СИСТЕМЫ АВТОМАТИЗИРОВАННОГО РАСЧЁТА В УПРАВЛЕНИИ КАЧЕСТВОМ И ПРИ ЗАЩИТЕ ИНФОРМАЦИИ

♦ ИЗДАТЕЛЬСТВО ТГТУ ♦

Министерство образования и науки Российской Федерации

ГОУ ВПО «Тамбовский государственный технический университет»

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО РАСЧЁТА В УПРАВЛЕНИИ КАЧЕСТВОМ И ПРИ ЗАЩИТЕ ИНФОРМАЦИИ

Лабораторные работы
для студентов специальностей 220501, 200503, 220301

Тамбов
Издательство ТГТУ
2009

УДК 65(075)
ББК В192.2я75-5я73+У9(2)301-823.2я73+з973-018.2я73
Б20

Рецензенты:

Директор ФГУ «Тамбовский ЦСМ» кандидат технических наук
С.В. Григорьева

Директор ИДО ТГТУ кандидат технических наук
С.Н. Кузьмин

Составители:

П.В. Балабанов
С.В. Пономарёв

Б20 Системы автоматизированного расчёта в управлении качеством и при защите информации : лабораторные работы / сост. :П.В. Балабанов, С.В. Пономарёв. – Тамбов : Изд-во Тамб. гос. техн. ун-та, 2009. – 32 с. – 100 экз.

Представлен материал, необходимый студентам при выполнении ими лабораторных работ по дисциплине «Информационные технологии в управлении качеством и защита информации».

Предназначены для студентов, обучающихся по специальностям 220501, 200503, 220301; могут быть использованы студентами, изучающими программу MatLab.

УДК 65(075)
ББК В192.2я75-5я73+У9(2)301-823.2я73+з973-018.2я73

Учебное издание

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО РАСЧЁТА В УПРАВЛЕНИИ КАЧЕСТВОМ И ПРИ ЗАЩИТЕ ИНФОРМАЦИИ

Лабораторные работы

Составители:

БАЛАБАНОВ Павел Владимирович,
ПОНОМАРЁВ Сергей Васильевич

Редактор З.Г. Чернова
Инженер по компьютерному макетированию Т.Ю. Зотова

Подписано в печать 10.02.2009
Формат 60 × 84 / 16. 1,86 усл. печ. л. Тираж 100 экз. Заказ № 44

Издательско-полиграфический центр
Тамбовского государственного технического университета
392000, Тамбов, Советская, 106, к. 14

АВТОМАТИЗАЦИЯ ОБРАБОТКИ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ СРЕДСТВАМИ MATLAB

Цель работы: закрепить навыки работы с программным продуктом MatLab.

Задание

1. Загрузить экспериментальные данные в программу MatLab.
2. Построить графики зависимости $T = f(\tau)$ для показаний каждого из датчиков, где T – температура; τ – время.
3. Построить графики зависимости $T = f(x)$ для заданного момента времени, где x – продольная координата по оси реактора.
4. Вычислить максимальное показание каждого из датчиков.
5. Вычислить значение $\int_0^L T(x)\,dx$ для заданного момента времени.

Методические указания

В ходе технологического процесса очистки газа в реакторе I (рис. 1.1) происходит химическая реакция взаимодействия компонентов газовой смеси ГС, подлежащих удалению, с веществом шихты II, наполняющей реактор. В результате химической реакции происходит нагрев шихты, температура которой измеряется датчиками *1, 2, 3, 4*, установленными на равных расстояниях друг от друга и от входа в реактор. Сигнал с датчиков через модуль ввода МВ поступает в компьютер ПК и сохраняется в файле temp.txt. Файл temp.txt содержит пять колонок, в первой из которых записано время в секундах, в остальных четырёх – температура, регистрируемая датчиками *1*, *2*, *3* и *4*. Общая длина реактора составляет L = 0,16 м.

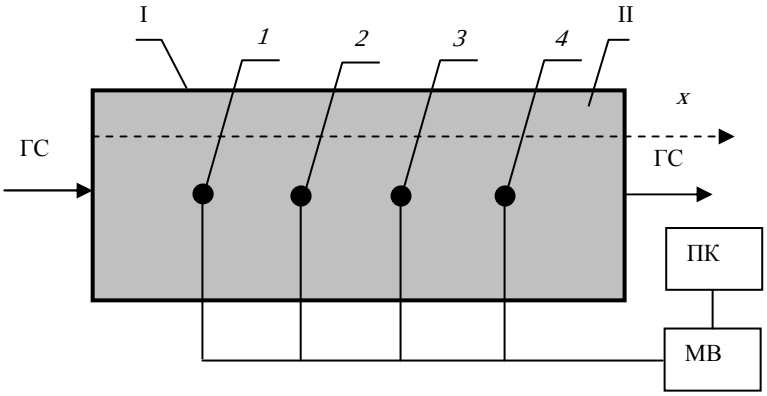


Рис. 1.1. Схема установки

Для загрузки данных из файла в MatLab можно использовать функцию **load('Имя файла')**, возвращающую массив данных из файла в заданную переменную. Например, оператор **T=load('temp.txt')** позволяет записать в переменную **T** данные из файла с именем **temp.txt.**

После загрузки данных из файла к ним можно обратиться, используя принятые в MatLab методы обращения к векторам и матрицам. Например, для обращения к элементам первого столбца массива **T** используют оператор **T(:,1)**, для обращения к элементам второй строки используют оператор **T(2,:)**, для обращения к элементу, стоящему в первом столбце второй строки используют оператор **T(2,1)**.

После загрузки данных в массив **T** следует построить графики зависимости $T = f(\tau)$ для показаний каждого из датчиков. Для построения двухмерных графиков в MatLab можно использовать функцию **plot(x,y,'s')**, где **x, y –** строки данных, **'*s*'** – параметр, задающий цвет, тип линий графиков, а также тип точек на графике (табл. 1.1). При построении графиков функции $T = f(\tau)$ не следует забывать, что в массиве **T** данные по времени и температуре содержатся в столбцах.

Для построения графиков зависимости $T = f(x)$ в произвольные моменты времени вначале требуется аппроксимировать исходные данные по координате *x*. Для проведения аппроксимации можно последо-

вательно использовать вначале функцию **polyfit(x,T,n)**, возвращающую строку коэффициентов аппроксимирующего полинома **n** степени, а затем функцию **polyval(P,xi)**, возвращающую строку значений полиномиальной функции в точках с координатами **xi**. В приведенных функциях **x** – строка данных, содержащая координаты датчиков температуры; **T** – строка данных, содержащая показания датчиков в заданный момент времени; **P** – строка значений коэффициентов аппроксимирующего полинома.

