Задание по направлению «Прикладная математика» с решением

Профиль

«Системы управления и обработки информации в инженерии»

Время выполнения задания – 240 мин.

Задача 1.(20 баллов)
(дифференциальные уравнения)

Найти решение системы дифференциальных уравнений:

$$\begin{cases} \dfrac{dx(t)}{dt} = 6x(t) - 4y(t) \\ \dfrac{dy(t)}{dt} = 3x(t) - y(t) \end{cases}$$

при $x(t_0) = 1,\ y(t_0) = -0.5$, на интервале $t \in [t_0, t_f],\ t_0 = 0,\ t_f = 5$.

Определить устойчиво ли решение и, если решение устойчиво, определить значения переменных x и y при $t \to \infty$.

Решение.

Для установления устойчивости решения системы определим корни характеристического уравнения:

$$\begin{vmatrix} 6-\lambda & -4 \\ 3 & -1-\lambda \end{vmatrix} = 0.$$

Вычисляя определитель, получим:

$\lambda^2 - 5\lambda + 6 = 0$.

Корни этого уравнения $\lambda_1 = 3; \lambda_2 = 2.$

Оба корня положительны, следовательно, система не имеет установившегося решения при $t \to \infty$.

Найдем общее решение однородной системы

$$\begin{cases} \dfrac{dx(t)}{dt} = 6x(t) - 4y(t) \\ \dfrac{dy(t)}{dt} = 3x(t) - y(t) \end{cases}.$$

В соответствии с полученными корнями характеристического уравнения будем иметь два возможных решения:

$x_1 = \alpha_1^{(1)} \cdot e^{3\cdot t}; y_1 = \alpha_2^{(1)} \cdot e^{3\cdot t};$

$x_2 = \alpha_1^{(2)} \cdot e^{2\cdot t}; y_2 = \alpha_2^{(2)} \cdot e^{2\cdot t},$

где в $\alpha_i^{(j)}$, i – номер коэффициента, j – номер решения.

Подставим первое из решений в первое уравнение однородной системы

$$\frac{dx_1}{dt} = 3\alpha_1^{(1)} \cdot e^{3\cdot t} = 6\alpha_1^{(1)} \cdot e^{3\cdot t} - 4\alpha_2^{(1)} \cdot e^{3\cdot t},$$

тогда

$$3\alpha_1^{(1)} e^{3t} = 4\alpha_2^{(1)} e^{3t}; \Rightarrow \alpha_2^{(1)} = \frac{3}{4}\alpha_1^{(1)}.$$

Положив $\alpha_1^{(1)} = C_1$, получим $\alpha_2^{(1)} = 0,75 \cdot C_1$. И, далее

$x_1 = C_1 \cdot e^{3t}; y_1 = 0,75 \cdot C_1 \cdot e^{3t}.$

Подставим теперь второе решение во второе уравнение однородной системы:

$$\frac{dy_2}{dt} = 2 \cdot \alpha_2^{(2)} \cdot e^{2t} = 3 \cdot \alpha_1^{(2)} \cdot e^{2t} - 1 \cdot \alpha_2^{(2)} \cdot e^{2t}.$$

$\alpha_2^{(2)} \cdot e^{2t} = \alpha_1^{(2)} \cdot e^{2t}; \Rightarrow \alpha_2^{(2)} = \alpha_1^{(2)}.$
Положив $\alpha_1^{(2)} = C_2$, получим $\alpha_2^{(2)} = C_2$ и далее

$x_2 = C_2 \cdot e^{2t}; y_2 = C_2 \cdot e^{2t}.$
Обобщая найденные решения можно записать:

$$\begin{cases} x(t) = C_1 \cdot e^{3t} + C_2 \cdot e^{2t} \\ y(t) = 0,75 \cdot C_1 \cdot e^{3t} + C_2 \cdot e^{2t} \end{cases}.$$

Определим частное решение при начальных условиях $x(0) = 1; y(0) = -0,5.$

$$\begin{cases} C_1 + C_2 = 1 \\ 0,75 \cdot C_1 + C_2 = -0,5 \end{cases}$$

$C_2 = -5; C_1 = 6;$

Таким образом, имеем решение как функцию независимой переменной *t*:

$$\begin{cases} x(t) = 6 \cdot e^{3t} - 5 \cdot e^{2t} \\ y(t) = 0,75 \cdot 6 \cdot e^{3t} - 5 \cdot e^{2t} \end{cases}$$

Задача 2.(20 баллов)
(теория управления)

Объект управления описывается дифференциальным уравнением
$y''(t) + 3y'(t) + 2y(t) = x'(t) + 3x(t).$

Найти математическую модель переходного процесса представленного объекта.

