1. **Определение численных методов. Классификация численных методов**
2. **Элементы теории погрешностей. Виды и источники погрешностей**
3. **Абсолютная и относительная погрешности**
4. **Сходимость и устойчивость численных методов**
5. **Классификация численных методов решения нелинейных уравнений**
6. **Общая постановка задачи решения нелинейного уравнения, локализация корней уравнения**
7. **Уточнение корней методом половинного деления, алгоритм реализации метода**
8. **Программная реализация метода половинного деления, пример**
9. **Уточнение корней методом Ньютона, алгоритм реализации метода**
10. **Программная реализация метода Ньютона, пример**
11. **Уточнение корней методом хорд, алгоритм реализации метода**
12. **Программная реализация метода хорд, пример**
13. **Уточнение корней методом простых итераций, алгоритм реализации метода**
14. **Аппроксимация и интерполяция данных. Практическое применение. Математические определения**
15. **Аппроксимация по методу наименьших квадратов. Суть метода**
16. **Вывод системы нормальных уравнений для аппроксимации линейной функцией**
17. **Вывод системы нормальных уравнений для аппроксимации полиномом второго порядка**
18. **Интерполяция полиномом Лагранжа, вывод формулы Лагранжа для линейной, квадратичной и кубической интерполяции**
19. **Вывод формулы Лагранжа для глобальной интерполяции**
20. **Программная реализация интерполяции полиномом Лагранжа**
21. **Тригонометрическая интерполяция. Постановка задачи. Вид тригонометрического полинома**
22. **Формулы для расчета коэффициентов тригонометрического полинома и интерполирующей функции**
23. **Постановка задачи вычисления определенного интеграла, геометрический смысл и аналитическое вычисление определенного интеграла**
24. **Численные методы вычисления определенного интеграла: общий обзор, условия применения**
25. **Метод прямоугольников и трапеций, вывод формул, графическая интерпретация, примеры**
26. **Метод Симпсона, вывод формулы, графическая интерпретация, примеры**
27. **Погрешности численного интегрирования. Правило Рунге**
28. **Основные определения и постановка задачи решения ОДУ. Классификация ОДУ, примеры**
29. **Классификация и общий обзор численных методов решения ОДУ**
30. **Метод Эйлера, суть метода, графическая интерпретация, примеры**
31. **Метод Эйлера-Коши, суть метода, графическая интерпретация,** **примеры**
32. **Метод Рунге-Кутты 2 порядка, суть метода, графическая интерпретация, примеры**
33. **Метод Рунге-Кутты 4 порядка, суть метода, графическая интерпретация, примеры**
34. **Метод Рунге-Кутты 4 порядка для решения систем ОДУ**
35. **Метод Рунге-Кутты 4 порядка для решения ОДУ второго порядка**
36. **Постановка краевой задачи решения ОДУ, метод стрельбы для решения линейного дифференциального уравнения**
37. **Обзор многошаговых методов решения ОДУ. Алгоритм реализации метода Адамса**
38. **Постановка задачи поиска экстремума унимодальной функции**
39. **Классификация и общий обзор методов одномерной оптимизации**
40. **Метод золотого сечения: алгоритм метода, графическая интерпретация, пример**
41. **Метод Фибоначчи: алгоритм метода, графическая интерпретация, пример**
42. **Метод дихотомии: алгоритм метода, графическая интерпретация, пример**
43. **Метод Пауэлла: алгоритм метода, графическая интерпретация, пример**
44. **Метод средней точки, алгоритм метода**
45. **Определение численных методов. Классификация численных методов**

численные методы – методы в которых решение сложных математических задач сводится к последовательному выполнению большого числа простых арифметических операций.

+ дают хотя бы какое-то решение

Можно оценить ошибку

-решение всегда приближенное

**Классификация численных методов**

Можно привести следующую классификацию численных методов:

♣ методы решения уравнений;

♣ методы решения систем уравнений;

♣ методы вычисления интегралов;

♣ методы аппроксимации и интерполяции;

♣ методы решения дифференциальных уравнений и систем;

♣ методы оптимизации и т.д.

**Классификация численных методов**

Для алгебраических и трансцендентных уравнений пригодны одни и те же методы уточнения приближенных значений действительных корней:

• метод половинного деления (метод дихотомии);

• метод простых итераций;

• метод Ньютона (метод касательных);

• модифицированный метод Ньютона (метод секущих);

• метод хорд и др.

1. **Элементы теории погрешностей. Виды и источники погрешностей**

При решении любой практической задачи необходимо всегда указывать требуемую точность результата.

В связи с этим необходимо уметь:

• 1) зная заданную точность исходных данных, оценивать точность результата (прямая задача теории погрешностей);

• 2) зная требуемую точность результата, выбирать необходимую точность исходных данных (обратная задача теории погрешностей).

**При численном решении задач приходится оперировать двумя видами чисел – точными и приближенными.**

• К точным числам относятся числа, которые дают истинное значение исследуемой величины.

• К приближенным относятся числа, близкие к истинному значению, причем степень близости и определяется погрешностью вычислений.

**Источники погрешностей**

1) погрешность математической модели;

2) погрешность исходных данных (неустранимая погрешность);

3) погрешность численного метода;

4) вычислительная погрешность

**Погрешность математической модели** возникает из-за стремления обеспечить сравнительную простоту ее технической реализации и доступности исследования.

**Погрешность численного метода (погрешность аппроксимации**), связана, например, с заменой интеграла суммой, с усечением рядов при вычислении функций, с интерполированием табличных значений функциональных зависимостей и т.п.

**Вычислительная погрешность** возникает из-за округления чисел, промежуточных и окончательных результатов счета.

1. **Абсолютная и относительная погрешности**

1. **Абсолютная погрешность** — это разность между точным значением числа х и его приближенным значением а: x = | х – а |.

2. **Относительная погрешность** δх = Δx / |а|.

Относительную погрешность часто выражают в процентах: δх = Δx / |а| 100% . •

х – точное неизвестное значение некоторой величины определяется по формуле х=а±x.

1. **Сходимость и устойчивость численных методов**

При анализе вычислительного процесса одним из важнейших критериев является сходимость численного метода.

Он означает близость полученного численного решения к истинному.

**Последовательность** сходится к точному решению х, если при неограниченном возрастании числа итераций n предел этой последовательности существует и равен = x.

**Устойчивость метода**

• Пусть в результате решения задачи по исходному значению x находится значение искомой величины y.

• Пусть исходная величина имеет абсолютную погрешность dx, а решение имеет погрешность dy.

• Задача называется устойчивой по исходному параметру x, если малое приращение исходной величины dx приводит к малому приращению искомой величины dy.

• Другими словами, малые погрешности в исходных величинах приводят к малым погрешностям в результатах расчетов.

Отсутствие устойчивости означает, что даже незначительные погрешности в исходных данных приводят к большим погрешностям в решении или вовсе к неверному результату.

1. **Классификация численных методов решения нелинейных уравнений**

Для алгебраических и трансцендентных уравнений пригодны одни и те же методы уточнения приближенных значений действительных корней:

*метод половинного деления* (*метод дихотомии*);

*метод простых итераций* ;

*метод Ньютона* *(метод касательных)* ;

*модифицированный метод Ньютона* ( *метод секущих* );

*метод хорд* и др.

Для алгебраических и трансцендентных уравнений пригодны одни и те же методы уточнения приближенных значений действительных корней:

*метод половинного деления* (*метод дихотомии*);

*метод простых итераций* ;

*метод Ньютона* *(метод касательных)* ;

*модифицированный метод Ньютона* ( *метод секущих* );

*метод хорд* и др.

1. **Общая постановка задачи, локализация корней уравнения**

Метод половинного деления

Постановка задачи и алгоритм решения

• Дано нелинейное уравнение: f(x)=0 •

Найти корень уравнения, принадлежащий интервалу \*a,b+, с заданной точностью ε. • Для уточнения корня методом половинного деления последовательно осуществляем следующие операции: **(+ sследующий вопрос)**

1. **Уточнение корней методом половинного деления, алгоритм реализации метода**

Метод половинного деления

1. Делим интервал пополам: •

t=(a+b)/2

1. В качестве нового интервала изоляции принимаем ту половину интервала, на концах которого функция имеет разные знаки

Для этого:

• a) Вычисляем значение функции f(x) в точках a и t.

• b) Проверяем: если f(a)f(t) < 0, то корень находится в левой половине интервала \*a,b+. Тогда отбрасываем правую половину интервала и делаем переприсвоение b=t.

c) Если f(a)f(t) < 0 не выполняется, то корень находится в правой половине интервала \*a,b+. Тогда отбрасываем левую половину и делаем переприсвоение a=t. В обоих случаях мы получим новый интервал \*a,b+ в 2 раза меньший предыдущего.

Процесс, начиная с пункта 1, циклически повторяем до тех пор, пока длина интервала \*a,b+ не станет равной либо меньшей заданной точности.

1. **Программная реализация метода половинного деления, пример**
2. Описать функцию левой части уравнения f(x)
3. . Задать a, b, eps – границы интервала изоляции и точности вычислений
4. •Организовать цикл реализации метода •
5. Вывести полученные результаты

from math import\*

def f(x): return x\*\*2\*sin(x)

a=12

b=13

eps=0.001

while (b-a)>eps :

t=(a+b)/2.0

if f(t)\*f(a)<0:

b=t

else

a=t

print(t)

1. **Уточнение корней методом Ньютона, алгоритм реализации** **метода**

* Выберем начальную точку x0=b (конец интервала изоляции).
* Находим значение функции в этой точке и проводим к ней касательную, пересечение которой с осью Х дает нам первое приближение корня x1.
* В результате итерационный процесс схождения к корню реализуется рекуррентной формулой
* Процесс поиска продолжаем до тех пор, пока не выполнится условие:

1. **Программная реализация метода Ньютона, пример**

1.Описать функцию левой части уравнения f(x)=0 и функции первой производной 2. Задать x1, x2, eps –первое и второе приближения корня и точности вычислений

3. Организовать цикл реализации метода

4 Вывести полученные результаты

x0 = (a + b) / 2

xn = F(x0)

xn1 = xn - F(xn) / F1(xn)

while abs(xn1 - xn) > math.pow(10, -6):

xn = xn1

xn1 = xn - F(xn) / F1(xn)

return xn1

1. **Уточнение корней методом хорд, алгоритм реализации метода**

Входные данные: f(x), f’’(x), a, b, ε.