1.1. Значения строкового параметра s

Цвет		Тип точки		Тип линии	
y	жёлтый	**.**	точка	**-**	– сплошная линия
m	сиреневый	**o**	кружок	**--**	– пунктирная линия
c	голубой	**x**	крестик	**-.**	– штрих-пунктирная линия
r	красный	*****	звездочка	**:**	– штриховая линия
b	синий	**+**	плюс		
g	зелёный	**s**	квадрат		
w	белый	**d**	ромб		
k	чёрный	**v**	треугольник		

Аппроксимируйте исходные данные полиномами 1 – 5 степени и найдите такое **n**, при котором погрешность аппроксимации будет минимальной.

Рассмотрим пример, когда датчики расположены на расстоянии 0,1; 0,2; 0,3; 0,4 м от входа в реактор. В этом случае строка **x** запишется в виде **x=[0.1 0.2 0.3 0.4]**.

В строку **Tr**, запишем показания датчиков температуры в *i*-й момент времени:

```
for i=1:1:length(T)
Tr(i,1)= T(i,2);
Tr(i,2)= T(i,3);
Tr(i,3)= T(i,4);
Tr(i,4)= T(i,5);
end
```

Далее вычисляем коэффициенты аппроксимирующего полинома третьей степени для экспериментальных данных, записанных, например, в 10 строке массива **Tr** и записываем их в строку **P**:

P = **polyfit(x,Tr(10,:),3)**.

Задаём координаты точек, в которых следует вычислить значения полиномиальной функции:

xi=[0.1:0.01:0.4].

Вычисляем значения полиномиальной функции в заданных точках:

Tappr=polyval(P, xi).

Далее можно построить график функции $T = f(x)$ для момента времени, значение которого записано в ячейке **T(10,1)**:

plot(xi,Tappr).

Для вычисления максимального показания каждого из датчиков можно применить функцию **max(R)**, которая возвращает максимальный элемент строки **R**.

Для вычисления интеграла $\int_0^L T(x)\,dx$ в произвольный момент времени можно использовать функцию **trapz(x,y)**, где **x** – строка, содержащая диапазон изменения переменной интегрирования, т.е. **x=[0:L]**; **y** – известная подынтегральная функция $T(x)$, полученная путём аппроксимации исходных данных полиномом *n*-й степени.

1.2. Варианты заданий для самостоятельной работы

Вариант	Заданный для расчёта момент времени τ, с	Вариант	Заданный для расчёта момент времени τ, с
1	2.3	9	3252.46
2	199.760	10	3359.51
3	322.96	11	4143.13
4	1311.450	12	4192.18
5	589.460	13	4362.67
6	4801.69	14	4745.94
7	2240.46	15	4802.24
8	2318.34	16	5068.19

Результаты выполнения лабораторной работы

Показания датчиков температуры в момент времени _____с

№ датчика	1	2	3	4
Температура, °С				

При аппроксимации исходных данных полиномом степени _____ получено выражение для аппроксимирующего полинома (записать выражение в виде $T(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$)

_____

Максимальное показание датчиков

№ датчика	1	2	3	4
Температура, °С				

Значение $\int_0^L T(x)dx$=_____ для момента времени _____с.

Контрольные вопросы

1. Привести примеры использования функций MatLab, позволяющих загружать данные из файлов.
2. Привести примеры использования функций MatLab, позволяющих загружать данные из переменных MatLab в файлы.
3. Как провести аппроксимацию данных в MatLab?
4. Привести примеры вычисления определённых интегралов в MatLab.
5. Какие функции можно использовать в MatLab для нахождения максимальных, минимальных и среднеквадратичных значений массива данных?
6. Какие функции MatLab используются для построения двумерных, трёхмерных графиков в декартовой системе координат, а также в полярной системе координат?

Лабораторная работа 2

АВТОМАТИЗАЦИЯ ВЫПОЛНЕНИЯ РАСЧЁТОВ ПРИ МЕТРОЛОГИЧЕСКОЙ ОБРАБОТКЕ РЕЗУЛЬТАТОВ ИСПЫТАНИЙ

Цель работы: закрепить навыки использования MatLab при проведении метрологической обработки результатов измерений.

Задание

1. Написать программу на языке MatLab, предназначенную для метрологической обработки результатов испытаний.
2. Провести метрологическую обработку экспериментальных данных.

Методические указания

При проведении серии опытов, результатами которых является набор результатов измерений, актуальной задачей является оценка погрешности измерений. Для расчётов погрешностей существуют строгие алгоритмы метрологической обработки результатов испытаний, которые можно запрограммировать на любом из языков программирования. Это позволит не только ускорить обработку результатов испытаний, но и снизить вероятность возникновения ошибок, неизбежных при обработке большого объёма экспериментальных данных, уменьшить трудозатраты на расчёты и, как следствие, повысить производительность труда.

Сформулируем задачу и рассмотрим алгоритм её решения.

Пусть в результате серии из n экспериментов по измерению физической величины y был получен набор её возможных значений y_i, $i=\overline{1,n}$. Требуется определить доверительный интервал, в котором с заданной вероятностью лежит измеренная величина y, а также относительную погрешность её измерения.

Алгоритм решения поставленной задачи заключается в следующем.

1. Вычисляем среднее арифметическое результатов измерений по формуле $\bar{y}=\sum_{i=1}^{n} y_i / n$.

2. Вычисляем среднее квадратичное отклонение результатов измерений по формуле $S_y=\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2/(n-1)}$.

3. Вычисляем среднее квадратичное отклонение среднего арифметического результатов измерений по формуле $S_{\bar{y}} = S_y / \sqrt{n}$.

4. Определяем границы доверительного интервала среднего арифметического результатов измерений по формуле $t_p S_{\bar{y}}$, где t_p – коэффициент Стьюдента, определяемый по таблицам при заданной доверительной вероятности.

5. Результаты измерений представляем в виде $y = \bar{y} \pm t_p S_{\bar{y}}$.

Программа, предназначенная для метрологической обработки результатов испытаний, должна отвечать следующим требованиям.

1. Исходные данные (экспериментальные данные) должны загружаться из текстового файла.

2. Вначале программы с клавиатуры, в зависимости от числа экспериментальных данных, вводится значение коэффициента Стьюдента.

Для ввода с клавиатуры значения переменной tp в MatLab можно использовать выражение **t_p=input ('Введите tp')**.

Варианты заданий для самостоятельной работы

Сгенерируйте исходные данные, воспользовавшись следующим выражением MatLab

y = a + sqrt(b) · randn(c,1) .