Рещение.

Исходному дифференциальному уравнению соответствует передаточная функция $W(s)=\dfrac{Y(s)}{X(s)}=\dfrac{s+3}{s^2+3s+2}=\dfrac{s+3}{(s+1)(s+2)}$.

Согласно определению, переходной процесс есть реакция динамической системы на единичное ступенчатое воздействие, изображение которого по Лапласу равно $1/s$. Исходя из сказанного, запишем:

$$Y(s)=\frac{s+3}{(s+1)(s+2)}\frac{1}{s}=\frac{1}{2(s+2)}-\frac{2}{s+1}+\frac{3}{2s}.$$

Используя таблицу преобразований Лапласа, переведем полученное решение из пространства изображений в пространство оригиналов:

$y(t)=0.5e^{-2t}-2e^{-t}+1.5,\ \ t>0$.

Задача 3.(10 баллов)
(линейная алгебра)

Линейный оператор φ в двумерном пространстве $\Re^2$ задан матрицей A=$\begin{pmatrix}-2 & 3\\ 2 & 3\end{pmatrix}$ в стандартном базисе $\vec{l}_1=\begin{pmatrix}1\\0\end{pmatrix}, \vec{l}_2=\begin{pmatrix}0\\1\end{pmatrix}$ пространства $\Re^2$.

Найти собственные числа $\lambda_1,\ \lambda_2$ оператора φ и для каждого из собственных чисел λ_1,λ_2 найти по одному собственному вектору $\vec{f}_1,\vec{f}_2$, соответственно. Доказать, что $\vec{f}_1,\vec{f}_2$ образуют базис в $\Re^2$, и найти матрицу данного оператора в базисе $\vec{f}_1,\vec{f}_2$, а также найти координаты вектора $\vec{a}=\begin{pmatrix}1\\1\end{pmatrix}$ в базисе $\vec{f}_1,\vec{f}_2$.

Решение.

Собственный вектор $\vec{x}\neq\vec{0}$ с собственным числом λ удовлетворяет равенству $\varphi(\vec{x})=\lambda\vec{x}$.

Собственные числа являются корнями характеристического уравнения:

$$\begin{vmatrix}(-2-\lambda) & 3\\ 2 & (3-\lambda)\end{vmatrix}=0,\quad \lambda^2-\lambda-12=0,\qquad \lambda_1=-3,\quad \lambda_2=4.$$

Найдем собственные векторы для каждого из полученных собственных чисел.

1). $\lambda_1=-3$. Тогда равенство $\varphi(\vec{x})=-3\vec{x},$ где $\vec{x}=\begin{pmatrix}x_1\\x_2\end{pmatrix}$ можно переписать в виде системы:

$$\begin{pmatrix}-2 & 3\\ 2 & 3\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix}=-3\begin{pmatrix}x_1\\x_2\end{pmatrix},\qquad \begin{cases}x_1+3x_2=0\\ 2x_1+6x_2=0\end{cases},\quad \vec{x}=\begin{pmatrix}-3t\\t\end{pmatrix},\quad \forall t\in\Re,\quad t\neq 0.$$

Положив $t=1$, получим $\vec{f}_1=\begin{pmatrix}-3\\1\end{pmatrix}$ - один из собственных векторов с собственным числом $\lambda_1=-3$.

2). $\lambda_2=4.$ Тогда равенство $\varphi(\vec{x})=4\vec{x}$, где $\vec{x}=\begin{pmatrix}x_1\\x_2\end{pmatrix}$ можно переписать в виде системы:

$$\begin{pmatrix}-2 & 3\\ 2 & 3\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix}=4\begin{pmatrix}x_1\\x_2\end{pmatrix},\quad \begin{cases}-6x_1+3x_2=0\\ 2x_1-x_2=0\end{cases},\qquad \vec{x}=\begin{pmatrix}t\\2t\end{pmatrix},\quad \forall t\in\Re\ ,\quad t\neq 0.$$

Положив $t=1$, получим $\vec{f}_2=\begin{pmatrix}1\\2\end{pmatrix}$ - один из собственных векторов с собственным числом $\lambda_2=4.$

