

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО
ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНОЙ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ

ОТЧЕТ О ПРАКТИКЕ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

к.ф.-м.н., доцент
должность, уч. степень звание

подпись, дата

М.В. Фаттахова
инициалы, фамилия

ОТЧЕТ ПО УЧЕБНО-ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКЕ

**РАЗРАБОТКА ПРОГРАММЫ ДЛЯ РЕШЕНИЯ НЕЛИНЕЙНЫХ
УРАВНЕНИЙ МЕТОДОМ ХОРД**

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. Z7431

06.01.2018
подпись, дата

М.Д.Семочкин
инициалы, фамилия

Санкт-Петербург
2018

Содержание

1. Математическая постановка задачи	3
2. Описание метода численной реализации задания	3
3. Описание ручного счета тестового примера предложенным методом	4
4. Разработка алгоритмов решения задачи	5
5. Листинг программы	7
6. Результат работы программы	7
7. Результат реализации метода с помощью встроенных функций	8
8. Сравнение обоих результатов	9
9. Заключение	9
10. Список использованной литературы	10

Математическая постановка задачи

Разработать алгоритм и программную реализацию заданного математического метода в виде функции на языке программирования MATLAB в среде MatLab, FreeMat, Octave или SciLab. Разработанная функция должна быть снабжена пользовательским интерфейсом. Для недопустимых значений входных данных, при которых невозможно провести вычисления, должно отображаться сообщение об ошибке. Также необходимо провести сравнение разработанной функции со встроенными функциями используемого математического пакета (MatLab, FreeMat, Octave), решающими те же задачи.

В соответствии с вариантом 12:

Решение нелинейного уравнения методом хорд. Решите уравнение

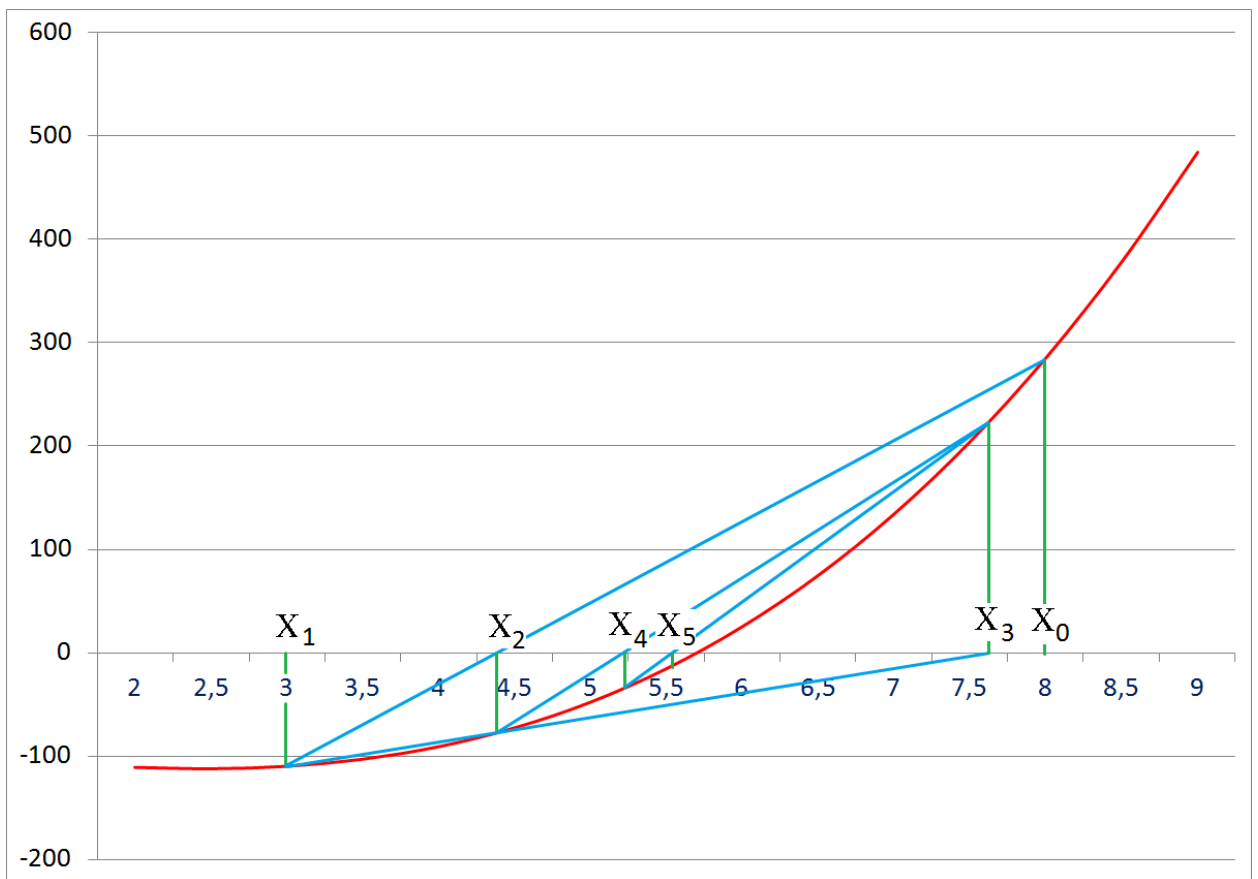
$$3e^{-5t} \sin(7t + 0,1) = 1$$

методом хорд. Для отделения корней используйте графический метод.

Описание метода численной реализации задания

Метод хорд — итерационный численный метод приближённого нахождения корня уравнения.

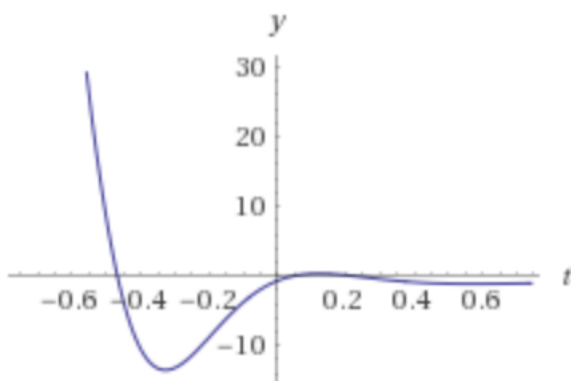
Будем искать нуль функции $f(x)$. Выберем две начальные точки $C_1(x_1; y_1)$ и $C_2(x_2; y_2)$ и проведем через них прямую. Она пересечет ось абсцисс в точке $(x_3; 0)$. Теперь найдем значение функции с абсциссой x_3 . Временно будем считать x_3 корнем на отрезке $[x_1; x_2]$. Пусть точка C_3 имеет абсциссу x_3 и лежит на графике. Теперь вместо точек C_1 и C_2 мы возьмём точку C_3 и точку C_2 . Теперь с этими двумя точками проделаем ту же операцию и так далее, то есть будем получать две точки C_{n+1} и C_n и повторять операцию с ними. Отрезок, соединяющий последние две точки, пересекает ось абсцисс в точке, значение абсциссы которой можно приближённо считать корнем. Эти действия нужно повторять до тех пор, пока не получим значение корня с нужным приближением. [1]



Описание ручного счета тестового примера предложенным методом

Возьмем функцию $f(t) = 3e^{-5t} \sin(7t + 0,1) - 1$

Построим график функции и найдем интервал, на котором функция меняет свое значение:



Возьмем интервал $t_{\text{Min}} = -0.6$; $t_{\text{Max}} = -0.4$, точность 0.001

Пока $t_{\text{Max}} - t_{\text{Min}}$ меньше заданной точности, будем вычислять новые значения t_{Max} и t_{Min} по формулам:

$$t_{\text{Min}} = t_{\text{Max}} - (t_{\text{Max}} - t_{\text{Min}}) * f(t_{\text{Max}}) / (f(t_{\text{Max}}) - f(t_{\text{Min}}))$$

$$t_{\text{Max}} = t_{\text{Min}} + (t_{\text{Min}} - t_{\text{Max}}) * f(t_{\text{Min}}) / (f(t_{\text{Min}}) - f(t_{\text{Max}}))$$