2.1. Значения параметров *a, b, c* по вариантам

Вариант	Значения параметров			Вариант	Значения параметров		
	a	*b*	*c*		*a*	*b*	*c*
1	10	1	20	9	15	3	25
2	5	0,3	30	10	65	8	35
3	100	5	50	11	4	0,5	45
4	200	10	60	12	90	10	55
5	0,9	0,03	70	13	35	7	65
6	1,5	0,4	80	14	0,7	0,1	75
7	50	2	90	15	0,3	0,05	85
8	1000	45	100	16	76	8	95

Результаты выполнения лабораторной работы

$\bar{y}$	S_y	$S_{\bar{y}}$	t_p	$t_p S_{\bar{y}}$

Контрольные вопросы

1. Привести функцию MatLab для вычисления среднего арифметического значения результатов измерений.

2. Для чего применяется функция **randn ()** в MatLab.

3. Привести функцию MatLab для вычисления среднего квадратичного отклонения результатов измерений.

4. Построить график распределения случайной величины *y* в соответствии с вариантом.

5. Как проверить гипотезу о нормальности распределения результатов измерений?

АВТОМАТИЗАЦИЯ ПОСТРОЕНИЯ ГИСТОГРАММЫ И ВЫЧИСЛЕНИЯ ПОКАЗАТЕЛЕЙ КАЧЕСТВА ПРОЦЕССА

Цель работы: Изучить технологию статистической обработки данных в MatLab.

Задание

1. По исходным данным, характеризующим протекание процесса, построить статистическую гистограмму.
2. Вычислить значения характеристик качества.

Методические указания

Построение гистограммы на практике производят для того, чтобы оценить качество выпускаемой продукции и качество процесса производства этой продукции. Для оценки качества процесса используют следующие характеристики:

P_p – индекс пригодности процесса удовлетворять технический допуск;

k – показатель настроенности процесса на целевое значение;

$P_{pk} = P_p(1-k)$ – оценка индекса пригодности процесса удовлетворять технический допуск с учётом положений среднего значения.

Приведённые выше индексы вычисляют по формулам:

$$P_p = \frac{USL - LSL}{6\sigma},$$

где USL, LSL – верхняя и нижняя границы поля допуска соответственно; σ – стандартное отклонение, вычисляемое по формуле

$$\sigma \approx \sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 / (n-1)};$$

$$k = \frac{\left|\bar{x} - Ц\right|}{(USL - LSL)/2},$$

где $Ц = (LSL + USL)/2$; x_i – результаты измерений характеристик процесса; $\bar{x}$ – среднее значение результатов измерений.

В лабораторной работе требуется построить гистограмму по данным варианта и вычислить значения характеристик качества процесса.

Рассмотрим примерный порядок выполнения работы.

Пусть на токарном станке было изготовлено 90 осей. Для исследования качества процесса изготовления осей были измерены их диаметры. Результаты измерений приведены в табл. 3.1.

Построим гистограмму по этим данным.

Определим количество интервалов n на гистограмме по следующей формуле:

$$n = 1 + 3{,}322 \lg N,$$

где N – общее количество собранных данных.

В нашем случае $n = 1 + 3{,}322 \lg 90 = 7{,}49 \approx 7$.

3.1. Результаты измерений

Номер наблюдений	Результаты наблюдений				
1 – 5	2,510	2,517	2,522	2,533	2,510
6 – 10	2,527	2,536	2,542	2,524	2,542
11 – 15	2,529	2,523	2,514	2,519	2,519
16 – 20	2,520	2,514	2,521	2,514	2,533
21 – 25	2,535	2,523	2,510	2,542	2,524
26 – 30	2,533	2,510	2,532	2,522	2,502
31 – 35	2,525	2,515	2,526	2,530	2,532
36 – 40	2,531	2,545	2,526	2,532	2,522
41 – 45	2,518	2,527	2,502	2,530	2,522
46 – 50	2,532	2,522	2,502	2,530	2,522
51 – 55	2,514	2,533	2,510	2,524	2,526
56 – 60	2,524	2,513	2,518	2,532	2,522
61 – 65	2,502	2,530	2,522	2,530	2,521
66 – 70	2,522	2,535	2,540	2,528	2,525
71 – 75	2,515	2,520	2,522	2,542	2,540
76 – 80	2,528	2,531	2,545	2,524	2,522
81 – 85	2,520	2,522	2,527	2,511	2,519
86 – 90	2,531	2,527	2,529	2,528	2,519

Вычислим диапазон данных по формуле

$$R = x_{\max} - x_{\min},$$

где $x_{\max}, x_{\min}$ – наибольшее и наименьшее наблюдаемое значение, соответственно.

В нашем случае $R = 2{,}545 - 2{,}502 = 0{,}043$ мм.

Определим размер (ширину) интервала по формуле

$$h = R / n.$$

В нашем случае $h = 0{,}043 / 7 = 0{,}006$ мм.

Определим нижнюю границу первого интервала. Она равна $x_{\min}$ = 2,502. Верхняя граница первого интервала равна $x_{\min} + h$, т.е. 2,502 + 0,006 = 2,508. Нижняя граница второго интервала равна верхней границе первого интервала, т.е. 2,508, а верхняя граница равна 2,508 + 0,006 = 2,514. Границы остальных пяти интервалов можно получить аналогичным образом. В результате таких вычислений получим следующие границы: 2,502; 2,508; 2,514; 2,520; 2,526; 2,533; 2,539; 2,545.

Теперь остаётся подсчитать, сколько наблюдений попадает в каждый из интервалов: в первый интервал – 4; во второй – 11; в третий – 12; в четвёртый – 26; в пятый – 24; в шестой – 2, в седьмой – 7.

Далее строим гистограмму (в нашем случае гистограмма будет иметь вид, представленный на рис. 3.1).

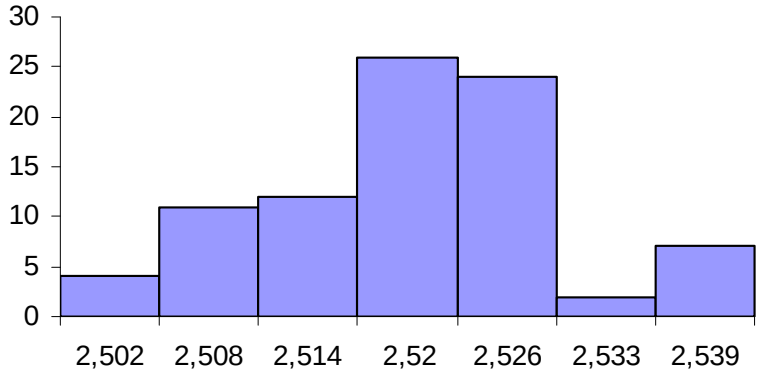


Рис. 3.1. Гистограмма

Автоматизируйте в MatLab алгоритм построения гистограммы, изложенный выше. Для вычисления максимальных и минимальных значений выборок используйте функции **min()** и **max()**. Для реализации алгоритма подсчёта числа попаданий результатов наблюдений в каждый интервал используйте возможности программирования на языке MatLab, а именно операторы цикла **for** и условный оператор **if**. Для построения гистограммы используйте функцию **bar()**. После построения гистограммы следует вычислить значения величин P_p, k и P_{pk}.