Если f(a)·f ”(a)>0, то c=a, иначе если f(b)·f ’’(b)>0, то c=b.

Если f(a)·f’’(a)<0, то x1=a, иначе если f(b)·f ’’(b)<0, то x1=b

dx=f(x1)·(x1-c)/(f(x1)-f(c)).

x1=x1-dx

Если |dx|>ε, то идти к 3.

1. **Программная реализация метода хорд, пример**

a =4

b = 3

eps = 0.00001

while abs(b - a) > eps:

a = b - (b - a) \* f(b) / (f(b) - f(a))

b = a + (a + b) \* f(a) / (f(a) - f(b))

1. **Аппроксимация и интерполяция данных. Практическое применение. Математические определения**

**Практическое применение**

Из всех способов задания зависимостей наиболее удобным является аналитический способ задания зависимости в виде функции n=f(Мкр.), P=f(t), y=f(t). В результате проведения эксперимента **табличную функцию.**

Можно ли проинтегрировать или продифференцировать табличную функцию?

При такой постановке задачи моделирования нужно заменить табличную функцию аналитической. Для этой цели используются методы ***аппроксимации и интерполяции*.**

**Математические определения**

**Аппроксимация** –это замена исходной функции f(x) функцией φ(x)так, чтобы отклонение f(x) от φ(x)в заданной области было наименьшим. Функция φ(x)называется аппроксимирующей.

Если исходная функция f(x) задана таблично (дискретным набором точек), то аппроксимация называется ***дискретной***

Если исходная функция f(x) задана аналитически (на отрезке), то аппроксимация называется ***непрерывной или интегральной***

***Интерполяция*** –это замена исходной функции f(x) функцией φ(x)так, чтобы φ(x)точно проходила через точки исходной функции f(x). Интерполяция еще называется ***точечной аппроксимацией***

Точки исходной функции f(x)называются ***узлами интерполяции***

Экстраполяцией называетсяаппроксимация вне заданной области определения исходной функции, т.е **x < x0 и x > xn**

Найдя интерполяционную функцию, мы можем вычислить ее значения между узлами интерполяции, а также определить значение функции за пределами заданного интервала (**провести экстраполяцию**).

1. **Классификация методов аппроксимации и интерполяции, общий обзор методов**

Если интерполяция проводится только для отдельных участков отрезка [а, b] (области определения f(х)), т.е. для mинтерполяционных узлов, где m < n, то интерполяцию называют **локальной**.

Если интерполяционный многочлен ищется для всего интервала области определения x, т.е. для [x0, xn], то интерполяция называется **глобальной.**

Простейшими видами локальной интерполяции является **линейная и квадратичная**

При линейной интерполяции точки заданной функции соединяются линейными отрезками, а при квадратичной –отрезками парабол.

Такая интерполяция еще называется **параболической**.

**Типовые виды глобальной интерполяции**

* Интерполяционный многочлен Лагранжа
* Интерполяционный многочлен Ньютона
* Сплайны

Сплайн представляет собой математическую модель гибкого тонкого стержня из упругого материала, закрепленного в двух соседних узлах интерполяции с заданными углами наклона αи βтак, чтобы потенциальная энергия стержня была минимальна

Сплайн –функция, которая вместе с несколькими производными непрерывна на всем заданном отрезке [a,b], а на каждом частичном отрезке [ xi,xi+1] в отдельности является некоторым алгебраическим многочленом.

Наиболее известными методами аппроксимации являются метод наименьших квадратов, метод многочленов Чебышева, рядов Тейлор и т.д.

**При решении задач аппроксимации часто используются функции регрессии**

**Регрессия**–представление совокупности данных некоторой функцией **f(x)**.

Задачей регрессии является вычисление параметров функции **f(x)**таким образом, чтобы функция приближала последовательность исходных точек с наименьшей погрешностью.

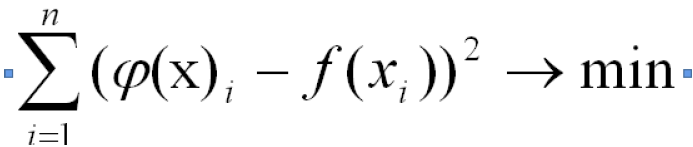
При этом функция **f(x)**называется **уравнением регрессии**.

При регрессии не требуется чтобы функция проходила через все заданные точки, что особенно важно при аппроксимации данных, заведомо содержащих ошибки.

1. **Аппроксимация по методу наименьших квадратов. Суть метода**

Метод наименьших квадратов (МНК), идея которого принадлежит А.Лежандру, а теоретическое обоснование К.Гауссу. В соответствии с этим методом, оценки параметров ai определяют из условия минимума суммы квадратов отклонений измеренных значений yi от соответствующей ординаты рассмотренной кривой

Критерий метода наименьших квадратов

****Основной мерой отклонения аппроксимирующей функции φ(x)от исходной функции f(x) при аппроксимации является величина, равная сумме квадратов разностей между значениями аппроксимирующей и исходной функций

**Алгоритм**

•1. Ввести два одномерных массива xи yперечислением элементов

•2. Вычислить суммы (какие?)

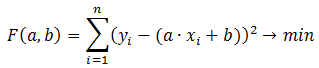
•3. Вычислить aи bпо формулам

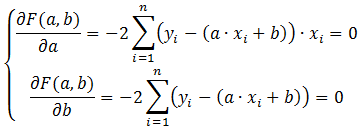
•4. Создать одномерный массив аргумента аппроксимирующей функции, например, x1, содержащий числа в пределах изменения x, но их количество должно быть гораздо больше (50-100 чисел)

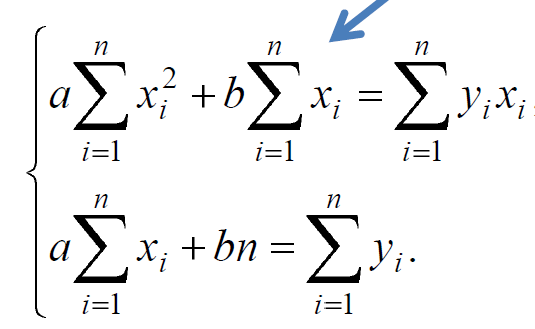
•5. Рассчитать значения аппроксимирующей функцииy1 для каждого значения x1

•6. Построить график исходной функции в виде точек, не соединенные отрезками прямых и помеченные маркером и график аппроксимирующей функции

1. **Вывод системы нормальных уравнений для аппроксимации линейной функцией**

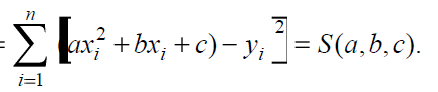
y = ax+b.  
Аппроксимация заключается в отыскании коэффициентов a и b уравнения таких, чтобы все экспериментальные точки лежали наиболее близко к аппроксимирующей прямой.

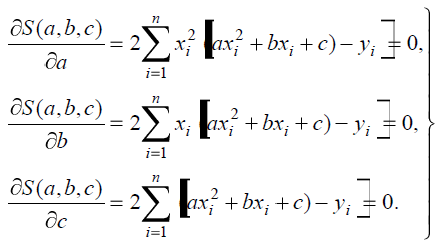


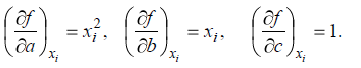


1. **Вывод системы нормальных уравнений для аппроксимации полиномом второго порядка**

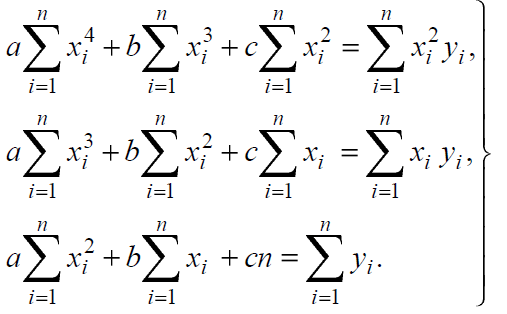
*,*  ,







Аналогично тому, как это сделано для аппроксимации прямой, получим систему нормальных уравнений:



1. **Интерполяция полиномом Лагранжа, вывод формулы Лагранжа для линейной, квадратичной и кубической интерполяции**

многочлен Лагранжа:

**Линейная интерполяция**•При **n=1**формула Лагранжа имеет вид:

**Квадатическая интерполяция** •При **n=2**формула Лагранжа имеет вид:

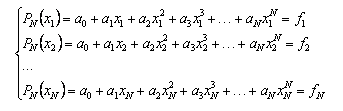
**Кубическая интерполяция** •При **n=3**формула Лагранжа имеет вид:

1. **Вывод формулы Лагранжа для глобальной интерполяции**

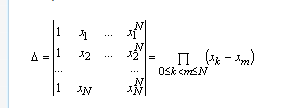
В случае глобальной интерполяции отыскивается единый полином на всем интервале [a, b], т.е. строится полином, который используется для интерполяции функции f(x) на всем интервале изменения аргумента x.

Нужно искать интерполирующую функцию в виде полинома (многочлена) m–ой степени .

Уравнения системы представляют собой условия интерполяции в при каждом :



После этого нужно найти определитель системы



Определитель отличен от нуля, если (т.е. все узлы интерполяции различные).

Если функция f(x) интерполируется на отрезке [a,b] с помощью единого многочлена для всего отрезка, то такую интерполяцию называют глобальной. В случае локальной интерполяции на каждом интервале [] строится интерполяционный отдельный интерполяционный полином невысокой степени.

