Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное   
учреждение высшего образования

Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

**Отчет по лабораторной работе**

**«Вычисление синуса, косинуса, экспоненты и натурального логарифма»**

**Выполнил**:

студент группы 3821Б1ПМ2

Ермолаев Д.А.

**Проверил**:

преподаватель каф. МОСТ,

Волокитин В.Д.

Нижний Новгород

2022

Оглавление

**Постановка задачи3**

**Методы решения4**

**Руководство** **пользователя5**

**Описание программной реализации** **6**

**Подтверждение** **корректности7**

**Результаты экспериментов8**

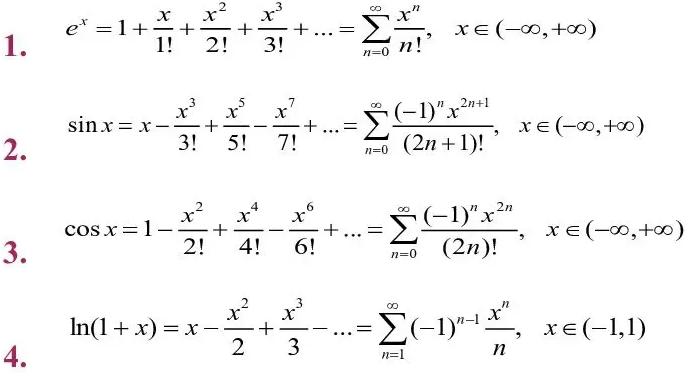
**Заключение12**

**Приложение13**

# Постановка задачи

Целью лабораторной работы является разобрать идею и реализовать на языке программирования Си вычисление функций синуса, косинуса, экспоненты, натурального логарифма с помощью рядов Маклорена. Вычисления я реализовал для данных типа float. Нужно описать программную реализацию и алгоритмы работы данных вычислений. Необходимо подтвердить корректность реализации путём вычисления абсолютной и относительной погрешности данных вычислений. Сделать вывод, какой метод суммирования более точен для определённых функций.

**Методы решения**

Для вычисления дынных функций использовались ряды Маклорена.

Как видно по формулам, ряды Маклорена не имеют определённого последнего элемента, чем больше слагаемых, тем выше точность вычисления. В своей работе я считал функции используя 100 слагаемых. Вычисления функций можно производить 3 способами суммирования:

1. Прямое суммирование (суммирование начинается с первого элемента)
2. Обратное суммирование (суммирование начинается с последнего элемента)
3. Попарное суммирование (сначала суммирование 1-го слагаемого с 2, 3 с 4 и т.д, потом суммирование этих частей) (имеет смысл для синуса, косинуса и логарифма)

Руководство пользователя

Пользователю предлагается выбрать функцию (sin, cos, exp, ln) и число, от которого нужно будет взять эту функцию. Программа высчитывает результат и выводит на экран этот результат, число, от которого берется функция, абсолютную и относительную погрешность от стандартной функции из библиотеки “math.h”.

Описание программной реализации

Библиотеки, которые использовались в работе: “stdio.h”, “math.h”.

Основные функции:

1. float sin\_next(float pre, int i, float x) – функция возвращает следующее слагаемое ряда Маклорена для синуса.
2. float cos\_next(float pre, int i, float x)– функция возвращает следующее слагаемое ряда Маклорена для косинуса.
3. float ln\_next(float pre, int i, float x) – функция возвращает следующее слагаемое ряда Маклорена для натурального логарифма.
4. float exp\_next(float pre, int i, float x)– функция возвращает следующее слагаемое ряда Маклорена для экспоненты.
5. float sin\_ln\_first(float x) – функция возвращает первое слагаемое ряда Маклорена для функций синуса и натурального логарифма.
6. float cos\_exp\_first() – функция возвращает первое слагаемое ряда Маклорена для функций косинуса и экспоненты.
7. float directsum(float\* mas) – прямое суммирование. Возвращает значение функции от числа x.
8. float reversesum(float\* mas) – обратное суммирование. Возвращает значение функции от числа x.
9. float pairwisesum(float\* mas) – попарное суммирование. Возвращает значение функции от числа x.
10. void create\_mas(float\* mas, float(\*next)(float, int, float), float x) – создаёт массив из 100 слагаемых ряда Маклорена для каждой функции.
11. void printfunc(float func, double original, float x)– выводит на экран число, от которого берётся функция, значение функции от числа x, абсолютную и относительную погрешность.

