**Высокопроизводительные вычисления**

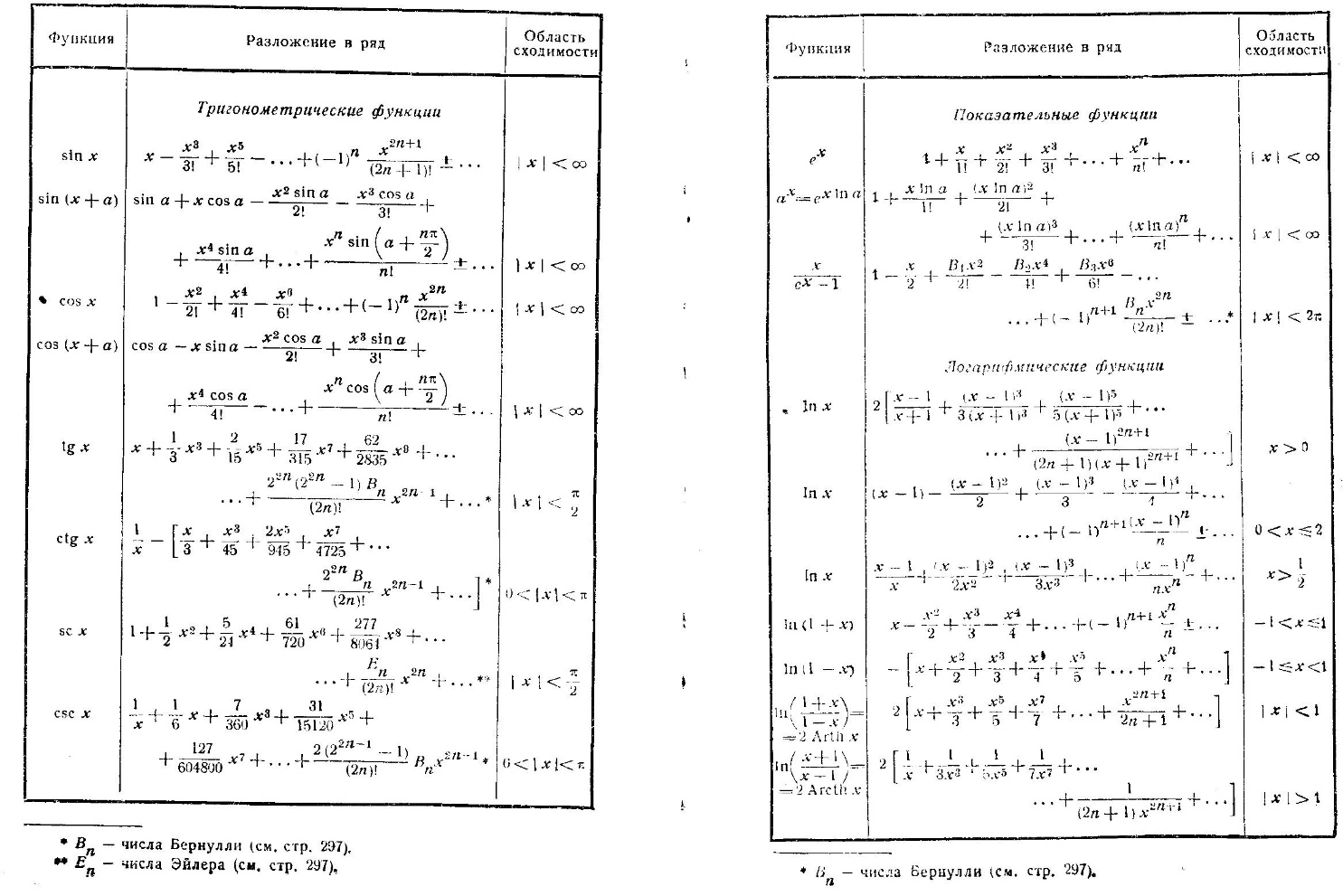
**Индивидуальные функции по лабораторной работе № 2**