Варианты заданий для самостоятельной работы

Результаты измерений для построения гистограммы возьмите из лабораторной работы 2. Значения верхней и нижней границ поля допуска приведены в табл. 3.2.

3.2. Исходные данные по вариантам

Вариант	Значения параметров *USL*, *LSL*		Вариант	Значения параметров *USL*, *LSL*	
	USL	*LSL*		*USL*	*LSL*
1	10,8	9,1	9	18	12
2	5,3	4,7	10	73	57
3	105	95	11	4,5	3,5
4	210	190	12	100	80
5	0,93	0,87	13	42	28
6	1,9	1,1	14	0,8	0,6
7	52	48	15	0,35	0,25
8	1040	950	16	84	68

Результаты выполнения лабораторной работы

$x_{\min}$	$x_{\max}$	R	n	P_p	k	P_{pk}

Контрольные вопросы

1. Дать определение понятию генеральной совокупности.
2. Что такое выборка?
3. Что называют объёмом выборки?
4. В чём обычно состоит первичная обработка экспериментальных данных?
5. Как можно автоматизировать процесс обработки данных?
6. Что такое полигон частот? Где его применяют?
7. Какие характеристики относятся к показателям разброса выборки?
8. Как вычислить дисперсию выборки?
9. Как вычислить стандартное отклонение?

Лабораторная работа 4

МЕТОДЫ ИССЛЕДОВАНИЯ ЗАВИСИМОСТЕЙ МЕЖДУ ДВУМЯ ВЕЛИЧИНАМИ

Цель работы: изучить возможности программы MatLab применительно к такому инструменту управления качеством как диаграмма рассеивания.

Задание

1. Смоделировать пять выборок из нормального распределения с различными значениями дисперсии.
2. Построить диаграмму рассеивания.
3. Найти коэффициент корреляции для каждого из видов распределений.

Методические указания

На практике часто бывает важно изучить зависимости между парами каких-либо переменных. Одним из инструментов управления качеством продукции и процессов, позволяющим определить вид и тесноту связей между парами переменных, характеризующих качество продукции и процессов, является диаграмма рассеивания.

Для построения диаграммы рассеивания необходимо собрать данные (x, y), зависимость между которыми необходимо исследовать. Затем построить график зависимости $y = f(x)$. Нанести на график все необходимые обозначения (название, интервал времени сбора данных, число пар данных, названия и единицы измерения для каждой оси, дату составления диаграммы и т.д.).

Для изучения связей между переменными необходимо вычислить коэффициент корреляции по следующим формулам:

$$r = S(xy)\Big/\sqrt{S(xx)S(yy)}\ ; \quad S(xx) = \sum_{i=1}^{n}(x_i - \bar{x})^2\ ;$$

$$S(yy) = \sum_{i=1}^{n}(y_i - \bar{y})^2\ ; \quad S(xy) = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})\,,$$

где n – число пар данных; x_i, y_i – исходные данные; $\bar{x}, \bar{y}$ – средние арифметические значения; r – коэффициент корреляции, принимающий значения из диапазона $-1 \le r \le 1$.

Для вычисления среднего арифметического значения используется функция **mean()**. Для вычисления коэффициента корреляции используется функция **corrcoef()**.

Сгенерируйте исходные данные, воспользовавшись выражениями MatLab

$$\mathbf{yi = a + sqrt(b + ib/2) \cdot randn(c,1)};$$

$$\mathbf{x_i = [1:c]},$$

где **i=1…5** – номер выборки.

Исходные данные для моделирования выборок приведены в табл. 4.1.

4.1. Значения параметров *a, b, c* по вариантам

Вариант	Значения параметров			Вариант	Значения параметров		
	a	*b*	*c*		*a*	*b*	*c*
1	10	1	20	9	15	3	25
2	5	0,3	30	10	65	8	35
3	100	5	50	11	4	0,5	45
4	200	10	60	12	90	10	55
5	0,9	0,03	70	13	35	7	65
6	1,5	0,4	80	14	0,7	0,1	75
7	50	2	90	15	0,3	0,05	85
8	1000	45	100	16	76	8	95

Результаты выполнения лабораторной работы

№ выборки	Значение дисперсии	Значение коэффициента корреляции
1		
2		
3		
4		
5		

ШИФРОВАНИЕ ТЕКСТА КАК МЕТОД ЗАЩИТЫ ИНФОРМАЦИИ

Цель работы: изучить метод перестановки для шифрования открытого текста.

Задание

1. Изучить теоретические основы метода перестановки.
2. Зашифровать (расшифровать) одно слово открытого текста ключом, длина которого равна длине шифруемого слова.
3. Придумать символьный пароль, преобразовать его в ключ и зашифровать (расшифровать) фразу открытого текста с помощью этого ключа.

Методические указания

Шифрование является одним из эффективных способов защиты текстовой информации. При шифровании существуют следующие понятия.

Открытый текст – информация, содержание которой может быть понятно любому субъекту.

Шифрование – процесс преобразования открытого текста в шифротекст или криптограмму с целью сделать его содержание непонятным для посторонних лиц. В общем виде процесс шифрования описывается выражением вида $C = E_k(P)$, где C – шифротекст; E – функция шифрования; k – ключ шифрования; P – открытый текст.

Расшифрование – процесс обратного преобразования шифротекста в открытый текст. В общем виде процесс расшифрования описывается выражением вида $P = D_{k'}(C)$, где D – функция расшифрования; k' – ключ расшифрования.

Криптосистема – совокупность алгоритмов, реализуемых функциями E и D, множества ключей k, k' и шифротекстов.

Криптограмма (загадочное письмо или тайнопись) – наука о защите информации с помощью шифрования.

Криптоанализ – наука о методах дешифрования.

Криптостойкость – характеристика надёжности шифротекста от вскрытия.

Криптостойкость шифра характеризуют двумя величинами:

1) минимальным объёмом шифротекста, статическим анализом которого можно его вскрыть и получить открытый текст без знания ключа;

2) числом MIPS-часов (лет) – временем работы условного криптоаналитического компьютера производительностью 1 000 000 операций в секунду, необходимым для вскрытия шифротекста.

В настоящее время известно множество методов шифрования, одним из которых является метод перестановки.

В соответствии с этим методом биты (или символы) открытого текста переставляются в соответствии с задаваемым ключом шифрования правилом

$$1 \le i \le n,\ C_i = P_{k[i]}, \tag{1}$$

где $P = \{P_1, P_2, P_3, ...P_i, ...P_n\}$ – открытый текст; n – длина открытого текста (количество символов текста); $C = \{C_1, C_2, C_3, ...C_i, ...C_n\}$ – шифротекст; $k = \{k_1, k_2, k_3, ...k_i, ...k_n\}$ – ключ шифрования.