Подтверждение корректности

Для подтверждения корректности в программе использовал стандартные функции (sin(), cos(), exp(), log()) из стандартной библиотеки “math.h” и высчитывал относительную и абсолютную погрешность относительно результата стандартной функции и результата реализованной функции.

**Результаты экспериментов**

Результаты экспериментов приведены в виде таблицы (1-й столбец - число, от которого берётся функция, 2, 3 и 4 столбцы относительная погрешность (%) разных видов суммирования). Число, от которого берётся функция, менялось от - -8 до 8 с интервалом в 0.5.

|  |  |  |  |
| --- | --- | --- | --- |
| x | Прямое сумм. | Попарное сумм. | Обратное сумм. |
| -8.0 | 0.00167486 | 0.00169294 | 0.00012655 |
| -7.5 | 0.00055145 | 0.00055145 | 0.00037988 |
| -7.0 | 0.00139784 | 0.00137062 | 0.00005512 |
| -6.5 | 0.00013178 | 0.00018719 | 0.00103887 |
| -6.0 | 0.00100616 | 0.00101683 | 0.00149680 |
| -5.5 | 0.00007129 | 0.00010508 | 0.00008819 |
| -5.0 | 0.00001440 | 0.00000818 | 0.00000197 |
| -4.5 | 0.00001787 | 0.00001787 | 0.00006055 |
| -4.0 | 0.00002415 | 0.00002415 | 0.00000840 |
| -3.5 | 0.00005131 | 0.00005981 | 0.00003432 |
| -3.0 | 0.00005883 | 0.00004827 | 0.00003620 |
| -2.5 | 0.00001426 | 0.00000430 | 0.00000430 |
| -2.0 | 0.00000222 | 0.00000222 | 0.00000434 |
| -1.5 | 0.00000688 | 0.00000090 | 0.00000090 |
| -1.0 | 0.00000333 | 0.00000333 | 0.00000333 |
| -0.5 | 0.00000394 | 0.00000394 | 0.00000227 |
| 0.0 | 0.00000000 | 0.00000000 | 0.00000000 |
| 0.5 | 0.00000394 | 0.00000394 | 0.00000227 |
| 1.0 | 0.00000333 | 0.00000333 | 0.00000333 |
| 1.5 | 0.00000688 | 0.00000090 | 0.00000090 |
| 2.0 | 0.00000222 | 0.00000222 | 0.00000434 |
| 2.5 | 0.00001426 | 0.00000430 | 0.00000430 |
| 3.0 | 0.00005883 | 0.00004827 | 0.00003620 |
| 3.5 | 0.00005131 | 0.00005981 | 0.00003432 |
| 4.0 | 0.00002415 | 0.00002415 | 0.00000840 |
| 4.5 | 0.00001787 | 0.00001787 | 0.00006055 |
| 5.0 | 0.00001440 | 0.00000818 | 0.00000197 |
| 5.5 | 0.00007129 | 0.00010508 | 0.00008819 |
| 6.0 | 0.00100616 | 0.00101683 | 0.00149680 |
| 6.5 | 0.00013178 | 0.00018719 | 0.00103887 |
| 7.0 | 0.00139784 | 0.00137062 | 0.00005512 |
| 7.5 | 0.00055145 | 0.00055145 | 0.00037988 |
| 8.0 | 0.00167486 | 0.00169294 | 0.00012655 |

**Синус**

Вывод: из таблицы видно, что обратное суммирование считает лучше остальных, за редким исключением. Результаты попарного и прямого суммирований очень близки друг к другу (или одинаковы), но у попарного суммирования в среднем погрешность меньше.

