,e,e,e] \Rightarrow [[a,a,a,a],[b],[c,c],[a,a],[d],[e,e,e,e]]$

Выполнить преобразование списка.  $[a,a,a,b,c,c,a,a,d,e,e,e,e] \Rightarrow [[4,a],[1,b],[2,c],[2,a],[1,d],[4,e]]$

Выполнить преобразование списка.  $[a,a,a,b,c,c,a,a,d,e,e,e,e] \Rightarrow [[4,a],b,[2,c],[2,a],d,[4,e]]$

Выполнить обратное преобразование списка.  $[[4,a],b,[2,c],[2,a],d,[4,e]] \Rightarrow [a,a,a,b,c,c,a,a,d,e,e,e,e]$

Дублировать все элементы списка.

Дублировать элементы списка, количество дублей задается.  $[a,b,c] \Rightarrow [a,a,a,b,b,b,c,c,c]$ .

Создать список, получаемый случайной перестановкой элементов списка.

Сортировать список списков в соответствии с длиной подсписков. Выполнить сортировку подсписков.

$[[b,a,c],[e,d],[f,h,g],[d,e],[k,l,i,j],[n,m],[o]] \Rightarrow [[i,j,k,l],[o],[a,b,c],[f,g,h],[d,e],[d,e],[m,n]]$