1. **Программная реализация интерполяции полиномом Лагранжа**

|  |
| --- |
| import matplotlib.pyplot as plt  import numpy as np  from math import\*  x=np.array([-1,0,1])  y=np.array([-0.8, 1.6, 2.3])  n = len(x)  xt=0.5  Yt=0  for j in range(0, n):  L=1  for i in range (0,n):  if i!=j :  L=L\*(xt-x[i])/(x[j]-x[i])  Yt=Yt+y[j]\*L  print(Yt) |

1. **Тригонометрическая интерполяция. Постановка задачи. Вид тригонометрического полинома**

**Тригонометрическим полиномом** от переменной ***x*** называется выражение вида(1)

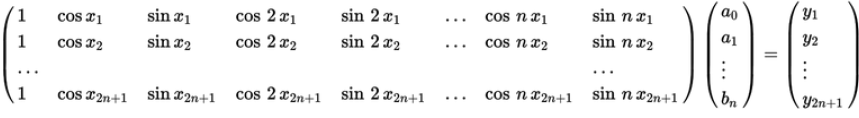
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | X1 | X2 | … | x(2n+1) |
| Y | Y1 | Y2 | … | y(2n+1) |

Постановка задачи. Построить тригонометрический полином порядка n, принимающий значения по следующей таблице

Интерполирующий полином имеет вид

В узловых точках

Для получения коэффициентов интерполирующего полинома нужно решить систему 2n+1 линейных уравнений

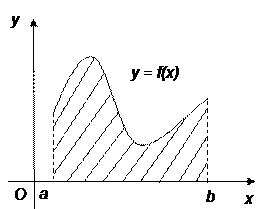


1. **Формулы для расчета коэффициентов тригонометрического полинома и интерполирующей функции**

Если в качестве узлов интерполяции взять равноотстоящие

то коэффициенты интерполяционного тригонометрического полинома находятся по формулам:

1. **Постановка задачи вычисления определенного интеграла, геометрический смысл и аналитическое вычисление определенного интеграла**

Пусть требуется вычислить определенный интеграл

, где f(x) – непрерывная на отрезке [a; b] функция.

С **геометрической точки зрения** интеграл (1) при f(x) > 0 равен площади криволинейной трапеции, ограниченной кривой y = f(x), осью Ox и прямыми x = a, x = b. Другими словами, равен площади заштрихованной фигуры на рисунке

Геометрический смысл определенного интеграла. Рисунок --------->

Вычислить определенный интеграл (1) можно с помощью аналитической формулы Ньютона-Лейбница (2):

(2), где F(x) – первообразная функция для заданной функции f(x).

Однако во многих случаях не удается найти никакой аналитической формулы в виду невозможности определения F(x).

Таким интегралом, например, является . (3)

В математическом анализе доказывается, что для данной подынтегральной функции нельзя выразить первообразную F(x) через элементарные функции. С другой стороны площадь криволинейной трапеции, задаваемой интегралом (3) существует. Значит, должно существовать и значение интеграла, которое, однако, мы не можем найти точно.

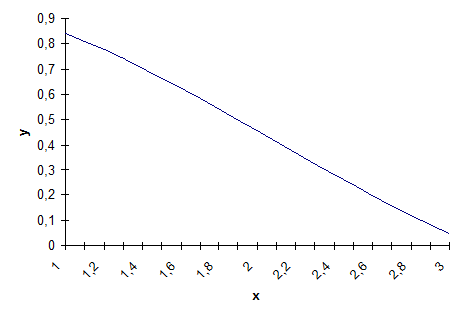


Рис. 2. График функции на отрезке [1; 3].

В таких случаях приходится применять методы численного интегрирования.

Основной принцип построения всех приближенных формул численного интегрирования состоит в том, что интервал интегрирования разбивается на множество меньших отрезков, внутри которых подынтегральная кривая y = f(x) заменяется с некоторой степенью точности более простыми функциями, интегралы от которых можно вычислить. С геометрической точки зрения выполняется следующее: искомая площадь криволинейной трапеции приближенно заменяется суммой площадей элементарных геометрических фигур.

1. **Численные методы вычисления определенного интеграла: общий обзор, условия применения**

К методам вычисления определенного интеграла относятся:

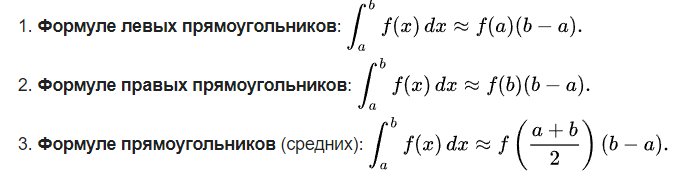
- метод прямоугольников;

- метод трапеций;

- метод Симпсона.

**Метод прямоугольников:**

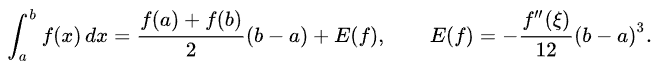
Если рассмотреть график подынтегральной функции, то метод будет заключаться в приближённом вычислении площади под графиком суммированием площадей конечного числа прямоугольников, ширина которых будет определяться расстоянием между соответствующими соседними узлами интегрирования, а высота — значением подынтегральной функции в этих узлах.



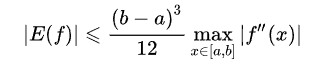
**Метод трапеций:**

Метод трапеций — метод численного интегрирования функции одной переменной, заключающийся в замене на каждом элементарном отрезке подынтегральной функции на многочлен первой степени, то есть линейную функцию. Площадь под графиком функции аппроксимируется прямоугольными трапециями. Алгебраический порядок точности равен 1.

Если отрезок [a,b] является элементарным и не подвергается дальнейшему разбиению, значение интеграла можно найти по формуле



Это простое применение формулы для площади трапеции — произведение полусуммы оснований, которыми в данном случае являются значения функции в крайних точках отрезка, на высоту (длину отрезка интегрирования). Погрешность аппроксимации для элементарного отрезка можно оценить через максимум второй производной



**Метод Симпсона(парабол):**

На каждом интервале подынтегральная функция приближается квадратичной параболой , проходящей через точки . Отсюда и название метода - метод парабол.



Это делается для того, чтобы в качестве приближенного значения определенного интеграла взять , который мы можем вычислить по формуле Ньютона-Лейбница. В этом и заключается **суть метода парабол.**



Геометрически это выглядит так:



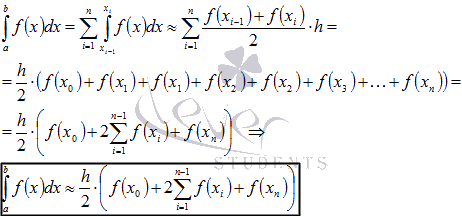
1. **Метод прямоугольников и трапеций, вывод формул, графическая интерпретация, примеры**

**Суть метода трапеций.**

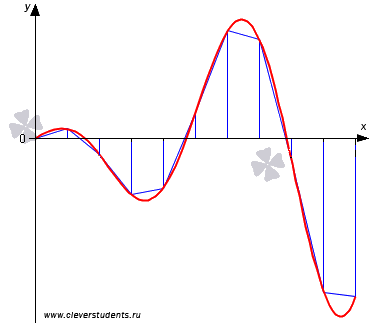
Состоит в представлении определенного формула интеграла в виде суммы интегралов вида формула на каждом элементарном отрезке и в последующей приближенной замене .

формула

**Формула метода трапеций.**

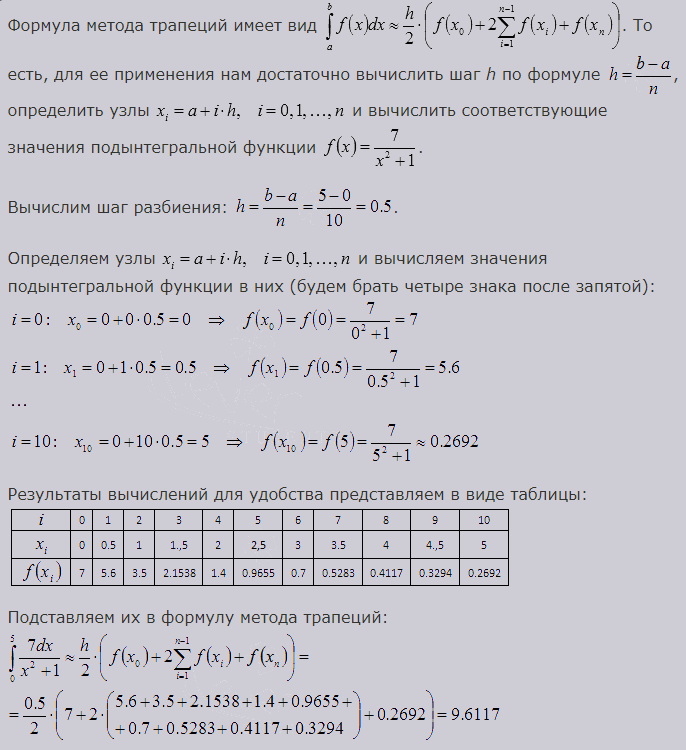


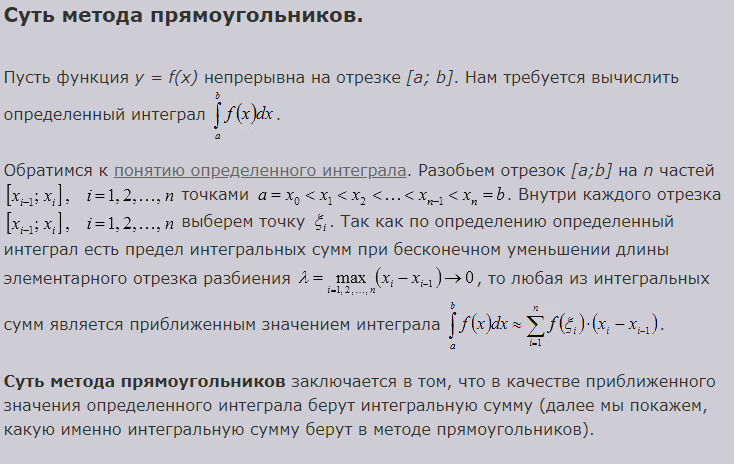
**Графическая иллюстрация метода трапеций.**