Чтобы доказать, что $\vec{f}_1,\ \vec{f}_2$ линейно независимы, рассмотрим равенство $\alpha_1\vec{f}_1+\alpha_2\vec{f}_2=\vec{0}.$ Это равенство сведется к системе уравнений:

$$\alpha_1\begin{pmatrix}-3\\1\end{pmatrix}+\alpha_2\begin{pmatrix}1\\2\end{pmatrix}=\begin{pmatrix}0\\0\end{pmatrix},\quad \begin{cases}-3\alpha_1+\alpha_2=0\\ \alpha_1+2\alpha_2=0\end{cases}\Leftrightarrow\quad \begin{cases}\alpha_1=0\\ \alpha_2=0\end{cases}.$$

Это и будет означать, что $\vec{f}_1,\ \vec{f}_2$ - линейно независимы, следовательно $\vec{f}_1,\ \vec{f}_2$ будут образовывать базис в двумерном пространстве $\Re^2$. Заметим, что верно общее утверждение: любой набор собственных векторов с различными собственными числами, будет линейно независимым.

Чтобы найти числа α_1,α_2 – координат вектора $\vec{a}=\begin{pmatrix}1\\1\end{pmatrix}$ в базисе $\vec{f}_1,\ \vec{f}_2$, запишем равенство:

$$\alpha_1\vec{f}_1+\alpha_2\vec{f}_2=\vec{a},\quad \alpha_1\begin{pmatrix}-3\\1\end{pmatrix}+\alpha_2\begin{pmatrix}1\\2\end{pmatrix}=\begin{pmatrix}1\\1\end{pmatrix};\quad \begin{cases}-3\alpha_1+\alpha_2=1\\ \alpha_1+2\alpha_2=1\end{cases},\quad \begin{cases}\alpha_1=-\dfrac{1}{7}\\ \alpha_2=\dfrac{4}{7}\end{cases}$$

Задача 4.(20 балов)
(защита информации)

Одним из самых простых способов шифрования открытого текста является шифр простой замены. Он состоит в том, что каждая буква в алфавите, которым написано открытое сообщение, заменяется на какой-то другой символ, например, другую букву того же алфавита. При этом замена на букву не является обязательной – например, в одном из произведений А. Конан-Дойля, посвященном сыщику Шерлоку Холмсу, главному герою приходится иметь дело с шифром простой замены, в котором буквы сообщения заменены изображениями пляшущих человечков в разных позах.
Пусть дана таблица замены для алфавита русского языка:

Сообщение	Шифротекст	Сообщение	Шифротекст	Сообщение	Шифротекст
А	Г	К	Т	Х	З

Б	Ш	Л	Х	Ц	Ж
В	Ы	М	Я	Ч	Л
Г	О	Н	Ь	Ш	Ё
Д	Э	О	Ф	Щ	Н
Е	Ц	П	У	Ъ	Д
Ё	М	Р	К	Ы	Е
Ж	Ъ	С	Ю	Ь	Б
З	Щ	Т	Р	Э	Ч
И	А	У	П	Ю	И
Й	В	Ф	С	Я	Й

1. Можно ли по шифротексту, полученному по данной таблице, отличить зашифрованное слово «криптография» от слова «криптограмма»?
2. Изменит ли такое положение повторное применение замены, указанной в таблице, к шифротексту.
3. Пусть задано слово открытого текста «криптография», которое шифруется по данной таблице. Затем получившийся шифротекст шифруется снова по той же таблице, затем данный процесс повторяется неограниченное число раз. Можно ли таким образом получить из слова «криптография» все возможные наборы из 10 букв?
4. Сколько различных шифротекстов можно получить из слова «криптография» указанным выше способом?

Решение.

1. Да, можно. Шифр простой замены преобразует одинаковые символы в одинаковые, а различные – в различные. В различающихся частях слов «криптография» и «криптограмма» имеются отличия в структуре размещения одинаковых букв. В слове «криптограмма» одинаковы 9 и 12, а также 10 и 11 буквы. Следовательно, зашифрованное слово «криптограмма» можно отличить по наличию одинаковых символов в указанных позициях.
   Комментарий: фраза «по шифртексту, полученному по данной таблице» означает, что отличить одно слово от другого требуется, зная, что оба они получены по одной и той же таблице простой замены, и имея в распоряжении лишь сами зашифрованные тексты. Ответ, указывающий на то, что данные слова по данной таблице дают различные зашифрованные тексты, требует наличия в процессе

исследования зашифрованных текстов также и самой таблицы, что противоречит приведенному фрагменту из формулировки вопроса.

