

# Отчет по программе

Концевая Маргарита, ИУ8-54, 30 вариант задания

## Оглавление

Задание .....	1
Теоретическая часть.....	2
Методы решения .....	3
Инструкция по использованию программы .....	3
Описание логики работы программы.....	3
Формат входных и выходных данных.....	3
Входные данные .....	3
Выходные данные .....	4
Тесты .....	4
Оценка программы .....	4
Оценка сложности .....	4
Оценка используемой памяти.....	4

## Задание

Реализуйте программу аналитического вычисления производной функции

## Теоретическая часть

Вычисление производной производится по следующим правилам:

**1) Производная суммы равна сумме производных**

$$(u \pm v)' = u' \pm v'$$

**2) Производная произведения функций**

$$(uv)' = u'v + uv'$$

**3) Производная частного функций**

$$\left(\frac{u}{v}\right)' = \frac{u'v - uv'}{v^2}$$

**4) Производная сложной функции**

$$(u(v))' = u'(v) \cdot v'$$

### ПРОИЗВОДНЫЕ ЭЛЕМЕНТАРНЫХ ФУНКЦИЙ

Функция	Производная
$f(x) = c$	$c' = 0$ , где $c$ — const
$f(x) = x^a$	$(x^a)' = ax^{a-1}$
$f(x) = e^x$	$(e^x)' = e^x$
$f(x) = a^x$	$(a^x)' = a^x \ln a$
$f(x) = \ln x$	$(\ln x)' = \frac{1}{x}$
$f(x) = \log_a x$	$(\log_a x)' = \frac{1}{x \ln a}$
$f(x) = \sin x$	$(\sin x)' = \cos x$
$f(x) = \cos x$	$(\cos x)' = -\sin x$
$f(x) = \operatorname{tg} x$	$(\operatorname{tg} x)' = \frac{1}{\cos^2 x}$
$f(x) = \operatorname{ctg} x$	$(\operatorname{ctg} x)' = -\frac{1}{\sin^2 x}$
$f(x) = \arcsin x$	$(\arcsin x)' = \frac{1}{\sqrt{1-x^2}}$
$f(x) = \arccos x$	$(\arccos x)' = -\frac{1}{\sqrt{1-x^2}}$
$f(x) = \operatorname{arctg} x$	$(\operatorname{arctg} x)' = \frac{1}{1+x^2}$
$f(x) = \operatorname{arcctg} x$	$(\operatorname{arcctg} x)' = -\frac{1}{1+x^2}$

## Методы решения

Для решения данной задачи используется автоматическое дифференцирование, которое позволяет вычислить производную на этапе компиляции, но данный метод нам не подходит, т.к. выражение вводится пользователем только на этапе выполнения.

Другим методом нахождения аналитической производной является построение синтаксического дерева решения. В данной работе используется рекурсивный вызов функции производной от строки, которая разбивается на элементарные выражения, которые далее обрабатываются согласно правилам дифференцирования и таблице производных. Данный метод похож на построение синтаксического дерева, которое в данном случае осуществляется выделением элементарных частей и взятие производных от них.

## Инструкция по использованию программы

Для использования программы необходимо собрать проект, следуя инструкциям, приведенным на официальном сайте Microsoft (для конкретно вашего оборудования и окружения).

Например:

`-cl /EHsc derivative.cpp`

## Описание логики работы программы

Программа работает следующим образом:

1. При запуске программы выводится поле для введения входных данных. Данные проверяются на корректность и, если они проходят проверку, то далее вычисляется решение.
2. Решение задачи осуществляется следующим образом:
  - Считанная корректная строка подаётся в рекурсивную функцию `difer`.
  - В строке выполняется поиск операторов в следующем порядке: сложение, вычитание, умножение, деление, степень, тригонометрические функции, логарифм.
  - Позиции найденных операторов записываются в отдельные вектора.
  - Выполняется проверка на пустоту векторов позиций в порядке приоритета операторов. В случае непустоты данного вектора в соответствии с правилами дифференцирования вычисляется производная от подстроки, полученных в результате разбиения исходной строки по позициям в порядке приоритетности.
  - Вышеперечисленные действия повторяются до тех пор, пока подстроки не будут представлять собой элементарные функции: в функцию `difer` подаётся подстрока исходной строки, вектора позиций перезаполняются, и т.д.
3. В конце работы алгоритма решения получаем выходные данные: строка с вычисленной производной функции.

## Формат входных и выходных данных

### Входные данные

На вход программы подаётся строка, содержащая функцию от  $x$ .

Используемые знаки операций:  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $^$ ,  $($ ,  $)$ .

Элементарные функции записываются в виде:  $\cos(x)$ ,  $\sin(x)$ ,  $\operatorname{tg}(x)$ ,  $\operatorname{ctg}(x)$ ,  $\log(a, x)$ ,  $\ln(x)$  - где  $x$  может быть также функцией, зависящей от  $x$ , а  $a$  – основание логарифма.

Константы:  $e$  – экспонента,  $\pi$  – число Пи.

## Выходные данные

На выходе выводится строка, содержащая вычисленную производную от исходной функции.

## Тесты

К программе приложены тесты:

Тесты, проверяющие корректность работы программы на элементарных функциях:

1. Plus.in
2. Minus.in
3. Mult.in
4. Div.in
5. Pow.in
6. Log.in
7. Cos.in
8. Sin.in
9. Tg.in
10. Ctg.in
11. Arcsin.in
12. Arccos.in
13. Arctg.in
14. Arcctg.in

Тест, проверяющий работу программы на сложных функциях

Complicated.txt

Тест, проверяющий корректную обработку некорректного ввода:

Incorrect.txt

В результате проверки тестами установлено, что программа работает корректно. Для запуска тестов файл *test.cpp*

## Оценка программы

### Оценка сложности

Сложность зависит от количества вызовов функции взятия производной `dif`, которое в свою очередь зависит от количества операторов во вводимой строке. Обозначим количество операторов числом  $n$ , а длину строки – числом  $l$ . Тогда функция будет вызываться  $\approx 2^n$  раз, и т.к. на каждом вызове она будет заново считывать строку для поиска операторов, сложность в худшем случае составит:  $O(l \cdot 2^n)$ .

### Оценка используемой памяти

Программа хранит в себе строку, исходная длина которой  $= l$ , и на каждом шаге функции хранит новую строку, которая является частью исходной. Строка будет разбиваться на подстроки  $n$  раз, тогда сложность по памяти составит:  $O(l \cdot n)$ .