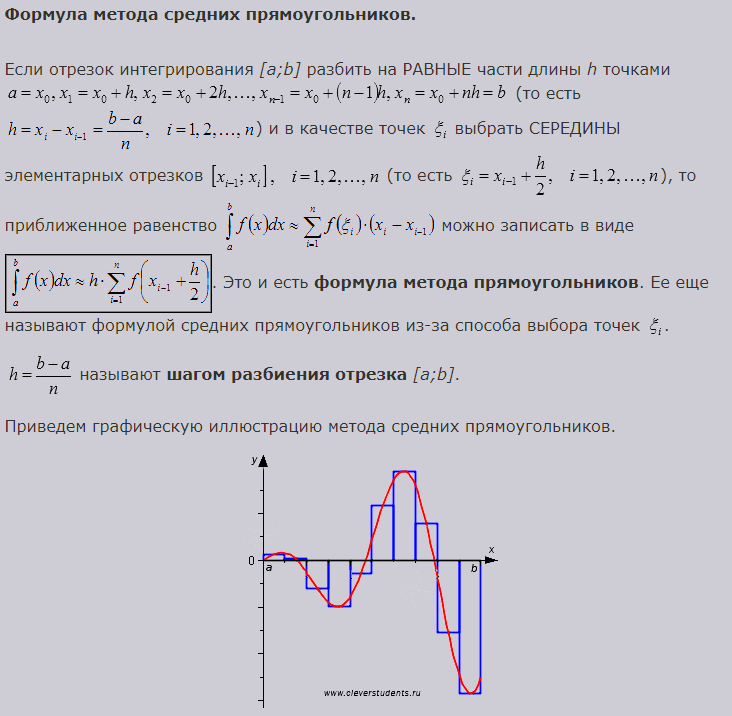


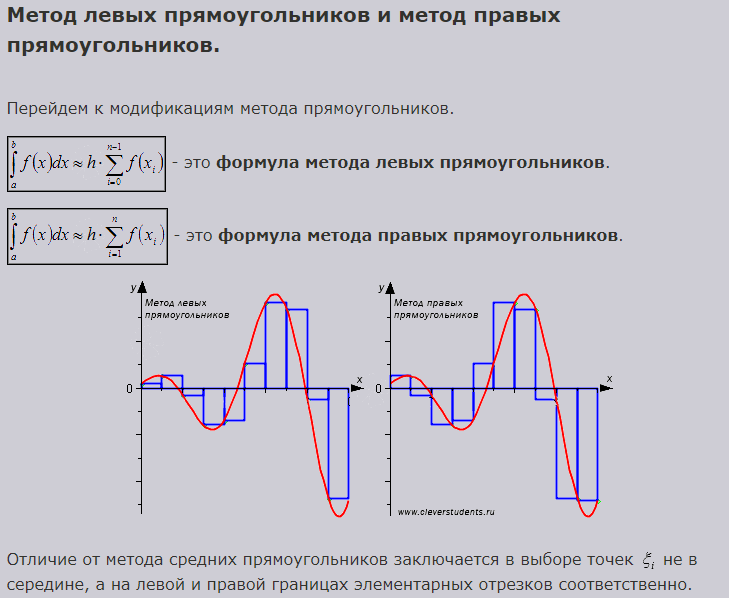
**Пример.(хз скорее всего не диктуй)**

Вычислить определенный интеграл формула методом трапеций для n = 10.





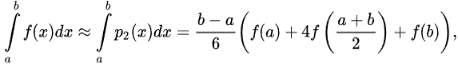




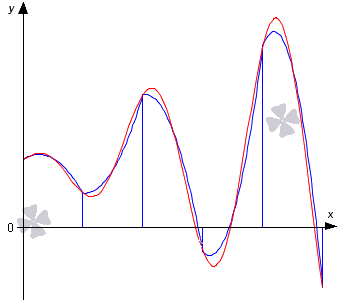
1. **Метод Симпсона, вывод формулы, графическая интерпретация, примеры**

Пусть функция *y = f(x)* непрерывна на отрезке *[a; b]* и нам требуется вычислить определенный интеграл .Разобьем отрезок *[a;b]* на *n* элементарных отрезков  длины  точками . Пусть точки  являются серединами отрезков  соответственно. В этом случае все "узлы" определяются из равенства .



Формула Симпсона 

Красной линией изображен график функции *y=f(x)*, синей линией показано приближение графика функции *y=f(x)* квадратичными параболами на каждом элементарном отрезке разбиения.



Пример



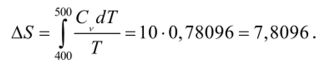
Принимаем количество молей *п=* 1, значение теплоемкости при v=const:



Разобьем интервал интегрирования на 10 равных частей. Шаг интегрирования будет равен Л = (500-400) /10=10. Результаты вычислений подынтегральной функции поместим в таблицу.



• по формуле Симпсона

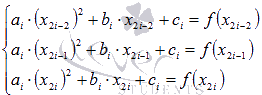


Вывод формулы:

Для получения формулы метода парабол (Симпсона) вычислить .



Пусть  Покажем, что через точки  проходит только одна квадратичная парабола . Так как  - точки параболы, то справедливо каждое из уравнений системы

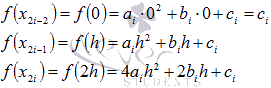


Записанная система уравнений есть система линейных алгебраических уравнений относительно неизвестных переменных .

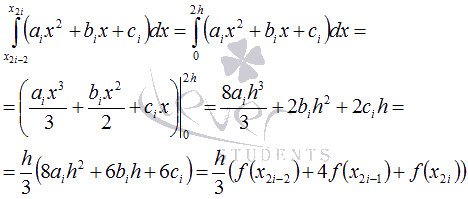


Перейдем к нахождению интеграла

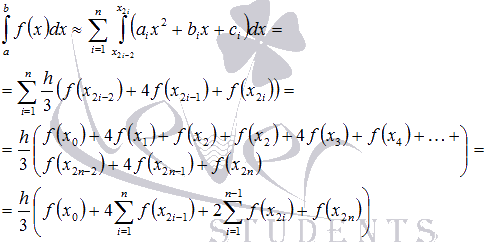
 .



Используем эти равенства, чтобы осуществить последний переход в следующей цепочке равенств:

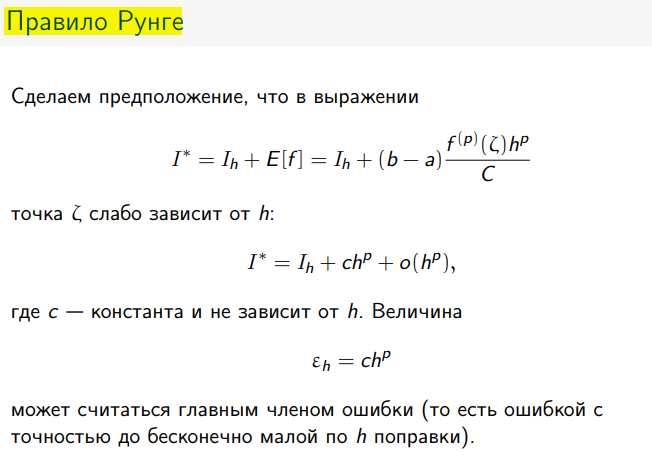


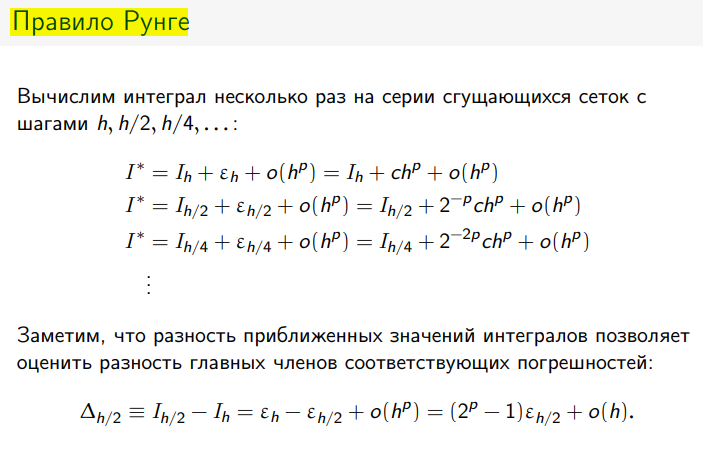
Таким образом, можно получить формулу метода Симпсона:

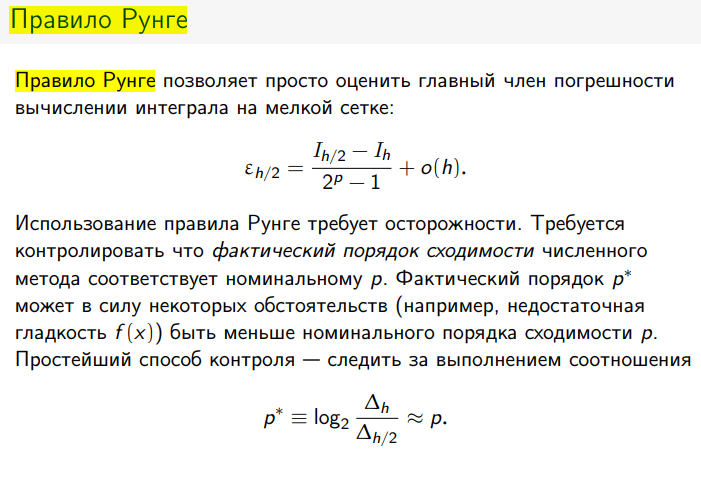


1. **Погрешности численного интегрирования. Правило Рунге**









1. **Основные определения и постановка задачи решения ОДУ. Классификация ОДУ, примеры**

*Обыкновенным дифференциальным уравнением* (ОДУ) n-го порядка называется следующее уравнение, которое содержит одну или несколько производных от искомой функции y(x):

, здесь *y(n)* обозначает производную порядка n некоторой функции y(x), x – это независимая переменная.