Результаты вычислений:

Шаг	t	F(t)	tMax - tMin
1	-0.436	-6.0585	0.1644
2	-0.454	-2.8543	0.1460
3	-0.462	-1.2080	0.1379
4	-0.465	-0.4874	0.1345
5	-0.467	-0.1928	0.1332
6	-0.467	-0.0756	0.1327

На 6-м шаге значение найденного корня перестало меняться для выбранной нами точности, поэтому вычисления завершены.

Разработка алгоритмов решения задачи

Разработаем алгоритм и блок-схему для решения задачи:

Псевдокод:

Ввести функцию fun , погрешность $epsilon$ и интервал t_{Min} , t_{Max} с клавиатуры;

Найти значения функции в точках, являющимися границами интервалов f_{ValMin} и f_{ValMax} ;

Если f_{ValMin} и f_{ValMax} являются оба положительными или отрицательными

 Вывести сообщение об ошибке;

 Завершить работу программы;

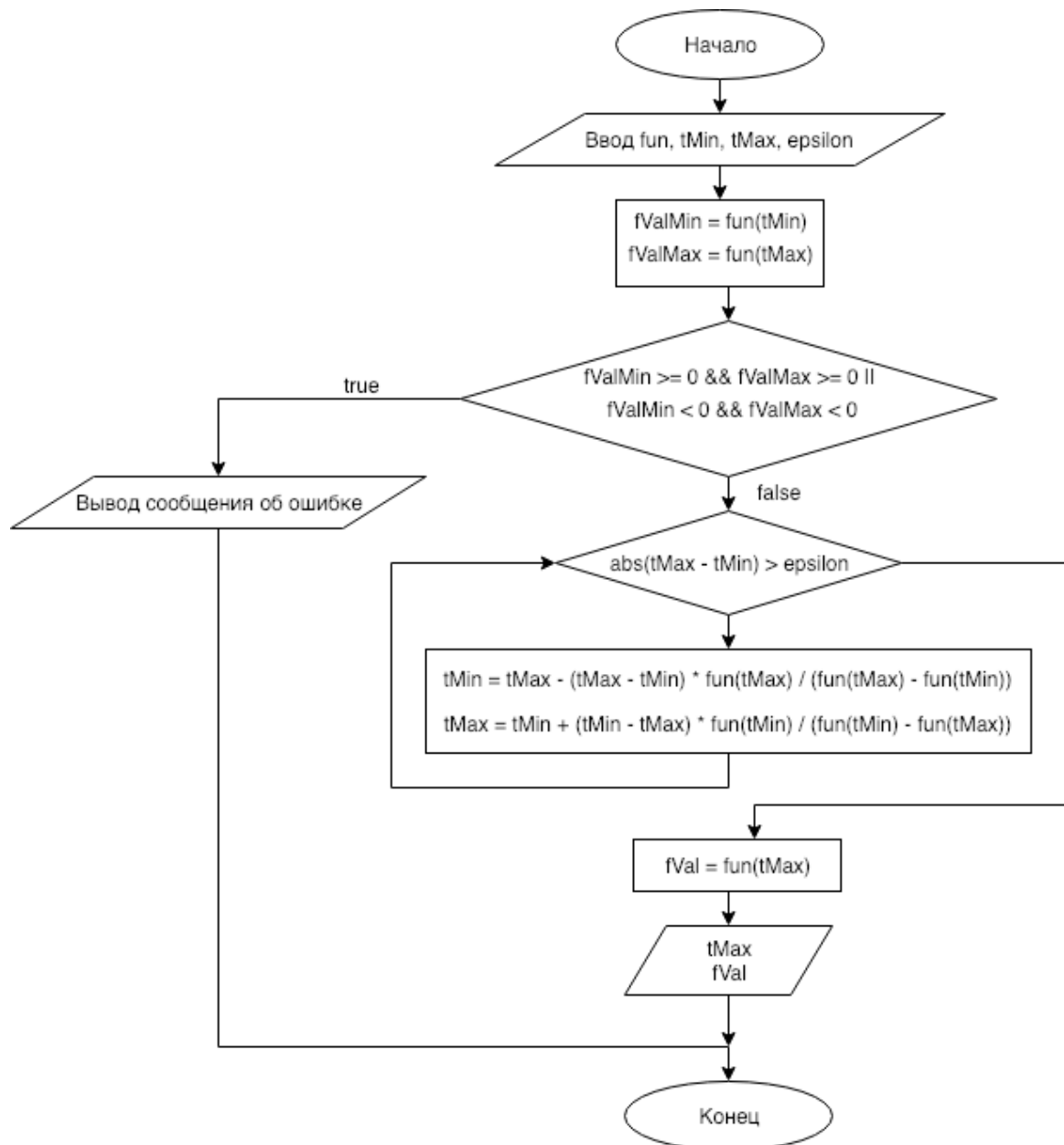
Пока $abs(t_{\text{Max}} - t_{\text{Min}}) > epsilon$