При расшифровании используется обратная перестановка:

$$P_{k[i]} = C_i. \tag{2}$$

Как видно из приведенных выражений, ключ должен удовлетворять условиям: $k_i \ne k_j$, $1 \le k_i \le n$.

Рассмотрим пример шифрования слова «Пример» методом перестановки. Зададим ключ, который должен быть равен 6-ти символам (количеству символов в шифруемом слове) в виде $k = \{1, 4, 6, 2, 3, 5\}$.

5.1. Данные для шифрования

Символы открытого текста	П	р	и	м	е	р
	P_1	P_2	P_3	P_4	P_5	P_6
Цифровые символы ключа	1	4	6	2	3	5
	k_1	k_2	k_3	k_4	k_5	k_6

Применим формулу (1) с выбранным ключом k к слову «Пример». Получим следующие выражения:

$$C_1 = P_{k[1]} = P_1 = \text{'П'};\ C_2 = P_{k[2]} = P_4 = \text{'м'};\ C_3 = P_{k[3]} = P_6 = \text{'р'};$$
$$C_4 = P_{k[4]} = P_2 = \text{'р'};\ C_5 = P_{k[5]} = P_3 = \text{'и'};\ C_6 = P_{k[6]} = P_5 = \text{'е'}.$$

В конечном итоге получим шифротекст $C = \text{Пмррие}$.

Очевидно, что применив другой ключ, получим другой вид шифрованного текста.

При дешифровании используем обратную операцию по формуле (2):

$$P_{k[1]} = P_1 = C_1 = \text{'П'};\ P_{k[2]} = P_4 = C_2 = \text{'м'};\ P_{k[3]} = P_6 = C_3 = \text{'р'};$$
$$P_{k[4]} = P_2 = C_4 = \text{'р'};\ P_{k[5]} = P_3 = C_5 = \text{'и'};\ P_{k[6]} = P_5 = C_6 = \text{'е'}.$$

Таким образом, получим $P = \{P_1, P_2, P_3, P_4, P_5, P_6\} = \{\text{Пример}\}$.

Если требуется зашифровать достаточно длинный текст длиной n, то его можно разбить на блоки, длина которых равна длине ключа m. Открытый текст записывают в таблицу с числом столбцов, равным длине ключа (каждый блок открытого текста записывается в столбец таблицы). Затем столбцы полученной таблицы переставляются в соответствии с ключом перестановки, а шифротекст считывается из строк таблицы последовательно.

Пусть требуется зашифровать открытый текст «этот пример шифрования». Длина текста (вместе с пробелами $n = 22$). Выберем ключ шифрования в виде $k = \{3,5,4,2,1\}$ $(m = 5)$.

Разбиваем строку «этот пример шифрования» на пять блоков, каждый из который располагаем в таблицу:

э	п	р	р	и
т	р		о	я
о	и	ш	в	
т	м	и	а	
	е	ф	н	

Переставляем столбцы полученной таблицы в соответствии с ключом $k = \{3,5,4,2,1\}$. Получим

р	и	р	п	э
	я	о	р	т
ш		в	и	о
и		а	м	т
ф		н	е	

Считываем последовательно текст из строк таблицы. Получим следующий шифр: рирпэ яортш виои амтф не.

Для расшифрования шифротекст записывают в таблицу того же размера по строкам, затем производится обратная перестановка столбцов в соответствии с ключом, после чего расшифрованный текст

считывается из таблицы по столбцам. Ниже приведены этапы расшифровывания: а) запись шифротекста в таблицу; б) перестановка столбцов в соответствии с ключом; в) считывание символов по столбцам.

Этап а

р	и	р	п	э
	я	о	Р	т
ш		в	и	о
и		а	м	т
ф		н	е	

Этап б

э	п	р	р	и
т	р		о	я
о	и	ш	в	
т	м	и	а	
	е	ф	н	

Результатом считывания данных таблицы этапа б будет фраза «этот пример шифрования».

Если в качестве ключа перестановки использовать последовательность не цифр, а произвольных символов (например пароль пользователя), то его необходимо предварительно преобразовать в последовательность целых чисел от 1 до *m*.

Например, пользователь ввел пароль «Петров».

Отсортируем символы в алфавитном порядке.

Получим Петров=>веопрт. Каждому символу присвоим порядковый номер:

в е о п р т
1 2 3 4 5 6

Заменим символы введённого пароля цифрами и получим ключ: 426531.

Выберите предложение открытого текста для шифрования в соответствии с номером своего варианта (табл. 5.2).

5.2. Таблица выбора заданий по вариантам

Вариант №	№ шифруемой строки	Вариант №	№ шифруемой строки	Вариант №	№ шифруемой строки
1	1	7	7	13	13
2	2	8	8	14	14
3	3	9	9	15	15
4	4	10	10	16	16
5	5	11	11	17	17
6	6	12	12	18	18

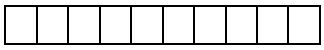
1. Существует три разновидности угроз.
2. Угроза нарушения конфиденциальности заключается в следующем.
3. Информация становится известной тому, кто не располагает полномочиями доступа к ней.
4. Она имеет место всякий раз, когда получен доступ к некоторой секретной информации.
5. Информация хранится в вычислительной системе или передается от одной системы к другой.
6. В связи с угрозой нарушения конфиденциальности, используется термин «утечка».
7. Угроза нарушения целостности включает в себя любое умышленное изменение информации.
8. Когда злоумышленники преднамеренно изменяют информацию, говорится, что целостность информации нарушена.
9. Целостность также будет нарушена, если к несанкционированному изменению приводит случайная ошибка программного или аппаратного обеспечения.
10. Санкционированными изменениями являются те, которые сделаны уполномоченными лицами с обоснованной целью.

11. Целостность информации это существование информации в неискаженном виде.
12. Чаще субъектов интересует обеспечение более широкого свойства – достоверности информации.
13. Угроза отказа служб возникает всякий раз, когда в результате преднамеренных действий, предпринимаемых другим пользователем или злоумышленником, блокируется доступ к некоторому ресурсу вычислительной системы.
14. Реально блокирование может быть постоянным – запрашиваемый ресурс никогда не будет получен.
15. Или оно может вызывать только задержку запрашиваемого ресурса, достаточно долгую для того чтобы он стал бесполезным.
16. В этих случаях говорят, что ресурс исчерпан.
17. Естественные угрозы – угрозы, вызванные воздействиями на АС и её компоненты объективных физических процессов или стихийных природных явлений, независящих от человека.
18. Искусственные угрозы это угрозы информационной безопасности АС, вызванные деятельностью человека.