**Косинус**

|  |  |  |  |
| --- | --- | --- | --- |
| x | Прямое сумм. | Попарное сумм. | Обратное сумм. |
| -8.0 | 0.00049178 | 0.00050202 | 0.00514096 |
| -7.5 | 0.00197918 | 0.00197058 | 0.00480779 |
| -7.0 | 0.00000028 | 0.00000819 | 0.00048199 |
| -6.5 | 0.00022912 | 0.00023522 | 0.00022912 |
| -6.0 | 0.00010116 | 0.00010737 | 0.00034947 |
| -5.5 | 0.00001578 | 0.00006833 | 0.00001578 |
| -5.0 | 0.00043615 | 0.00042564 | 0.00045716 |
| -4.5 | 0.00001156 | 0.00001156 | 0.00026413 |
| -4.0 | 0.00003843 | 0.00003843 | 0.00000195 |
| -3.5 | 0.00000075 | 0.00000562 | 0.00000075 |
| -3.0 | 0.00000574 | 0.00000630 | 0.00001232 |
| -2.5 | 0.00000383 | 0.00001127 | 0.00000383 |
| -2.0 | 0.00000522 | 0.00000522 | 0.00001954 |
| -1.5 | 0.00000534 | 0.00000534 | 0.00002626 |
| -1.0 | 0.00000541 | 0.00000541 | 0.00000541 |
| -0.5 | 0.00000544 | 0.00000544 | 0.00000135 |
| 0.0 | 0.00000000 | 0.00000000 | 0.00000000 |
| 0.5 | 0.00000544 | 0.00000544 | 0.00000135 |
| 1.0 | 0.00000541 | 0.00000541 | 0.00000541 |
| 1.5 | 0.00000534 | 0.00000534 | 0.00002626 |
| 2.0 | 0.00000522 | 0.00000522 | 0.00001954 |
| 2.5 | 0.00000383 | 0.00001127 | 0.00000383 |
| 3.0 | 0.00000574 | 0.00000630 | 0.00001232 |
| 3.5 | 0.00000075 | 0.00000562 | 0.00000075 |
| 4.0 | 0.00003843 | 0.00003843 | 0.00000195 |
| 4.5 | 0.00001156 | 0.00001156 | 0.00026413 |
| 5.0 | 0.00043615 | 0.00042564 | 0.00045716 |
| 5.5 | 0.00001578 | 0.00006833 | 0.00001578 |
| 6.0 | 0.00010116 | 0.00010737 | 0.00034947 |
| 6.5 | 0.00022912 | 0.00023522 | 0.00022912 |
| 7.0 | 0.00000028 | 0.00000819 | 0.00048199 |
| 7.5 | 0.00197918 | 0.00197058 | 0.00480779 |
| 8.0 | 0.00049178 | 0.00050202 | 0.00514096 |

Вывод: в основном обратное суммирование считает лучше остальных способов. В случае с косинусом попарное суммирование считает хуже, чем прямое.

**Экспонента**

Для вычисления экспоненты нет смысла использовать попарное суммирование, т.к. знаки слагаемых ряда Маклорена не меняются.

|  |  |  |
| --- | --- | --- |
| x | Прямое сумм. | Обратное сумм. |
| -8.0 | 0.27855276 | 2.20551044 |
| -7.5 | 1.32569959 | 2.07758993 |
| -7.0 | 0.26826148 | 0.43690046 |
| -6.5 | 0.39701802 | 0.47763316 |
| -6.0 | 0.11924049 | 0.12157937 |
| -5.5 | 0.00609413 | 0.03011343 |
| -5.0 | 0.01194918 | 0.01768533 |
| -4.5 | 0.00318400 | 0.00320077 |
| -4.0 | 0.00096362 | 0.00176703 |
| -3.5 | 0.00022343 | 0.00000480 |
| -3.0 | 0.00016013 | 0.00004789 |
| -2.5 | 0.00003614 | 0.00001799 |
| -2.0 | 0.00001230 | 0.00005634 |
| -1.5 | 0.00000382 | 0.00000286 |
| -1.0 | 0.00001372 | 0.00000249 |
| -0.5 | 0.00001092 | 0.00000109 |
| 0.0 | 0.00000000 | 0.00000000 |
| 0.5 | 0.00000404 | 0.00000319 |
| 1.0 | 0.00000573 | 0.00000304 |
| 1.5 | 0.00000210 | 0.00000210 |
| 2.0 | 0.00000790 | 0.00000145 |
| 2.5 | 0.00002182 | 0.00000616 |
| 3.0 | 0.00000966 | 0.00000017 |
| 3.5 | 0.00001196 | 0.00000044 |
| 4.0 | 0.00000390 | 0.00001088 |
| 4.5 | 0.00000368 | 0.00000368 |
| 5.0 | 0.00000211 | 0.00000817 |
| 5.5 | 0.00000952 | 0.00001576 |
| 6.0 | 0.00000533 | 0.00000979 |
| 6.5 | 0.00001362 | 0.00000444 |
| 7.0 | 0.00002411 | 0.00000185 |
| 7.5 | 0.00001660 | 0.00000310 |
| 8.0 | 0.00002527 | 0.00001708 |

Вывод: в среднем, обратное суммирование считает лучше прямого и, лишь при больших отрицательных числах, прямое суммирование выигрывает.