$t_{\text{Min}} = t_{\text{Max}} - (t_{\text{Max}} - t_{\text{Min}}) * fun(t_{\text{Max}}) / (fun(t_{\text{Max}}) - fun(t_{\text{Min}}))$;

$$tMax = tMin + (tMin - tMax) * fun(tMin) / (fun(tMin) - fun(tMax));$$

Вычислить значение функции в найденном корне $fVal = fun(tMax);$
Вывести результат;

Блок-схема:

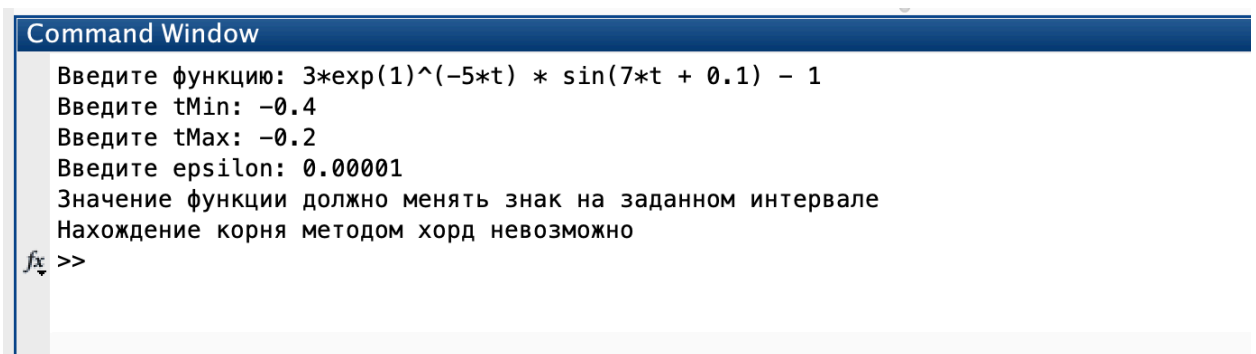


Листинг программы

```
1. clc; clear all; close all;
2.
3. % объявление функции
4. % fun = inline('3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1');
5. % disp('f(t) = 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1');
6. userInput = input('Введите функцию: ', 's');
7. fun = inline(userInput);
8.
9. % ввод переменных:
10. % tMin, tMax - пределы хорды
11. tMin = input('Введите tMin: ');
12. tMax = input('Введите tMax: ');
13. % epsilon - погрешность
14. epsilon = input('Введите epsilon: ');
15.
16. % проверить, что функция меняет знак на введенном интервале
17. fValMin = fun(tMin);
18. fValMax = fun(tMax);
19. if (fValMin >= 0 && fValMax >= 0 || fValMin < 0 && fValMax < 0)
20.     disp('Значение функции должно менять знак на заданном интервале');
21.     disp('Нахождение корня методом хорд невозможно');
22.     return;
23. end
24.
25. % вычислять методом хорд, пока погрешность больше требуемой
26. min = tMax;
27. max = tMax;
28. while(abs(tMax - tMin) > epsilon)
29.     tMin = tMax - (tMax - tMin) * fun(tMax) / (fun(tMax) - fun(tMin));
30.     tMax = tMin + (tMin - tMax) * fun(tMin) / (fun(tMin) - fun(tMax));
31. end
32. fVal = fun(tMax);
33.
34. % вывод результата
35. fprintf('\nКорнем уравнения является число %5.7f\n', tMax);
36. fprintf('Функция при таком значении t равна %5.7f\n\n', fVal);
```