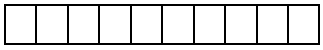
Результаты выполнения лабораторной работы

Исходное слово для шифрования:

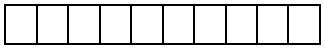
Ключ шифрования:



Результат шифрования

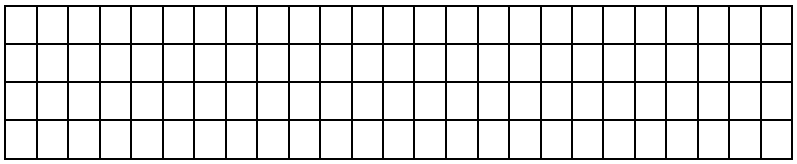


Символьный пароль

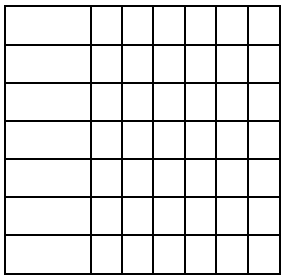


Цифровой пароль _ _ _ _ _ _ _ _ _ _

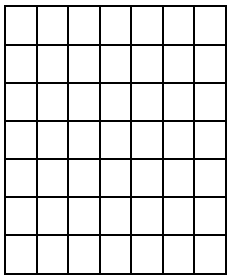
Фраза для шифрования



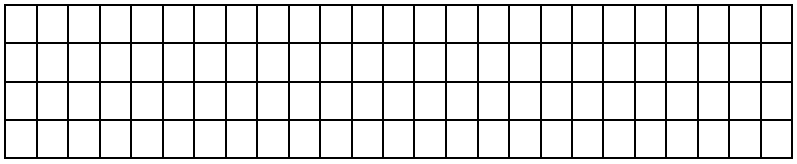
Этап а



Этап б



Результат шифрования



Контрольные вопросы

1. Охарактеризовать информацию и её свойства.
2. Что является предметом и объектом защиты информации?
3. Чем определяется ценность информации? Приведите классификацию конфиденциальной информации.
4. Охарактеризовать свойства достоверности и своевременности информации.
5. Что понимается под угрозой информации? Назвать разновидности угроз информации.
6. Привести классификацию угроз информации.
7. Какие основные направления и методы реализации угроз вам известны?
8. Пояснить классификацию злоумышленников.
9. Охарактеризовать причины и виды утечки информации.
10. Назвать и привести примеры каналов утечки информации.
11. Перечислить задачи государства в области безопасности информации.
12. Охарактеризовать основные законы РФ, регулирующие отношения в области информационных технологий.
13. Назвать государственные органы, обеспечивающие безопасность информационных технологий, и решаемые ими задачи.
14. Пояснить, что такое шифрование и в чём заключается сущность метода перестановки.

Лабораторная работа 6

АВТОМАТИЗАЦИЯ ШИФРОВАНИЯ МЕТОДОМ ПЕРЕСТАНОВКИ

Цель работы: разработать программу, реализующую шифрование слова методом перестановки.

Задание

1. Написать программу, позволяющую автоматически зашифровать (расшифровать) заданное в текстовом файле слово с помощью ключа, введенного с клавиатуры.
2. Протестировать программу, зашифровав (расшифровав) 3-5 слов различными ключами.

Методические указания

Для написания программы можно использовать любой из известных языков программирования. В данной работе используется встроенный язык программирования программы MatLab.

Исходное слово для шифрования помещается в текстовый файл, созданный, например, в редакторе «Блокнот».

Откроем файл из программы MatLab. Это осуществляется за счёт использования выражения **fid=fopen('nietzsche.txt','r')**. В приведенном выражении в оператор **fopen** передаются два значения: **nietzsche.txt** – имя открываемого файла, содержащего шифруемое слово; **r** – статус открытия файла, означающий, что файл будет открыт только для чтения. В переменную **fid** будет записан результат открытия файла. Если в переменную **fid** запишется число равное –1, то это означает, что файл не был открыт. Поэтому в программу можно добавить следующий код:

```
if(fid==–1)
disp('ошибка открытия файла')
error(mes)
```

который предназначен для вывода сообщения об ошибке, возникающей в случае, если файл не был открыт.

В случае успешного открытия файла, содержащуюся в нём строку (строки) необходимо записать в некоторую переменную (переменные). Оператор MatLab **feof(fid)** позволяет определить факт достижения конца файла. Оператор **fgetl(fid)** позволяет последовательно считывать строки файла и записывать их в заданные переменные. Оператор **size(string,2)** позволяет определить длину (количество символов) строки. Ниже приведён фрагмент программы, позволяющий считать строку (или строки) из файла и определить её размер:

```
while ~(feof(fid)) % пока не конец файла
```

```
string=fgetl(fid); % запись строки из файла в переменную string
char_tek=size(string,2); % запись длины строки string
end
```

После того как длина слова определена, пользователя программы можно попросить ввести с клавиатуры ключ шифрования и сохранить его в массив **kl** так, как показано в примере:

```
for i=1:char_tek
kl(i)=input('Введите число');
end
```

Теперь можно зашифровать слово методом перестановки. Это осуществляется в следующем коде программы:

```
for i=1:char_tek
shifr(i)=string(kl(i));
end
```

Шифр помещается в строку **shifr**, которую можно вывести на экран с помощью оператора **disp(shifr)**. После этого открытый ранее файл можно закрыть, применив оператор **fclose(fid)**.

Ниже приведён полный текст программы шифрования слова:

```
fid=fopen('nietzsche.txt','r');
if(fid==-1)
disp('ошибка открытия файла')
error(mes)
else
while ~(feof(fid))
string=fgetl(fid);
char_tek=size(string,2);
end
clc
disp('Шифруемое слово:')
disp(string)
disp('Длина ключа')
disp(char_tek)
for i=1:char_tek
kl(i)=input('Введите число');
end
for i=1:char_tek
shifr(i)=string(kl(i));
end
disp('Шифрованное слово')
disp(shifr)
fclose(fid);
end
```

6.1. Варианты заданий для самостоятельной работы

№ варианта	Слово	№ варианта	Слово	№ варианта	Слово
1	technical	7	simulation	13	matrix
2	Algorithm	8	standard	14	Scientific
3	matlab	9	analysis	15	complete
4	Modeling	10	logic	16	graphics

5	project	11	control	17	including
6	Interface	12	interactive	18	system

Результаты выполнения лабораторной работы

Исходное слово для шифрования

Результат шифрования Ключ шифрования

Результат шифрования Ключ шифрования

Результат шифрования Ключ шифрования

Результат шифрования Ключ шифрования

Результат шифрования Ключ шифрования

Контрольные вопросы

1. Пояснить алгоритм работы вашей программы.
2. Указать недостатки программы шифрования, приведённой в примере. Как можно их устранить.
3. Написать код для подсчёта числа символов ‘a’ в строке.
4. Написать код для подсчёта числа пустых строк в файле.
5. Написать код для подсчёта числа не пустых строк в файле.
6. Написать код для подсчёта общего числа строк в файле.
7. Написать код для преобразования пароля в ключ шифрования.
8. Написать код для подсчёта числа символов в файле.
9. Написать код для замены символов ‘p’ слова цифрой 1.
10. Написать код для сортировки букв слова в алфавитном порядке.
11. Написать код для подсчёта числа букв в первом слове предло-жения.
12. Написать код для подсчёта числа пробелов в предложении.
13. Написать код для замены пробелов в предложении цифрой 0.
14. Написать код для подсчёта числа слов в предложении. Слова разделены пробелами.