**Натуральный логарифм**

|  |  |  |  |
| --- | --- | --- | --- |
| x | Прямое сумм. | Попарное сумм. | Обратное сумм. |
| -0.99 | 4.60632294 | 4.68423963 | 4.68423963 |
| -0.93 | 0.00287292 | 0.00311499 | 0.00312396 |
| -0.87 | 0.00001795 | 0.00000626 | 0.00000626 |
| -0.81 | 0.00000138 | 0.00000856 | 0.00000580 |
| -0.75 | 0.00002552 | 0.00000027 | 0.00000027 |
| -0.69 | 0.00000139 | 0.00000139 | 0.00000139 |
| -0.63 | 0.00002357 | 0.00001158 | 0.00000041 |
| -0.57 | 0.00000079 | 0.00000627 | 0.00000079 |
| -0.51 | 0.00000145 | 0.00000145 | 0.00000690 |
| -0.45 | 0.00000130 | 0.00001127 | 0.00000130 |
| -0.39 | 0.00000320 | 0.00000923 | 0.00000283 |
| -0.33 | 0.00001816 | 0.00000417 | 0.00000328 |
| -0.27 | 0.00001336 | 0.00000389 | 0.00000389 |
| -0.21 | 0.00000126 | 0.00000758 | 0.00000126 |
| -0.15 | 0.00002617 | 0.00000133 | 0.00000133 |
| -0.09 | 0.00001377 | 0.00000203 | 0.00000203 |
| -0.03 | 0.00000185 | 0.00000185 | 0.00000185 |
| 0.03 | 0.00000422 | 0.00000422 | 0.00000208 |
| 0.09 | 0.00000185 | 0.00000185 | 0.00000185 |
| 0.15 | 0.00000639 | 0.00000427 | 0.00000427 |
| 0.21 | 0.00000061 | 0.00000061 | 0.00000061 |
| 0.27 | 0.00000146 | 0.00000477 | 0.00000146 |
| 0.33 | 0.00000255 | 0.00000255 | 0.00000255 |
| 0.39 | 0.00002321 | 0.00001416 | 0.00000511 |
| 0.45 | 0.00000402 | 0.00000400 | 0.00000402 |
| 0.51 | 0.00000754 | 0.00000692 | 0.00000031 |
| 0.57 | 0.00001292 | 0.00000690 | 0.00000029 |
| 0.63 | 0.00001351 | 0.00000479 | 0.00000479 |
| 0.69 | 0.00002099 | 0.00001309 | 0.00000173 |
| 0.75 | 0.00000052 | 0.00000052 | 0.00000052 |
| 0.81 | 0.00005690 | 0.00000338 | 0.00000338 |
| 0.87 | 0.00001619 | 0.00000286 | 0.00000286 |
| 0.93 | 0.00043353 | 0.00050923 | 0.00051830 |
| 0.99 | 0.25809000 | 0.26327928 | 0.26329660 |

x должен быть в диапазоне (-1;1) для выполнения условий ряда Маклорена. Число, от которого берётся функция, меняется от -0,99 до 0,99 с интервалом в 0,06.

Вывод: натуральный логарифм начинает сильно расходится в окрестностях -1 и 1 и появляются большие погрешности при любых видах суммирования. Прямое и попарное суммирования, в среднем, считают приблизительно равно. Обратное же суммирование, в основном, считает лучше при любом x.

**Заключение**

В ходе лабораторной работы на языке программирования Си были реализованы 3 метода суммирования рядов Маклорена для получения наиболее точных результатов вычисления функций синуса, косинуса, экспоненты и натурального логарифма. Были описаны алгоритмы работы данных вычислений, их программная реализация и проведены эксперименты для подтверждения их корректности, методом вычисления погрешностей относительно функций из стандартных библиотек. Также, были определены какие виды суммирования лучше справляются в тех или иных случаях.

**Приложение**