*Решением обыкновенного дифференциального уравнения* называется такая функция y(x), которая при любых х удовлетворяет этому уравнению в определенном конечном или бесконечном интервале.

*Общее решение ОДУ* *n*-го порядка содержит n произвольных констант *C1*, *C2*, …, *Cn*



***Задача Коши (начальная задача):*** Необходимо найти такое *частное решение* дифференциального уравнения, которое удовлетворяет определенным *начальными условиям, заданным в одной точке*:



то есть, задано определенное значение независимой переменной *(х0)*, и значение функции и всех ее производных вплоть до порядка *(n-1)* в этой точке.

***Краевая задача***. В этом случае известны значения функции и (или) ее производных в более чем одной точке, например, в начальный и конечный момент времени, и необходимо найти частное решение дифференциального уравнения между этими точками. Сами дополнительные условия в этом случае называются *краевыми* (*граничными*) условиями.

К основным видам дифференциальных уравнение, рассматриваемых в данном пособии относятся.

I. ОДУ первого порядка

1) С разделяющимися переменными

2) Однородные ДУ

3) Линейные ДУ

4) Уравнения Бернулли

5) Уравнения в полных дифференциалах.

II. Дифференциальные уравнения высших порядков, допускающие понижение порядка

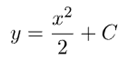
III. Линейные дифференциальные уравнения второго порядка с постоянными коэффициентами.

1) Однородные.

2) Неоднородные.

Пример 

|  |  |
| --- | --- |
|  |  |

Ответ : 

1. **Классификация и общий обзор численных методов решения ОДУ**

Для решения дифференциальных уравнений необходимо знать значение зависимой переменной и ее производных при некоторых значениях независимой переменной. Если дополнительные условия задаются при одном значении неизвестной, т.е. независимой переменной, то такая задача называется задачей Коши. Если начальные условия задаются при двух или более значениях независимой переменной, то задача называется краевой. При решении дифференциальных уравнений различных видов, функция, значения которой требуется определить, вычисляется в виде таблицы.

Классификация численных методов для решения дифр. ур. типов*.*

|  |  |  |
| --- | --- | --- |
| Численные методы решения дифр. ур. | | |
| Обыкновенные дифр. ур. | | Дифр. ур. в частных производных |
| Задача Коши | Краевая задача |  |

Задача Коши:

– одношаговые: методы Эйлера, методы Рунге- Кутта;

– многошаговые: метод Майна, Метод Адамса.

Краевая задача:

– метод сведения краевой задачи к задаче Коши;

–метод конечных разностей.

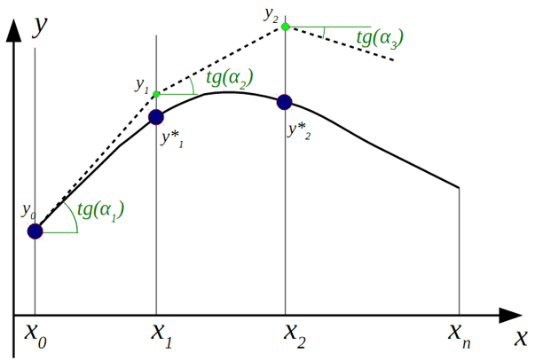
При решении задачи Коши должны быть заданы дифр. ур. порядка n или система дифр. ур. первого порядка из n уравнений и n дополнительных условий для ее решения. Дополнительные условия должны быть заданы при одном и том же значении независимой переменной. При решении краевой задачи должны быть заданы ур. n-ого порядка или система из n уравнений и n дополнительных условий при двух или более значениях независимой переменной. При решении задачи Коши искомая функция определяется дискретно в виде таблицы с некоторым заданным шагом Δ. При определении каждого очередного значения можно использовать информацию об одной предыдущей точке. В этом случае методы называют одношаговым, либо можно использовать информацию о нескольких предыдущих точках – многошаговые методы.

1. **Метод Эйлера, суть метода, графическая интерпретация, примеры**

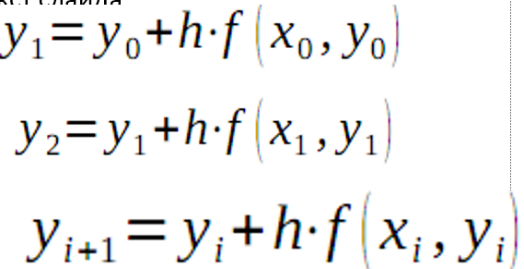
Одним из самых первых и однозначно простейшим сеточным методом решения задачи Коши является метод Эйлера.

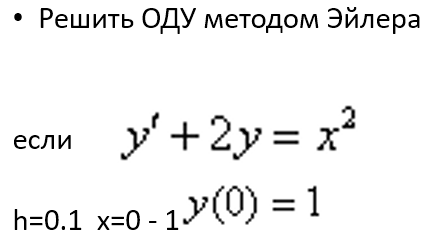
Рассмотрим несколько подходов к его построению.

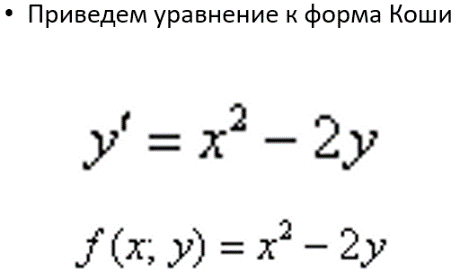
Данный подход основан на том свойстве производной, что она равнатангенсу угла наклона касательной в данной точке.

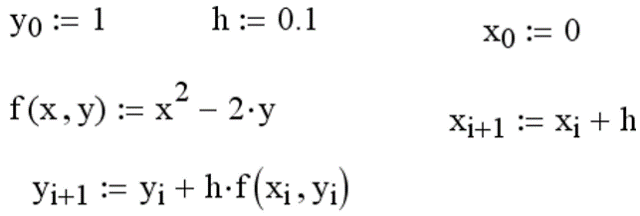










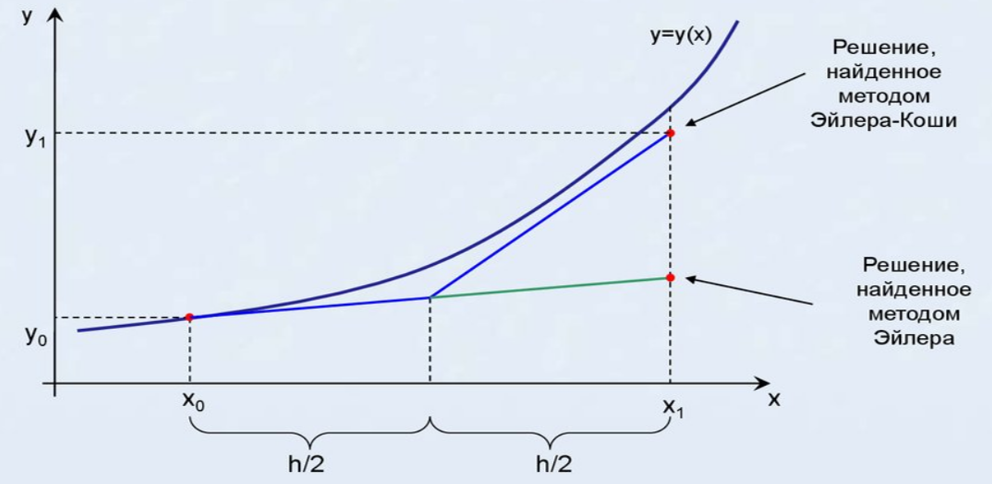


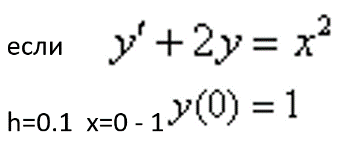
1. **Метод Эйлера-Коши, суть метода, графическая интерпретация, примеры**

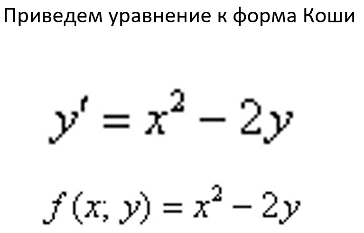
**Метод является модификацией метода Эйлера. Первая половина шага совершается с тангенсом угла наклона касательной в предыдущей точке, а вторая с тангенсом угла наклона к последующей точке.**

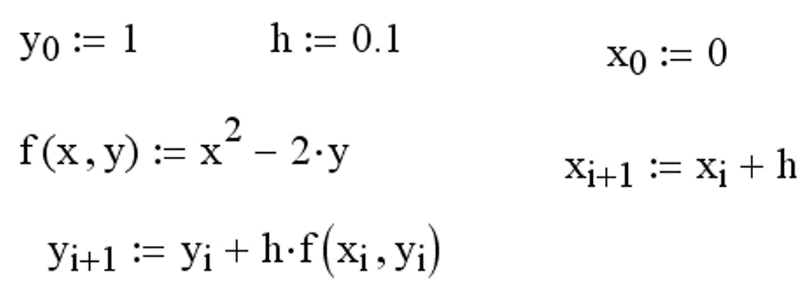
**Формула решения ОДУ методом Эйлера-Коши**

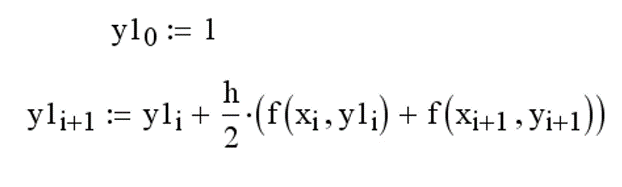
**Метод является неявным, одношаговым, устойчивым, второго порядка точности (более точным, чем метод Эйлера)**

******

******

******

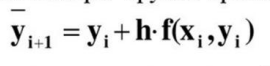
******

******

1. **Метод Рунге-Кутты 2 порядка, суть метода, графическая интерпретация, примеры**

По методу Рунге-Кутты 2-го порядка вычисление значения искомой функции в точке проводится в два этапа.