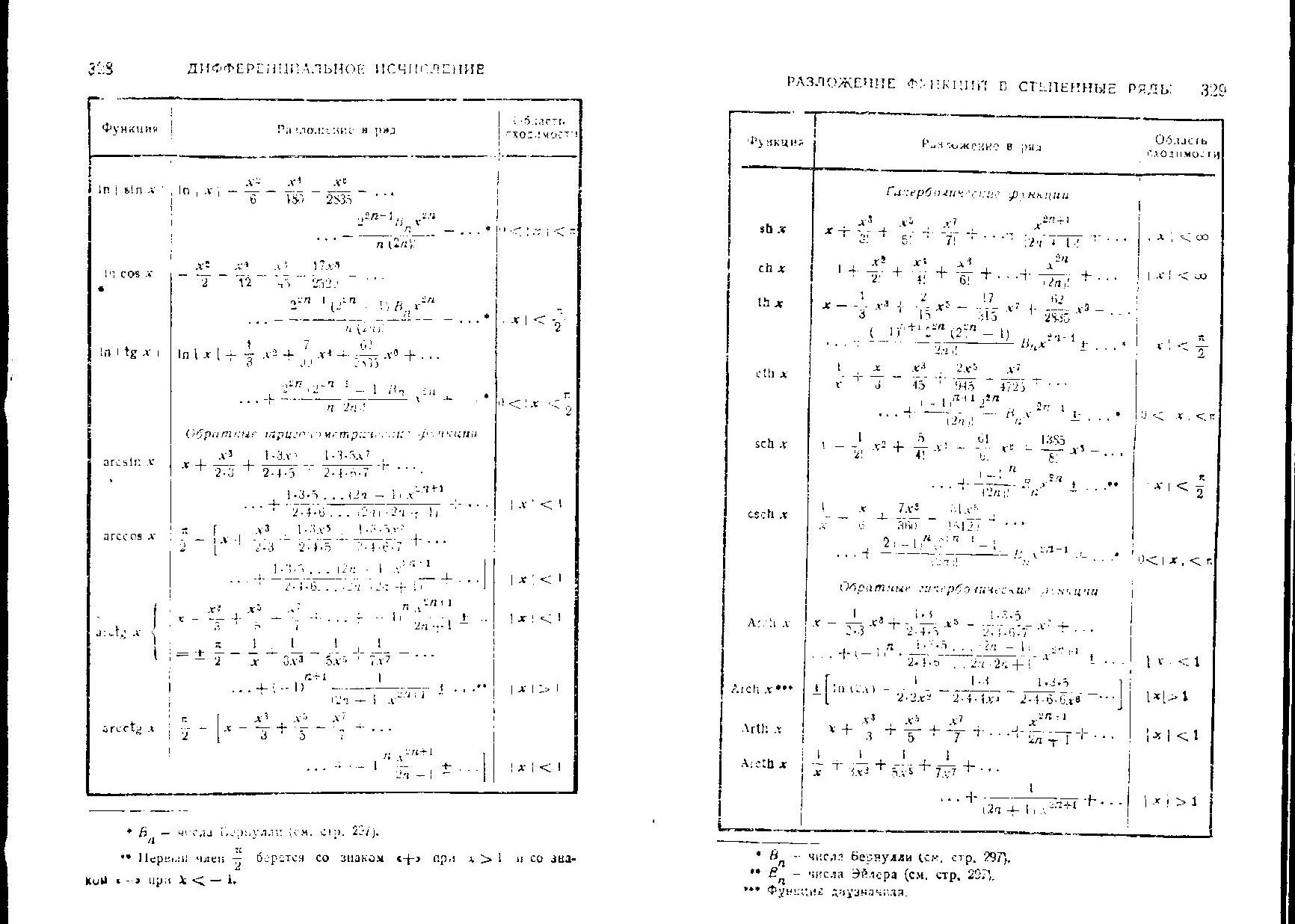
в процедуре

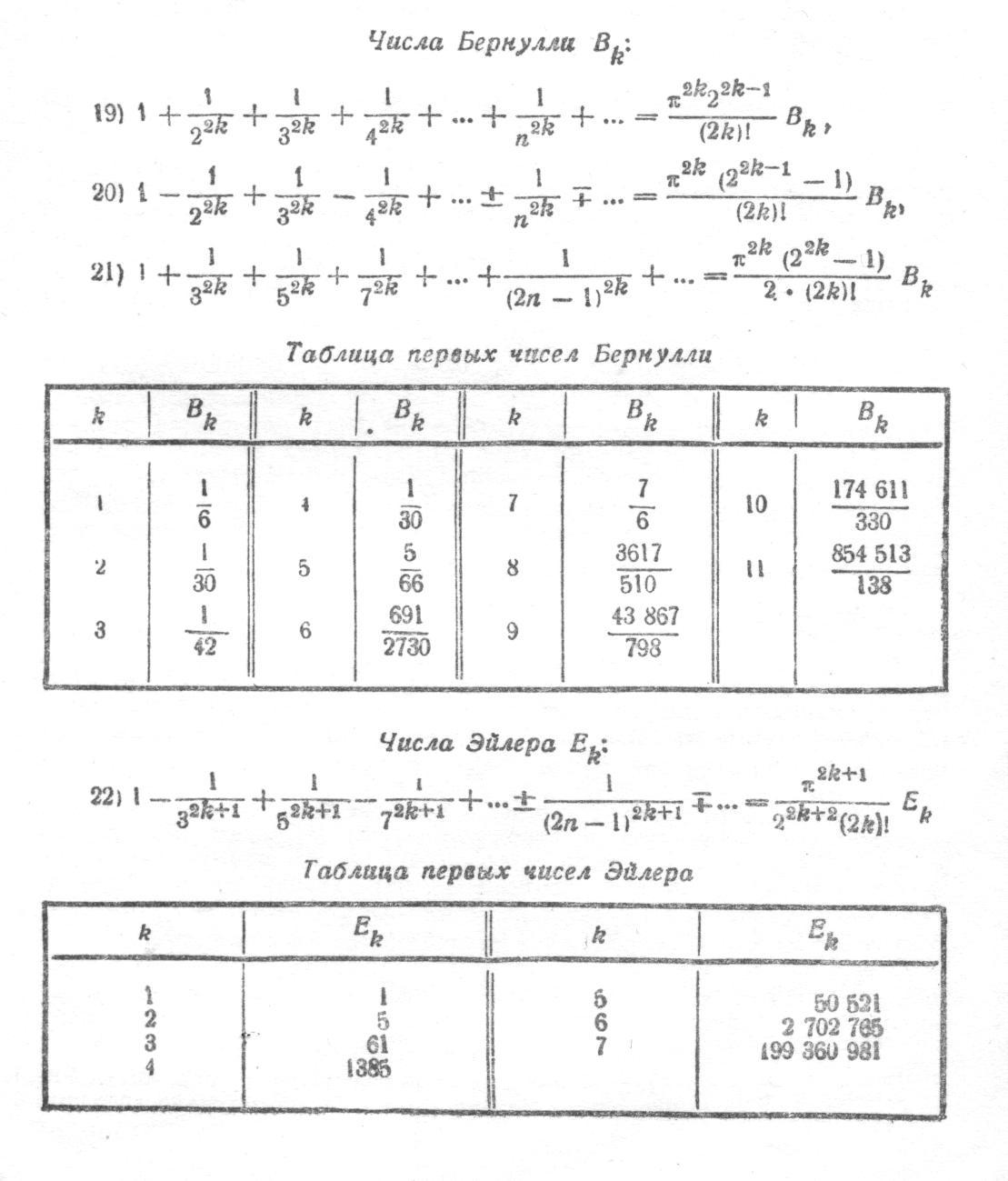
|  |  |  |
| --- | --- | --- |
| **Студент** | **Функция** | **maxErr** |
| **ИВТАПбд-31** | | |
| Астуков Павел | *sin x* | 2-21 |
| Васияров Михаил | *tg x* | 2-18 |
| Егорова Анастасия | *cos x* | 2-22 |
| Кузнеченков Роман | *csc x* | 2-20 |
| Кулакова Екатерина | *ln(1-x)* | 2-16 |
| Мамакин Антон | *ln((1+x)/(1-x))* | 2-15 |
| Никишкин Михаил | *ln(cos x)* | 2-17 |
| Сафронова Анна | *arcsin x* | 2-17 |
| Сорокина Марина | *arcos x* | 2-18 |
| Шишкин Алексей | *arcctg x* | 2-17 |
| Яровова Татьяна | *ch x* | 2-21 |
|  | **ИВТВМбд-31** | |
| Албутов Данил | *sin x* | 2-20 |
| Бахматов Данил | *tg x* | 2-19 |
| Вольтер Григорий | *cos x* | 2-20 |
| Дуванов Андрей | *csc x* | 2-19 |
| Ильиных Кирилл | *ln(1-x)* | 2-15 |
| Медведева Антонида | *ln((1+x)/(1-x))* | 2-16 |
| Родионов Михаил | *ln(cos x)* | 2-18 |
| Рыбин Валерий | *arcsin x* | 2-18 |
| Сатдинов Рустам | *arcos x* | 2-19 |
| Свиязов Павел | *arcctg x* | 2-18 |
| Семеленов Денис | *ch x* | 2-20 |
| Соловьев Михаил | *sch x* | 2-21 |
| Титова Елена | *sin (x + 0,6)* | 2-19 |
| Фаткуллов Р. | *cos (x+ 0,5)* | 2-18 |
| Чильманкина Елена | *ex* | 2-21 |
| Ямщиков Максим | *th x* | 2-20 |

**Приложение 1. Базовые сведения из математического справочника**

Таблица разложения функций в ряды







Числа Бернулли получаются как решения системы равенств:   
  
  
  
  
  
где



Имеем . Или .