2. Нет, не изменит. Многократное – даже различное по количеству для каждого из слов – применение простой замены не изменит указанной закономерности: одинаковые буквы будут заменены одинаковыми, а разные – разными. Следовательно, в слове «криптограмма» всегда будут сохраняться пары одинаковых букв на 9 и 12, а также на 10 и 11 позициях.
3. Нет, нельзя. Формулировка «все возможные наборы из 10 букв» означает все возможные варианты замены 10 различных букв, входящих в состав слова «криптография» на произвольные буквы шифрованного текста. Другими словами, это означает, можно ли, выписав все 10 различных букв слова «криптография» и произвольным образом заменив каждую из них какой-либо буквой алфавита, заботясь о том, чтобы различные буквы были заменены различными буквами, ожидать, что через некоторое количество последовательных применений приведенной таблицы будет реализована именно та замена, которая была задана. Для демонстрации того, что это невозможно, достаточно показать, что какая-либо буква слова «криптография» может быть заменена лишь на буквы некоторого набора. Например, буква «к» заменяется на букву «т». «т», в свою очередь, заменяется на «р», а «р» - снова на «к». Это означает, что при любом количестве последовательных применений данной таблицы буква «к» будет заменяться на одну из букв набора «т», «р» и «к». Никаким количеством последовательных зашифрований нельзя добиться ее замены на, например, букву «а».
4. Для подсчета различных шифротекстов, которые могут быть получены из слова «криптография» последовательным многократным применением процедуры зашифрования требуется для каждой буквы заданного слова посчитать, через сколько применений она будет заменена на саму себя. Далее следует вычислить наименьшее общее кратное этих значений, которое и будет ответом на вопрос.
   В нашем случае получаются следующие циклические группы:
   1) к– т – р – к – 3 применения до возврата к исходному значению;

   р – входит в ту же группу – 3 применения таблицы до возврата к исходному значению;
   2) и – а – г – о – ф – с – ю – и – 7 применений до возврата к исходному значению;
   3) п – у – п – 2 применения до возврата к исходному значению;

   т – входит в первую группу, 3 применения;

   о – входит во вторую группу, 7 применений;

   г – входит во вторую группу, 7 применений;

   р - входит в первую группу, 3 применения;

   а - входит во вторую группу, 7 применений;

   ф - входит во вторую группу, 7 применений;

   и - входит во вторую группу, 7 применений;
   4) я – й – в – ы – е – ц – ж – ъ – д – э – ч – л – х – з – щ – н – ь – б – ш – ё – м – я – 21 применение до возврата к исходному значению.

   Наименьшее общее кратное чисел 3, 7, 2 и 21 равно 42. Это и есть ответ на вопрос.

Для иллюстрации решения можно представить себе 12 нанизанных на одну ось дисков, на ободе каждого из которых записаны буквы, входящие в установленные нами циклические группы. Первый диск разделен всего на 3 деления – для букв «к», «т» и «р», а последний – на 21. Исходно они установлены так, что вдоль оси на них можно прочесть слово «криптография». Каждое применение операции зашифрования по данной таблице соответствует повороту всех дисков на 1 деление, независимо от того, сколько их всего. Очевидно, что первый диск сделает полный оборот уже за 3 применения, а последний – лишь за 21 применение. Когда все диски сделают целое число полных оборотов, будет снова получено слово «криптография». После этого процесс повторения комбинаций букв на них начнется заново, так что новых комбинаций мы уже не встретим. Ситуация же, когда все диски сделают целое число полных оборотов, впервые случится именно тогда, когда число применений операции шифрования будет равно наименьшему общему кратному количеств букв на всех дисках.

Задача 5.(30 баллов)
(операционные системы, системное программирование)

В нижеприведенном фрагменте программы на языке программирования С в UNIX – подобной операционной системе одна из инструкций (команд) задана некорректно. Укажите эту инструкцию (команду) и объясните, почему она некорректна?