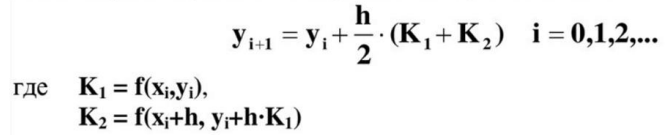
Снача вычисляют методом Эйлера “грубое приближение”:



Затем вычисляют значение производной в точке  и окончательно полагают:

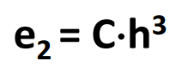


то есть устредняют значения производных в начальной точке и в точке “грубого приближения”. Оконочательно запишем рекуррентную формулу метода Рунге-Кутты 2го порядка в след виде:



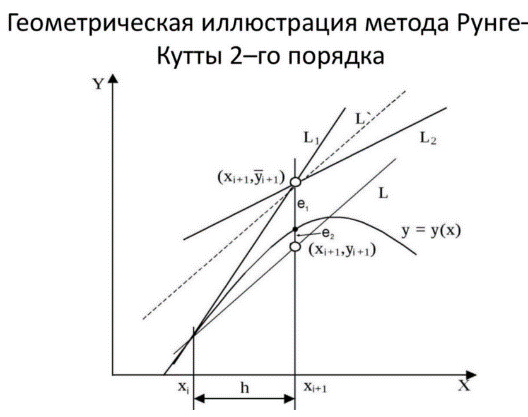
Локальная погрешность метода Рунге-Кутты 2-го порядка:

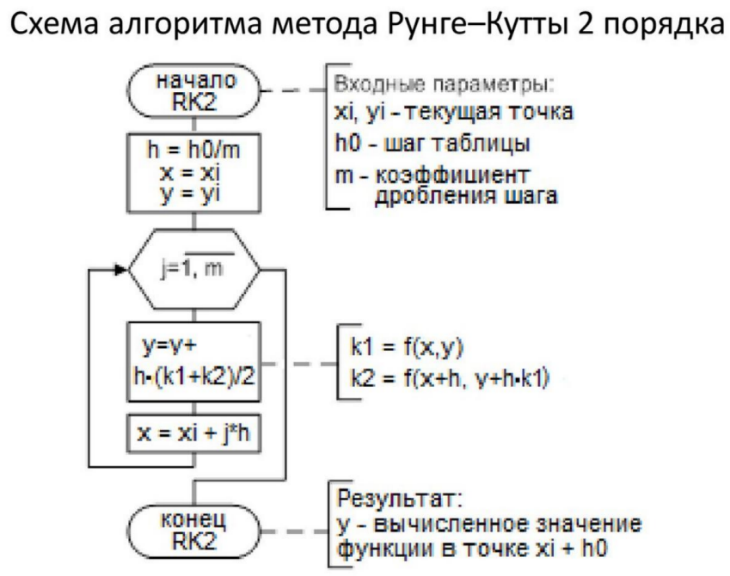
Локальная погрешность равна



С – некоторая постоянная.

Погрешность пропорциональна кубу шага интегрирования, при уменьшении шага в 2 раза погрешность уменьшается в 8 раз.





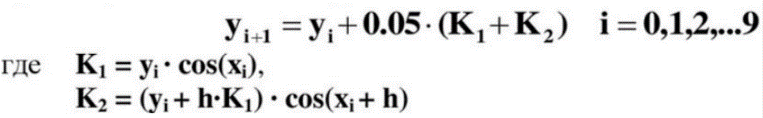
Решить ОДУ



Если х0 = 0, у0=1.

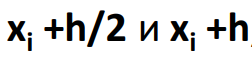
Найти числ решение на отрезке [0;1] c шагом h =0.1

Решение ищется по рекуррентной формуле:



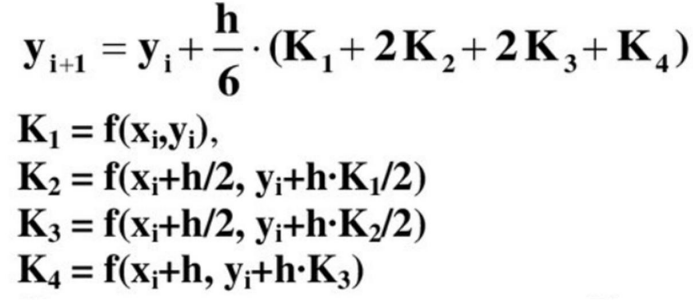
1. **Метод Рунге-Кутты 4 порядка, суть метода, графическая интерпретация, примеры**

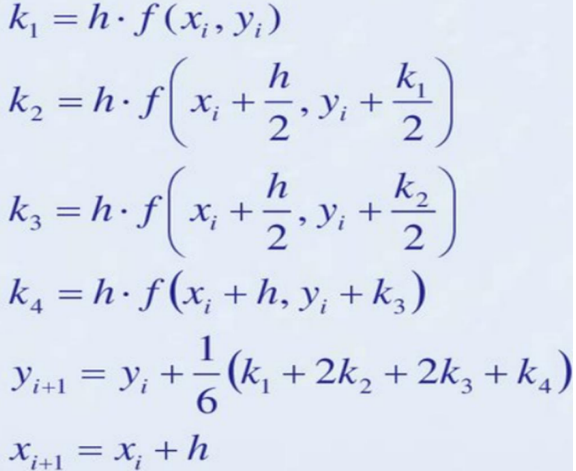
По этому методу вычисление результирующей функции в точке xi+1 проводится в несколько этапов:

- сначала прогнозируется вычисление искомой функции в точках 

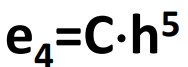
- затем производится коррекция с усреднением результатов прогноза.

Рекуррентная формула для этого метода имеет вид:





Локальная погрешность метода Рунге-Кутта 4-го порядка:

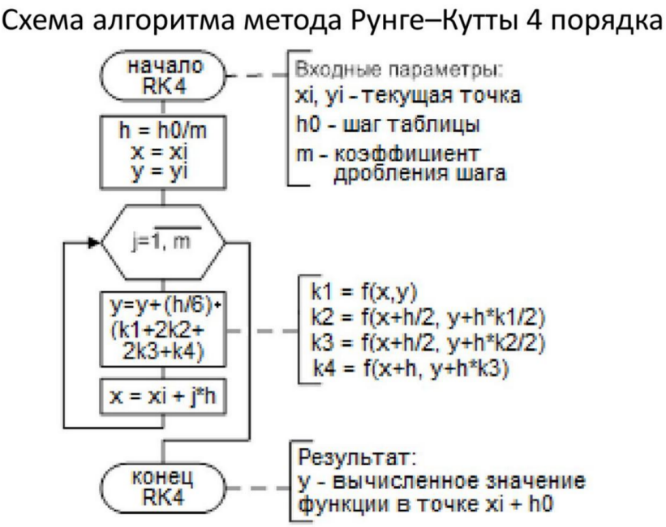


C- некоторая постоянная

Она пропорциональна пятой степени шага интегрирования, при уменьшении шага в два раза локальная погрешность уменьшается в 32 раза

Метод Рунге-Кутта 4 порядка является самым

точным из рассмотренных методов



1. **Метод Рунге-Кутты 4 порядка для решения систем ОДУ**

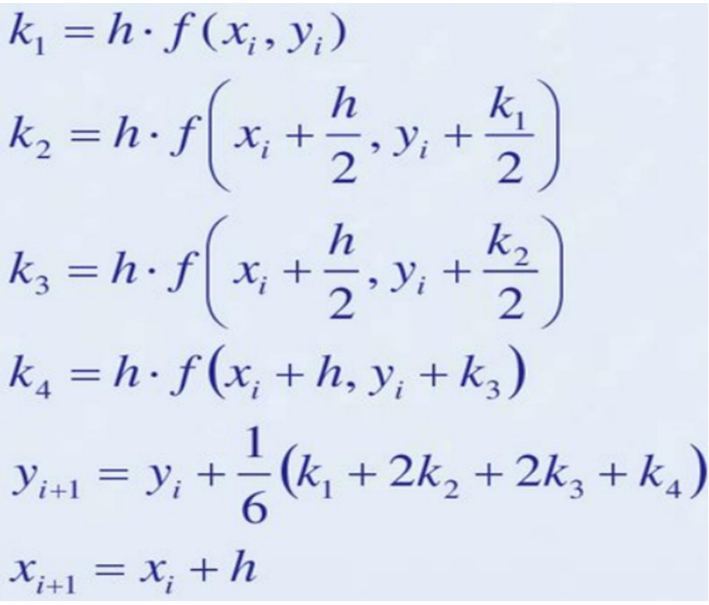
По этому методу вычисление результирующей функции в точке xi+1 проводится в несколько этапов:

--- сначала прогнозируется вычисление искомой функции в точках

xi +h/2 и xi +h;

--- затем производится коррекция с усреднением результатов прогноза.

Рекуррентная формула для этого метода имеет вид:

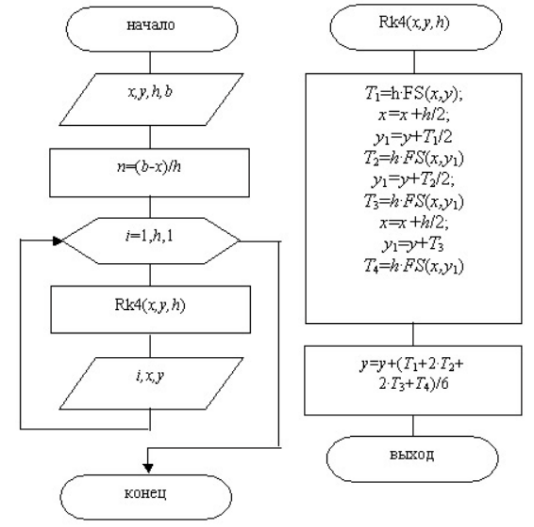


Локальная погрешность метода Рунге-Кутта 4-го порядка: **е4=С\*h^5**

C- некоторая постоянная. Она пропорциональна пятой степени шага интегрирования, при уменьшении шага в два раза локальная погрешность уменьшается в 32 раза

Метод Рунге-Кутта 4 порядка является самым точным из рассмотренных методов

1. **Метод Рунге-Кутты 4 порядка для решения ОДУ второго порядка**



(x,y) -при вводе начальная точка, далее текущие значения табличной функции,

h -шаг интегрирования дифференциального уравнения,

b -конец интервала интегрирования.