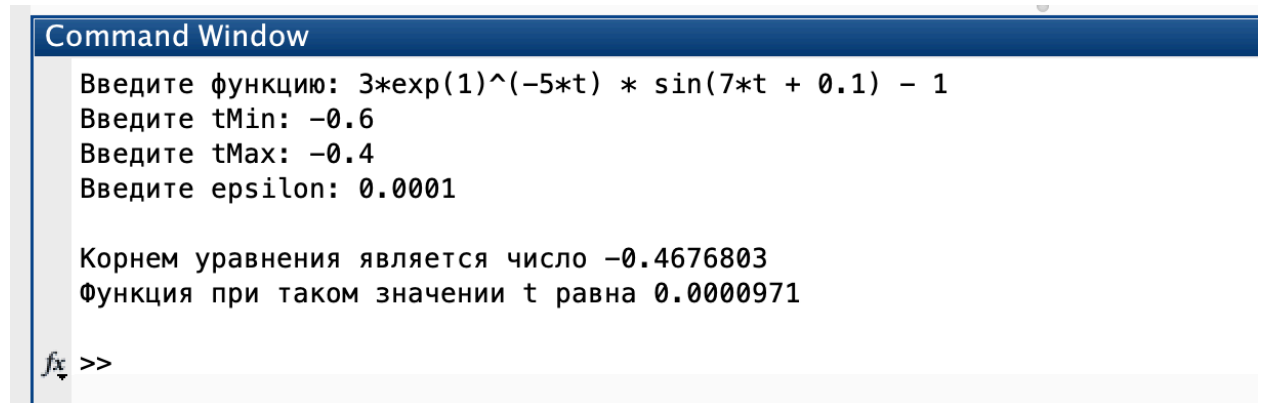
Результат работы программы

При вводе интервала, на котором значение функции не меняет знак, выводится сообщение об ошибке:



```
Command Window
Введите функцию: 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1
Введите tMin: -0.4
Введите tMax: -0.2
Введите epsilon: 0.00001
Значение функции должно менять знак на заданном интервале
Нахождение корня методом хорд невозможно
fx >>
```

При вводе подходящих для вычисления значений вычисляется и выводится результат:



```
Command Window

Введите функцию: 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1
Введите tMin: -0.6
Введите tMax: -0.4
Введите epsilon: 0.0001

Корнем уравнения является число -0.4676803
Функция при таком значении t равна 0.0000971

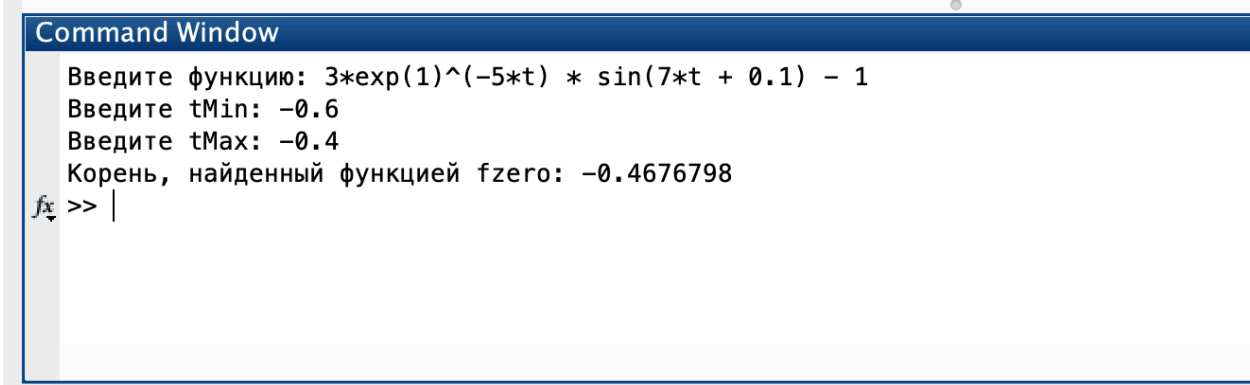
fx >>
```