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
int main()
{ int a, c, d,  p[2];
char buf[5000], name[256];
for(;;)
{ printf("введите имя файла для записи результата:");
scanf("%s",name);
a=creat(name, 0644);
pipe(p);
if( fork()==0)
{close(p[0]);
close(1);
dup(p[1]);
system("pwd");
system("ls –l");
exit(0);
}
else
{wait(&c);
close(p[1]);
```

```
d=read(p[0], buf, 50000);
buf[d+1]=0;
close(p[0]);
close(1);
dup2(a,1);
close(a);
printf("%s\n",buf);
close(1);
}
}
return 0;
}
```

Решение.

Бесконечный цикл. Первая итерация.
При создании процесса в UNIX-подобной ОС автоматически открываются три файла с пользовательскими дескрипторами: 0 – для файла стандартного ввода, 1 – для файла стандартного вывода, 2 – для файла стандартного протокола ошибок. Эта информация отражается в таблице пользовательских дескрипторов открытых файлов процесса(ТПДФ), в которой строки с индексами 0, 1 и 2, оказываются занятыми.

С выполнением системного вызова *creat()* создается и открывается на запись файл с пользовательским дескриптором *a=3* с именем, которое указывается в строковой переменной *name.* Информация об этом файле отображается в строке ТПДФ с индексом 3. Далее выполняется системный вызов (СВ) *pipe(p).* По его завершению открываются два файла с пользовательскими дескрипторами *p[0] и p[1],* и эта информация прописывается в свободные строки таблицы пользовательских дескрипторов файлов процесса, т.е. файл с пользовательским дескриптором *p[0]=4* прописывается в строку с индексом 4 и файл с пользовательским дескриптором *p[1]=5* – в строку с индексом 5. После этой операции выполняется системный вызов *fork().* Он создает копию текущего процесса – дочерний процесс, который начинает параллельно функционировать с родительским процессом. Таблица пользовательских дескрипторов открытых файлов процесса у дочернего процесса идентична ТПДФ родительского процесса.

Т.к. родительский процесс выполняет СВ *wait(),* - он будет ждать завершения работы дочернего процесса.

Дочерний процесс закрывает файлы с пользовательскими дескрипторами 4 и 1. Далее, он с помощью СВ *dup()* копирует файл с пользовательским дескриптором 5 на первое свободное место в ТПДФ, т.е. в строку с индексом 1. Теперь в качестве стандартного файла вывода используется файл с пользовательским дескриптором *p[1],* т.е. межпроцессный канал, и результат выполнения команд *"pwd" и "ls –l"* запишется вместо монитора в межпроцессный канал (в «трубу»). После этого с помощью СВ *exit()* дочерний процесс завершает свое существование, тем самым автоматически закрывая все открытые в нем файлы.

Дождавшись завершения дочернего процесса, родительский процесс закрывает файл с пользовательским дескриптором *p[1]=5* и считывает из межпроцессного канала (

из файла с пользовательским дескриптором *p[0]=4*) все, что в нем есть, т.е. *d* символов (байтов) в строку *buf*, начиная с 0-ого элемента, т.к. запись аргумента *buf* в СВ *read()* идентична записи *&buf[0]*.

После этого закрываются файлы с пользовательскими дескрипторами *p[0] и 1*. Далее, копируется строка ТПДФ с пользовательским дескриптором *a=3* на 1-ую позицию ТПДФ, т.е. в качестве файла стандартного вывода будет использоваться файл, имя которого определено в строке *name.* Содержимое строки *buf* переписывается с помощью функции *printf()* в файл, имя которого определено в строке *name* и закрывается файл с пользовательским дескриптором 1 *("name").*

Вторая итерация.

Процесс продолжает функционировать. В ТПДФ отмечены открытые файлы в строках с индексами 0, 2 и 3. После выполнения СВ *creat()* в ТПДФ на первой свободной позиции появляется файл с именем, определенном в строке *name.* Это свободная строка с индексом 1. Т.е. значение пользовательского дескриптора этого открытого файла *a=1.* Далее, после выполнения СВ *pipe()* открываются файлы с пользовательскими дескрипторами *p[0]=4* и *p[1]=5* и порождается (СВ *fork()*) дочерний процесс. ТПДФ у родительского процесса и дочернего процесса одинаковы. Процессы функционируют параллельно.

Т.к. родительский процесс выполняет СВ *wait(),* - он ждет завершения дочернего процесса. После завершения дочернего процесса родительский процесс закрывает файл с пользовательским дескриптором *p[1]=5,* считывает из файла с пользовательским дескриптором *p[0]=4 d* байтов, закрывает файл с пользовательским дескриптором *p[0]=4* и закрывает файл с пользовательским дескриптором *1 = a,* а этого делать **нельзя**, т.к. при выполнении следующего СВ *dup2()* произойдет ошибка.

Таким образом, использование инструкции *close(1)* в приведенном фрагменте программного кода – некорректно.