1. **Постановка краевой задачи решения ОДУ, метод стрельбы для решения линейного дифференциального уравнения**

**Краевая задача**

• Определение:

В краевой задаче требуется найти решение обыкновенного дифференциального уравнения с дополнительными условиями, заданными при нескольких различных значениях независимой переменной.

• Например, для обыкновенного дифференциального уравнения второго порядка

y " = f(x,y,y') эти условия задаются в двух разных точках x=a и x=b

y(a) = y0 , x = a; y(b) = y1 , x = b,

• Поскольку эти точки определяют границы области a < x < b, в которой обычно и отыскивается решение, поставленные в них дополнительные условия называют граничными или краевыми.

Метод стрельбы основан на приведении краевой задачи к задаче Коши путем замены дополнительных условий.

Если дана краевая задача в формулировке вида: **y " = f(x,y,y' ),**

**y(a) = y0 , x = a;**

**y(b) = y1 , x = b**

то в методе стрельбы она заменяется задачей Коши для того же уравнения но с начальными условиями

**y(a) = y0 , x = a;**

**y ' (a) = tg α, α : y(b,α) = y1 ,**

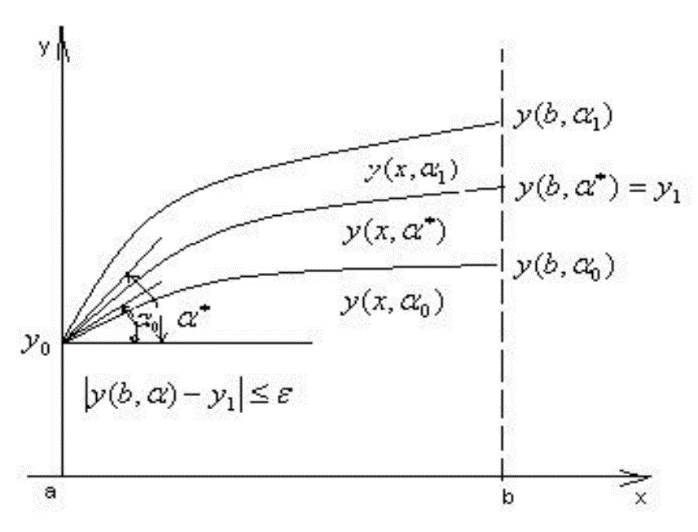
• В этой задаче интегральная кривая y(x,α) зависит не только от переменной x, но и от параметра α, который называется углом стрельбы.

Скажи тут короче что нужно попасть в определенную точку

Сначала мы кидаем два пробный броска образуя корридор(интервал)

Потом этот интервал делят на 2 и находят благоприятствующий

После этого снова делается выстрел и так пока не дойдем до точки



1. **Обзор многошаговых методов решения ОДУ. Алгоритм реализации метода Адамса**

Многошаговые методы используют для вычисления значения узла сетки

(xi+1, yi+1)

значения нескольких предыдущих узлов:

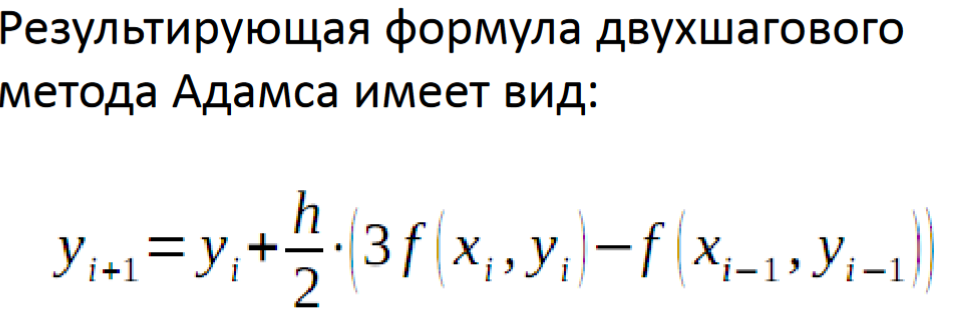
(xi, yi), (xi-1, yi-1), ... (xi-k+1, yi-k+1),

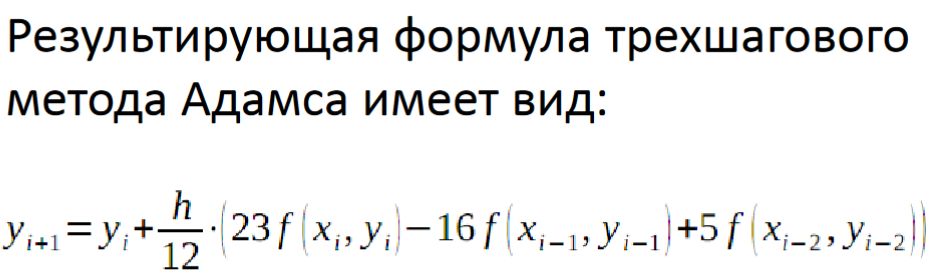
где k – количество шагов (шаговость) метода.

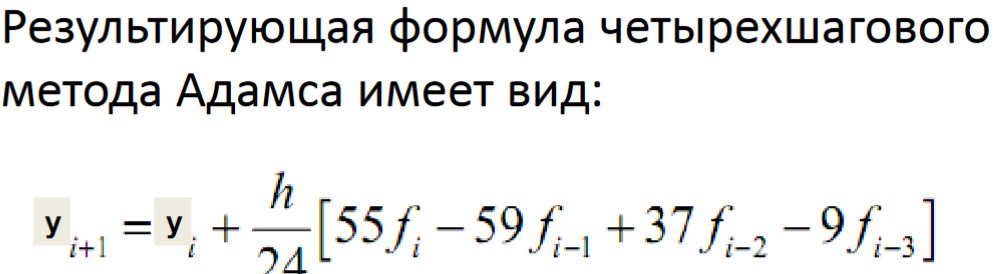
В зависимости от значения k методы делятся на:

* двухшаговые;
* трёхшаговые;
* четырёхшаговые.

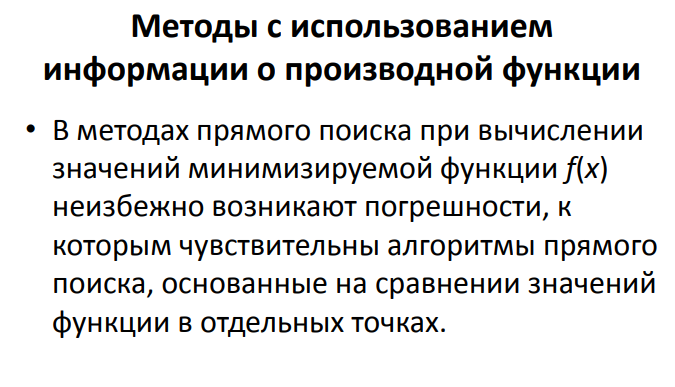
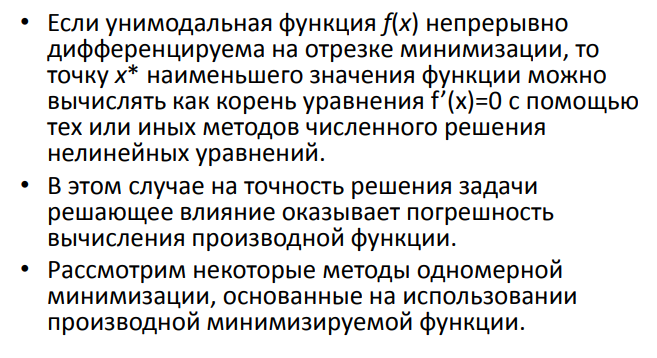
Метод Адамса второго порядка. По сравнению с методом Рунге-Кутта той же точности, Адамса экономичнее, так как на каждом шаге необходимо считать один раз вычисление правой части.







1. **Постановка задачи поиска экстремума унимодальной функции**

1. **Классификация и общий обзор методов одномерной оптимизации**

**Метод поиска экстремума с применением аппроксимации**

Основная идея метода: возможность аппроксимации гладкой функции полиномом достаточно высокого порядка и использование этого полинома для оценивания точки оптимума.

Качество этой оценки может быть повышено

двумя способами:

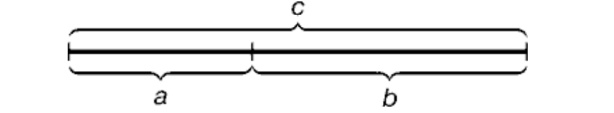
• увеличением степени полинома;

• уменьшением интервала аппроксимации.