МЕТОДЫ ЗАЩИТЫ КОМПЬЮТЕРА ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА ИЗ ВНЕШНЕЙ СЕТИ И ПОИСК УЯЗВИМОСТЕЙ В СИСТЕМЕ ЗАЩИТЫ

Цель работы: 1. Изучить возможности брандмауэра Windows и обозревателя Internet Explorer в защите от несанкционированного доступа. 2. Изучить методы анализа защищенности информационных ресурсов.

Задание

1. Определить все доступные узлы в локальной сети.
2. Просканировать порты сервера компьютерной сети.

Методы защиты компьютера от несанкционированного доступа из внешней сети

Брандмауэр – сочетание программного и аппаратного обеспечения, образующее систему защиты от несанкционированного доступа из внешней глобальной сети во внутреннюю сеть (интрасеть). Брандмауэр предотвращает прямую связь между внутренней сетью и внешними компьютерами, пропуская сетевой трафик через прокси-сервер, находящийся снаружи сети. Прокси-сервер определяет, следует ли разрешить файлу попасть во внутреннюю сеть. Брандмауэр называется также шлюзом безопасности.

Можно считать брандмауэр пограничным постом, на котором проверяется информация (часто называемая *трафик*), приходящая из Интернета или локальной сети. В ходе этой проверки брандмауэр отклоняет или пропускает информацию на компьютер в соответствии с установленными параметрами.

Когда к компьютеру пытается подключиться кто-то из Интернета или локальной сети, такие попытки называют «непредусмотренными запросами». Когда на компьютер поступает непредусмотренный запрос, брандмауэр Windows блокирует подключение. Если на компьютере используются такие программы, как программа передачи мгновенных сообщений или сетевые игры, которым требуется принимать информацию из Интернета или локальной сети, брандмауэр запрашивает пользователя о блокировании или разрешении подключения. Если пользователь разрешает подключение, брандмауэр Windows создает *исключение*, чтобы в будущем не тревожить пользователя запросами по поводу поступления информации для этой программы.

Если идёт обмен мгновенными сообщениями с собеседником, который собирается прислать файл (например фотографию), брандмауэр Windows запросит подтверждение о снятии блокировки подключения и разрешении передачи фотографии на компьютер. А при желании участвовать в сетевой игре через Интернет с друзьями пользователь может добавить эту игру как исключение, чтобы брандмауэр пропускал игровую информацию на компьютер.

Хотя имеется возможность отключать брандмауэр Windows для отдельных подключений к Интернету или локальной сети, это повышает вероятность нарушения безопасности компьютера.

Чтобы открыть компонент «Брандмауэр Windows», нажмите кнопку **Пуск**, выберите пункты **Настройка**, **Панель управления**, **Сеть и подключения к Интернету** и **Брандмауэр Windows**.

В обозревателе Internet Explorer имеется несколько возможностей, позволяющих обеспечить защиту конфиденциальности, а также повысить безопасность личных данных пользователя.

Параметры конфиденциальности позволяют защитить личные данные пользователя — с помощью этих параметров можно понять, как просматриваемые веб-узлы используют эти данные, а также задать значения параметров конфиденциальности, которые будут определять, разрешено ли веб-узлам сохранять файлы «cookie» на компьютере.

В число параметров конфиденциальности Internet Explorer входят следующие.

- Параметры конфиденциальности, определяющие обработку на компьютере файлов «cookie». Файлы «cookie» — это созданные веб-узлом объекты, которые сохраняют на компьютере определенные сведения, например о предпочтениях пользователя при посещении данного узла. Кроме того, эти файлы могут также сохранять личные данные пользователя, такие как имя и адрес электронной почты.
- Оповещение безопасности, выдаваемые пользователю при попытке получить доступ к веб-узлу, не соответствующему заданным параметрам конфиденциальности.
- Возможность просмотра политики конфиденциальности P3P для веб-узла.

Средства безопасности позволяют предотвратить доступ других пользователей к таким сведениям, на доступ к которым у них нет разрешения. Это, например, сведения о кредитной карточке, вводимые при покупках в Интернете. Эти средства безопасности могут также защитить компьютер от небезопасного программного обеспечения.

В число параметров безопасности Internet Explorer входят следующие.

- Возможность блокирования большинства всплывающих окон.
- Возможность обновления, отключения или повторного включения надстроек для веб-обозревателя.
- Средства повышения безопасности, предупреждающие пользователя о попытке веб-узла загрузить файлы или программы на компьютер.
- Цифровые подписи, которые подтверждают, что файл поступил действительно от указанного лица или издателя и с момента включения цифровой подписи в этот файл никем не внесены изменения.
- Безопасное подключение с использованием 128-разрядного ключа, которое применяется для связи с безопасными веб-узлами.

Поиск уязвимостей в системе защиты

Противостояние атакам – важное свойство защиты. Казалось бы, если в сети установлен межсетевой экран (firewall), то безопасность гарантирована, но это распространенное заблуждение может привести к серьёзным последствиям.

Например, межсетевой экран (МЭ) не способен защитить от пользователей, прошедших аутентификацию. А квалифицированному хакеру не составляет труда украсть идентификатор и пароль авторизованного пользователя. Кроме того, межсетевой экран не только не защищает от проникновения в сеть через модем или иные удалённые точки доступа, но и не может обнаружить такого злоумышленника.

При этом система защиты, созданная на основе модели адаптивного управления безопасностью сети (Adaptive Network Security, ANS), способна решить все или почти все перечисленные проблемы. Она позволяет обнаруживать атаки и реагировать на них в режиме реального времени, используя правильно спроектированные, хорошо управляемые процессы и средства защиты.

Компания Yankee Group опубликовала в июне 1998 г. отчёт, содержащий описание процесса обеспечения адаптивной безопасности сети. Этот процесс должен включать в себя анализ защищённости, т.е. поиск уязвимостей, обнаружение атак, а также использовать адаптивный (настраиваемый) компонент, расширяющий возможности двух первых функций, и управляющий компонент.