Отсюда .



Т.е. рекуррентно можно вычислить числа

**Приложение 2. Схемы вычисления степенных рядов**

2.1. Наивная схема (FlCyclNoGorner и FlNoCyclNoGorner)

Берется формула из справочника и программируется без всяких оптимизационных «премудростей».

2.2. Схема Горнера (FlCyclGorner и FlNoCyclGorner, FixCyclGorner и FixNoCyclGorner)

a[0] + a[1]\*x + a[2]\*x2 + a[3]\*x3 + … a[n]\*xn = ((…(a[n]\*x+a[n-1])\*x + a[n-2])\*x + … + a[1])\*x + a[0]

Бесцикловое вычисление предполагает непосредственную запись формулы из правой части в виде арифметического выражения. Здесь возможны два варианта: обращение к элементам массива коэффициентов и явное вписывание констант в выражение.

Цикловое вычисление схемы Горнера строится на основе тела цикла: s = s\*x+a[i];

Приложение 3. Таблично-алгоритмическая реализация

На этапе анализа разложения функции в ряд вида *f*(*x*) = *a0 + a1\*x + a2\*x2 + a3\*x3 + …*

внимание сосредоточено на значениях коэффициентов *ai .* В случае, когда 0 <= x < 1 длина ряда, обеспечивающего погрешность не более MAX\_ERR, зависит от того, насколько интенсивно убывают коэффициенты ряда по мере увеличения номера *i*. Это связано с тем, что даже самые «худшие» значения аргумента *x*, дающие наибольшую ошибку при отбрасывании части ряда, имеют значение почти равное 1, что дает значение степеней аргумента близкое к 1.

Если максимальное значение *x* было бы очень маленьким, то члены ряда по мере увеличения *i* убывали бы достаточно интенсивно благодаря возведению в степень. Например, если max(x) = 2-10, то значение x2 , было бы не больше 2-20. При значениях MAX\_ERR, близких к 2-20 , может стать возможным отбросить все члены ряда со степенями больше или даже равными 2.

Идея таблично-алгоритмического метода заключается в том, что весь диапазон значений аргумента *x* разбивается на много коротких поддиапазонов, для каждого из которых строится свой степенной ряд, у которого аргументом является отклонение от точки начала поддиапазона. Это можно сделать, например, на основе ряда Тейлора:

***f(x) = f(v)+ (x-v)\*f `(v) /1! + (x-v)2 \* f ``(v) /2! + …+ (x-v)n \* f (n)(x) /n! + …***

Разность ***(x-v)*** как раз и является отклонением от точки ***x = v*,** в которой начинается поддиапазон. Набор значений производных для любого заранее заданного значения *v* является величиной постоянной, что позволяет вычислить коэффициенты заранее и хранить в таблице. Индекс значения набора коэффициента при этом зависит от значения *v*.

Для получения 2m групп коэффициентов, где m – разрядность старшей части, можно использовать формулу разложения в ряд Тейлора:

при x = v+h

***f(x) = f(v)+ h\*f `(v) /1! + h2 \* f ``(v) /2! + …+ hn \* f (n)(v) /n! + …***

Выражение остаточного члена:

*Rn = (hn + 1) \*f (n+1)(a)\*(a+g\*h) /(n+1)!, где 0<g <1.*

Формулы для некоторых производных:

*(xn) ` = n\*xn-1*

*f `(a[0]+a[1]\*x+a[2]\*x2+ … + a[n]\*xn) =*

*a[1] + 2\*a[2]\*x + 3\*a[3]\*x2 + 4\*a[4]\*x3 + … + n\*a[n]\*xn-1*

*f ``( a[0]+a[1]\*x+a[2]\*x2+ … + a[n]\*xn) =*

*2\*a[2] + 2\*3\*a[3]\*x + 3\*4\*a[4]\*x2 + 4\*5\*a[5]\*x3 + … + n\*(n-1)\*a[n]\*xn-2*

*f ```( a[0]+a[1]\*x+a[2]\*x2+ … + a[n]\*xn) =*

*2\*3\*a[3] + 2\*3\*4\*a[4]\*x + 4\*5\*a[4]\*x3 + … + n\*(n-1)\*a[n]\*xn-2*

Впрочем, производная берется в точке (например, a = .101010101010 для приведенного выше примера), поэтому можно вычислить ее численным методом через ∆y/∆x, выбирая ∆x достаточно малым, чтобы не нарушить ограничения точности вычислений.