Результат реализации метода с помощью встроенных функций

Реализуем программу с помощью встроенной функции для нахождения корней нелинейного уравнения `fzero()`:

```
1. clc; clear all; close all;
2.
3. % объявление функции
4. % fun = inline('3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1');
5. % disp('f(t) = 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1');
6. userInput = input('Введите функцию: ', 's');
7. fun = inline(userInput);
8.
9. % ввод переменных:
10. % tMin, tMax - пределы хорды
11. tMin = input('Введите tMin: ');
12. tMax = input('Введите tMax: ');
13.
14. % найти корень на данном интервале с помощью
15. % встроенной функции fzero()
16. interval = [tMin tMax]; % интервал
17. fZeroResult = fzero(fun, interval);
18.
19. % вывод результата
20. fprintf('Корень, найденный функцией fzero: %5.7f\n', fZeroResult);
```


Запустим ее:

A screenshot of the MATLAB Command Window. The title bar is blue with the text 'Command Window'. The window contains the following text: 'Введите функцию: 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1', 'Введите tMin: -0.6', 'Введите tMax: -0.4', 'Корень, найденный функцией fzero: -0.4676798', and a prompt 'fx >> |' with a cursor.

```
Command Window
Введите функцию: 3*exp(1)^(-5*t) * sin(7*t + 0.1) - 1
Введите tMin: -0.6
Введите tMax: -0.4
Корень, найденный функцией fzero: -0.4676798
fx >> |
```

Сравнение обоих результатов

При вычислении написанной во время выполнения практики с точностью 0.0001 на интервале [-0.6, -0.4] получен результат -0.4676803

При вычислении встроенной функцией fzero() с теми же параметрами был получен результат -0.4676798

При сравнении результатов с выбранной точностью (0.0001) результаты совпадают.

Заключение

Был изучен метод хорд, и способ реализации решения нелинейных уравнений этим методом при помощи MATLAB.

Результаты вычислений с помощью написанной программы и с помощью встроенной функции fzero совпадают, из чего можно сделать вывод, что программа написана правильно.

Список использованной литературы

1. Ханова А.А, Численное решение уравнений и систем уравнений
- АГТУ, Астрахань, 2001
2. Иванова Г.С. Основы программирования: Учебник для вузов.-М.: Изд-во
- МГТУ им. Н.Э.Баумана, 2001;
3. Бураков М.В. Основы работы в MATLAB: учебное пособие/ М.В. Бураков
.- ГУАП.СПб., 2006;
4. Васильев Ф.П. Численные методы решения экстремальных задач. Учебное пособие для студ. ВУЗов
- М.,НАУКА 1982, 552 с., [Шифр 519.6/8,В-19]
5. Волков Е.А. Численные методы: учебное пособие
- М., Наука, 1982,254 с.;
6. Самарский А.А. Введение в численные методы. Учебное пособие для Вузов,
- М., Наука,1987,288 с.;
7. Мудров А.Е. Численные методы для ПЭВМ на языках Бейсик, Фортран, Паскаль.
- Томск, МП "Раско",1992,406 с.;
8. Бахвалов Н. С., Лапин А. В., Чижонков Е. В. Численные методы в задачах и упражнениях. Учеб. пособие.
- Высш. шк. 2000. – 190 с.
9. Дьяконов, В. П. MATLAB 7.*/R2006/R2007: самоучитель.
– М.: ДМК Пресс, 2008. – 767 с.