Анализ защищённости осуществляется на основе поиска уязвимых мест во всей сети, состоящей из соединений, узлов (например, коммуникационного оборудования), хостов, рабочих станций, приложений и баз данных. Эти элементы нуждаются как в оценке эффективности их защиты, так и в поиске в них неизвестных уязвимостей. Процесс анализа защищённости предполагает исследование сети для выявления в ней слабых мест и обобщение полученных сведений, в том числе в виде отчёта. Если система, реализующая данную технологию, содержит адаптивный компонент, то устранение найденной уязвимости будет осуществляться автоматически. При анализе защищённости обычно идентифицируются:

- люки в системах (back door) и программы типа «троянскийконь»;
- слабые пароли;
- восприимчивость к проникновению из внешних систем и к атакам типа «отказ в обслуживании»;
- отсутствие необходимых обновлений (parch, hotfix) операционных систем;
- неправильная настройка межсетевых экранов, WEB-серверов и баз данных.

Обнаружение атак – это процесс оценки подозрительных действий в корпоративной сети, реализуемый посредством анализа журналов регистрации операционной системы и приложения (log-файлов) либо сетевого трафика. Компоненты ПО обнаружения атак размещаются на узлах или в сегментах сети и оценивают различные операции, в том числе с учётом известных уязвимостей.

Адаптивный компонент ANS позволяет модифицировать процесс анализа защищенности, предоставляя самую последнюю информацию о новых уязвимостях. Он также модифицирует компонент обнаружения атак, дополняя его последней информацией о подозрительных действиях и атаках. Примером адаптивного компонента может служить механизм обновления баз данных антивирусных программ, которые являются частным случаем систем обнаружения атак.

Управляющий компонент предназначен для анализа тенденций, связанных с формированием системы защиты организации и генерацией отчётов.

К сожалению, эффективно реализовать все описанные технологии в одной системе пока не удаётся, поэтому пользователям приходится применять совокупность систем защиты, объединённых единой концепцией безопасности. Пример таких систем – семейство продуктов SAFEsuite, разработанных американской компанией Internet Security Systems (ISS). В настоящее время комплект ПО SAFEsuite поставляется в новой версии SAFEsuite Enterprise, в которую входит также ПО SAFEsuite Decisions, обеспечивающее принятие решений по проблемам безопасности.

Система анализа защищённости Internet Scanner предназначена для проведения регулярных всесторонних или выборочных тестов сетевых служб, операционных систем, используемого прикладного ПО, маршрутизаторов, межсетевых экранов, WEB-серверов и т.п.

Другим примером системы анализа защищенности является программа SuperScan, позволяющая сканировать открытые порты узлов с известными IP-адресами.

Для начала сканирования достаточно в поле Start указать IP-адрес сканируемого узла. Для более глубокого сканирования необходимо указать минимальную скорость сканирования, передвинув движок на отметку Min.

Результаты выполнения лабораторной работы

Результаты поиска уязвимостей в системе защиты
узла _____ IP: ___ . ___ . ___ . ___
Символьное имя узла IP адрес узла

№ п.п	№ порта	Наименование	№ п.п	№ порта	Наименование
1			7		
2			8		
3			9		
4			10		
5			11		
6			12		

Контрольные вопросы

1. Может ли брандмауэр блокировать компьютерным вирусам и «червям» доступ на компьютер?
2. Может ли брандмауэр обнаружить или обезвредить компьютерных вирусов и «червей», если они уже попали на компьютер?
3. Может ли брандмауэр запретить пользователю открывать сообщения электронной почты с опасными вложениями?
4. Может ли брандмауэр блокировать спам или несанкционированные почтовые рассылки?
5. Может ли брандмауэр запросить пользователя о выборе блокировки или разрешения для определённых запросов на подключение?
6. Для чего нужен журнал безопасности брандмауэра?
7. Сколько параметров определяет работу брандмауэра?
8. Перечислите параметры, определяющие работу брандмауэра.

9. Какой параметр брандмауэра обеспечивает наивысшую защиту компьютера?
10. Что такое брандмауэр, его назначение.
11. Как работает брандмауэр?
12. Какими возможностями обладает Интернет обозреватель для защиты личных данных пользователей?
13. Перечислите параметры безопасности Internet Explorer.
14. Перечислите параметры конфиденциальности Internet Explorer.
15. Для чего нужно проводить анализ защищенности сети?
16. Перечислите наиболее уязвимые «места» сети.
17. Приведите примеры систем анализа уязвимостей.

Лабораторная работа 8

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

Цель работы: изучить современные достижения в области защиты компьютерной информации, получить навыки формирования политики и программы защиты информационных ресурсов.

Задание

1. Составить список фирм, специализирующихся на защите компьютерной информации.
2. Составить политику и сформировать программу защиты информационных ресурсов.

Методические указания

В поисковых системах сети Интернет (например yandex.ru, aport.ru) найдите 3-4 фирмы, которые предлагают свои услуги по защите компьютерной информации. Просмотрите перечень услуг этих фирм и определите: объекты защиты, методы и средства защиты. Полученные данные внесите в табл. 8.1.

Выберите любую организацию (например, можно взять организацию – место прохождения производственной практики) и сформируйте: основные положения политики защиты информационных ресурсов; укажите компоненты программы защиты информационных ресурсов.

8.1. Перечень фирм, специализирующихся на защите информационных ресурсов

Наименование фирмы	Предложения по защите	
	Объекты защиты	Методы и средства

Контрольные вопросы

1. Что представляет собой политика защиты информации?
2. В чём заключается политика защиты информации верхнего уровня?
3. Какие темы освещает политика защиты информации среднего уровня?
4. Расскажите об аспектах, которые включает политика защиты информации нижнего уровня.
5. На что направлена программа защиты информации?
6. Что должна регламентировать программа защиты информации?
7. Какие компоненты входят в программу защиты информации?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кетков, Ю.Л. MatLab 7: программирование, численные методы / Ю.Л. Кетков, А.Ю. Кетков. – СПб. : БХВ-Петербург, 2005. – 752 с.

2. Плис, А.И. MathCAD: математический практикум для инженеров и экономистов : учеб. пособие для вузов /А.И. Плис, И.А. Славина. – 2-е изд., перераб. и доп. – М. : Финансы и статистика, 2003. – 636 с.

3. Гринберг, А.С. Защита информационных ресурсов государственного управления : учеб. пособие для вузов / А.С. Гринберг, Н.Н. Горбачёв, А.А. Тепляков. – М. : Юнити-ДАНА, 2003. – 327 с.

4. Пономарёв, С.В. Управление качеством продукции. Инструменты и методы менеджмента качества : учеб. пособие / С.В. Пономарёв, С.В. Мищенко, В.Я. Белобрагин [и др.]. – М. : РИА «Стандарты и качество», 2005. – 248 с.