**Методы с использованием информации о производной функции**

В методах прямого поиска при вычислении значений минимизируемой функции f(x) возникают погрешности

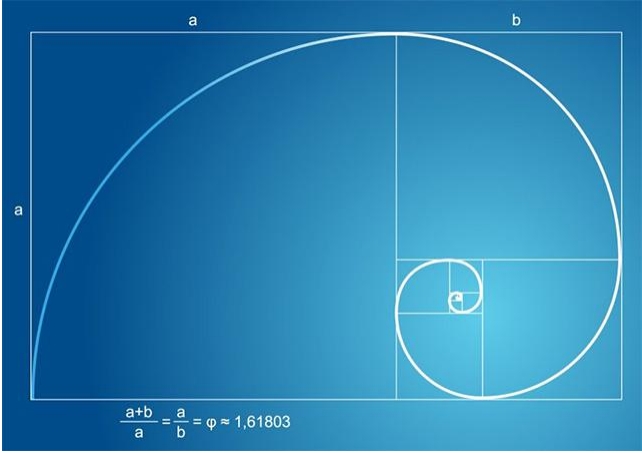
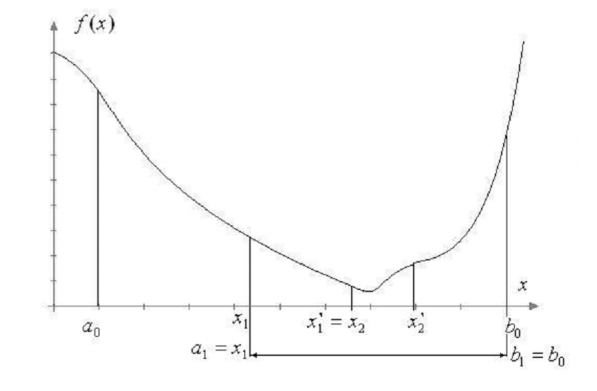
1. **Метод золотого сечения: алгоритм метода, графическая интерпретация, пример**



Золотое сечение - это такое пропорциональное деление отрезка на

неравные части, при котором весь отрезок так относится к большей части, как сама большая часть относится к меньшей; или другими словами, меньший отрезок так относится к большему, как больший ко всему.

a : b = b : с или с : b = b : а. Алгоритм метода

****

Шаг 1. На первой итерации находим две точки по следующим формулам:

и вычисляем значения функций и .

Шаг 2. Сокращаем интервал неопределенности

1) Если < , то = , =, = .

2) В противном случае, если > , то = , =, = .

Шаг 3. На последующих итерациях производим расчет только той точки и значение функции в ней, которые необходимо обновить:

в случае 1) вычисляем новое значение и

в случае 2) и

Шаг 4. Поиск прекращается при выполнении условия

**Пример кода(не диктуй)**

**from** math **import** pow, abs

**def** f(x):

    x = (a + b) / 2

    R = f(x) = pow(x, 2) + 2\*x

*# ввод значений*

a = float(input('Введите начало отрезка: '))

b = float(input('Введите конец отрезка: '))

eps = float(input('Введите точность: '))

*# Вычисляем значения функций f(x1), f(x2)*

x1 = a + 0.382 \* (b - a)

x2 = b - 0.382 \* (b - a)

f(x1) = pow(x1, 2) + 2\*x1

f(x2) = pow(x2, 2) + 2\*x2

**while** (True):

**if** (f(x1) < f(x2)):

        b = x2

**else**:

        a = x1

**if** (math.abs(b - a) < eps):

**return** x

**return** R

**print** (x, R)

**else**:

**if** b = x2:

                x2 = x1

                f(x2) = f(x1)

                x1 = a + 0.382 \* (b - a)

                f(x1) = pow(x1, 2) + 2\*x1

**if** a = x1:

                x1 = x2

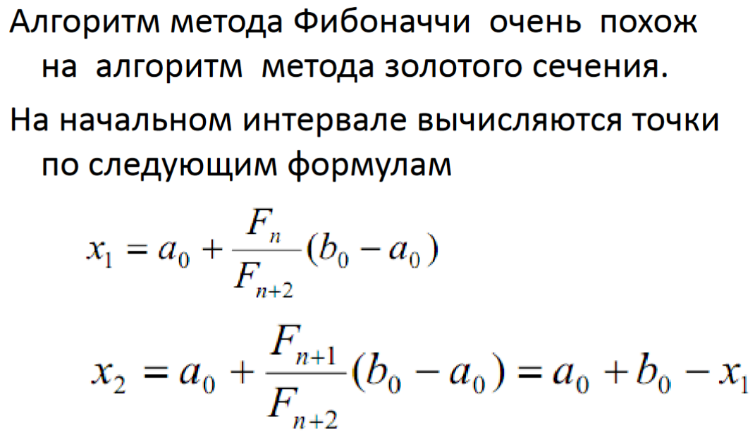
                f(x1) = f(x2)

                x2 = b - 0.382 \* (b - a)

                f(x2) = pow(x2, 2) + 2\*x

1. **Метод Фибоначчи: алгоритм метода, графическая интерпретация, пример**

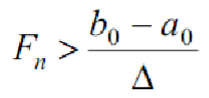
Ряд Фибоначчи имеет вид 1, 1, 2, 3, 5, 8, 13, 21…

****

Интервал неопределённости сокращается точно так же, как и в методе золотого сечения. На новой итерации вычисляется только одна новая точка и значение функции в ней.

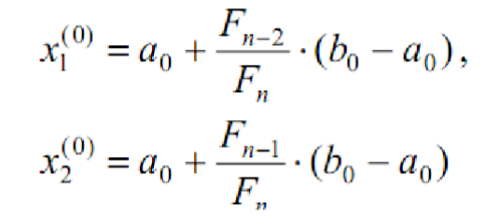
Алгоритм метода.

1) Выбирается начальный отрезок [a0, b0] и число вычислений n, таким образом, чтобы

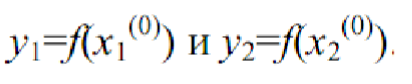


где > 0 – конечная длина отрезка.

2) Затем рассчитываются координаты двух точек.



И значение функции в этих точках



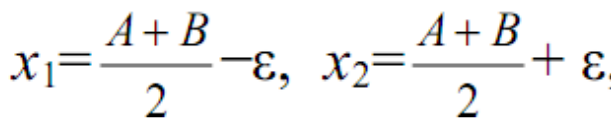
1. **Метод дихотомии: алгоритм метода, графическая интерпретация, пример**

Суть метода в следующем.

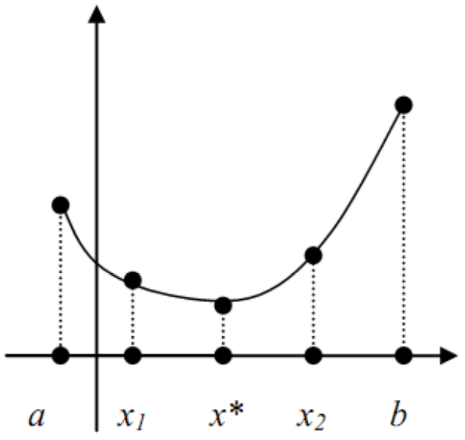
• Пусть функция F(х) задана на интервале A≤x≤B.

• Отрезок на каждом этапе делится пополам.

• За первые две поисковые точки принимаются:



где ε – величина, меньшая половины требуемой абсолютной погрешности решения.



Обобщенный алгоритм выглядит так:

1. Задать начальный интервал ;



1. Убедиться, что на концах функция имеет разный знак;
2. Повторять
   * выбрать внутри интервала точку ;



* + сравнить знак функции в точке  со знаком функции в одном из концов;



* + - если совпадает, то переместить этот конец интервала в точку ,



* + - иначе переместить в точку  другой конец интервала;



пока не будет достигнута нужная точность.

1. **Метод Пауэлла: алгоритм метода, графическая интерпретация, пример**

Оптимальное решение оценивается по формуле

**алгоритм**

1. Задаем ,
2. Вычисляем
3. Вычисляем

Если ,то

иначе

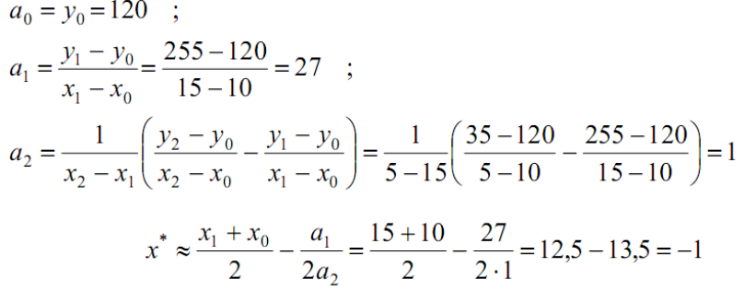
1. Вычисляем
2. Используя значения x0 , x1 , x2 и у0 , у1 , у2 вычислить x\*с помощью квадратичной аппроксимации.
3. Найти уmin = min(у0 , у1 , у2) и xmin , соответствующую уmin .
4. Проверить условие окончания поиска | xmin - x\*| ≤ ε ,

где ε - заданная точность поиска.

Если условие выполняется, закончить поиск; в противном случае перейти к следующему шагу.

1. Выбрать “наилучшую” точку (xmin или x\*) и две точки по обе стороны от нее и перейти к шагу 5). Если выбранная точка является “крайней”, то отбрасывается точка с наибольшим значением целевой функции.

**пример**

* Найти минимум функции

F(x) = x2 +2x

* eps=0.01
* x0=10
* dx=5

Пример

• Принимаем величину начального шага

• Δx = 5 для начальной точки x0 = 10.

• Вычисляем x1 = x0 + Δx =10+5 =15 ;

• у0 = f(10) = 120 и у1 = f(x1) = 255 .

• Поскольку у0 ≤ у1 , x2 = x0 - Δx =10 – 5 = 5 ; у2 = f(x2) = f(5) = 35.

• Используя значения x0 , x1 , x2 и у0 , у1 , у2 вычисляем x\*с помощью квадратичной аппроксимации

***(картинка)***

Проверяем условие окончания поиска |xmin-x\*|=|x2-x\*|=|5+1|> и продолжаем поиск

Отбрасываем точку с наибольшим значением целевой функции, т.е.

у1=ymax=255 при x1=15

Для оставшихся точек x0=10,x1=-1,x2=5 рассчитываем новые коэфф полинома a0=120, a1=11,a2=1 и оптимальное значение x\*=-1

Условие окончания поиска выполняется |xmin-x\*|=|x1-x\*|=|-1+1|> , поэтому поиск можно прекратить

• Минимум функции достигается в точке x\* ≈ = - 1.

• Количество итераций равно 2 при 4 вычислениях функции.

• Найти минимум функции

F(x) = x2 +15x

• x0=10 ∆x=5

1. **Метод средней точки, метод Ньютона, алгоритмы методов**

