**19**. Алгоритм Свенна (для нахождения начального интервала [a0; b0]):

1. Задать произвольно параметры: х0 – некоторую точку, h>0 – величину шага. Положить k = 0.
2. Вычислить значения в 3-х точках: х0, х0 – h, х0 + h.
3. Проверить условие окончания:

а) если f(х0 – h) ≥ f(х0) ≤ f(х0 + h), то начальный интервал неопределённости найден: [a0; b0] = [х0 – h; х0 + h];

б) если f(х0 – h) ≤ f(х0) ≥f(х0 + h), то функция не является унимодальной (рекомендуется задать другую точку х0)

в) если условия окончания не выполняются, то перейти к шагу 4.

4) Определить величину Δ:

а) если f(х0 – h) ≥ f(х0) ≥ f(х0 + h), то Δ = h; a0 = x0; x’=х0 + h; k=1

б) если f(х0 – h) ≤ f(х0) ≤ f(х0 + h), то Δ = -h; b0 = x0; x’=х0 - h; k=1

5) Найти точку xk+1= xk + 2kΔ

6) Проверить условие убывания функции:

а) если f(xk+1) < f(xk) и Δ = h, то a0 = xk;

если f(xk+1) < f(xk) и Δ = -h, то b0 = xk;

В обоих случаях положить k = k + 1 и перейти к шагу 5.

б) если f(xk+1) ≥ f(xk), процедура завершается.

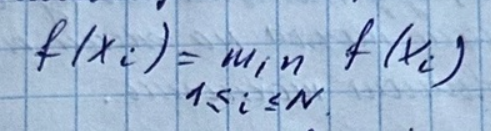
При Δ = h положить b0 = xk+1, а при Δ = -h положить a0 = xk+1. В результате [a0, b0] – искомый начальный интервал неопределённости.

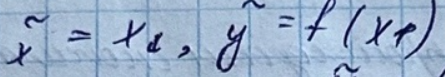
**20**. Метод перебора. Достоинства и недостатки. Алгоритм метода.

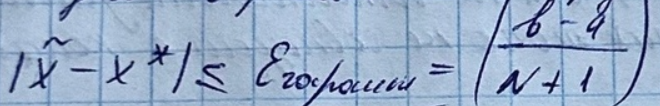
Метод перебора – простейший из прямых методов минимизации.

Алгоритм:

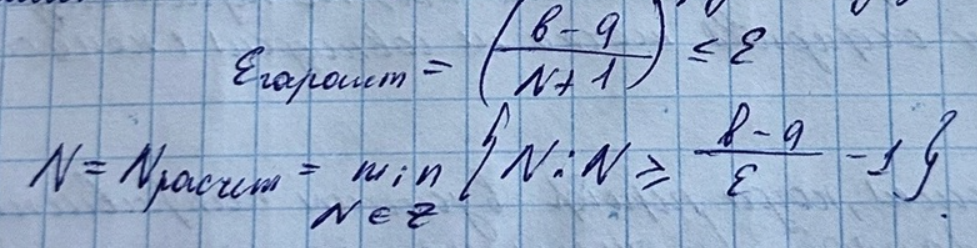
1. Разобьём отрезок [a, b] на N+1 равных частей точками деления: xi = a + i \* \*, i = 
2. Вычислим значение f(x) в точках xi. Найдём точку xi, для которой значение целевой функции минимально:



1. Далее положим . Точность найденного решения определяется по формуле:



Если необходимо найти приближённое решение с заданной точностью ε, то минимальное число экспериментов N для достижения этой точности определяется условием:



Достоинства:

* Простота

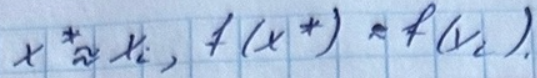
Недостатки:

* Чтобы точность определения точки минимума функции f(x) на отрезке [a, b] была приемлемая, необходимо большое число разбиений N отрезка, что приводит к увеличению объёма вычислений.
* Увеличение объёма вычислений приводят к увеличению суммарной ошибки - общей погрешности, следовательно, необходимо с большей точностью производить вычисления.
* Требуется вычислить значения функции f(x) в точках разбиения xi отрезка [a, b], сравнить их с вычисленными раннее значениями, выбрать наименьшее значение, при этом вычисление значений функций может быть трудоёмко и приближённо (остаточная погрешность).

**21.** Метод поразрядного поиска. Достоинства и недостатки. Алгоритм метода.

Метод поразрядного поиска – усовершенствованный метод перебора с целью уменьшения количества значений функции, которые необходимо находить в процессе минимизации.

Алгоритм:

1. Задать начальный интервал неопределённости L0 = [a0; b0], ε – точность и начальный шаг дискретизации Δ>ε.
2. Положить i = 1, x0 = a. Вычислить f(x0).
3. Определить точку xi = xi-1+Δ и значение функции f(xi).
4. Если f(xi) ≤ f(xi-1) и xi ≠ a, xi ≠ b, то положить i = i + 1и вернуться к шагу 3, иначе перейти к шагу 5.
5. Если Δ<ε, то перейти к шагу 7, иначе к шагу 6.
6. Задать новый шаг дискретизации (изменить направление и шаг поиска) Δ = -
7. Положить 

Достоинства:

* Простота
* Значительная область применимости

Недостатки:

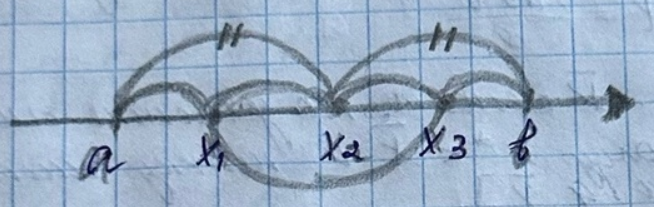
* Неэффективность

**22.** Метод деления отрезка пополам. Достоинства и недостатки. Алгоритм метода.

Метод деления отрезка пополам относится к последовательным методам и позволяет исключить на каждом шаге в точности половину текущего отрезка локализации [a; b].

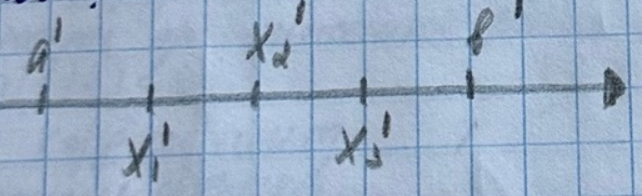
Алгоритм:

Для проведения экспериментов выбираются три точки x1, x2, x3 (x1 < x2 < x3), равномерно расположенные на отрезке локализации [a; b].



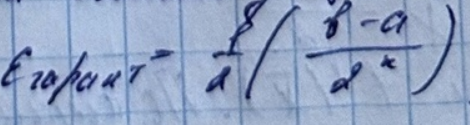
После обработки результатов экспериментов f(x1), f(x2), f(x3) получается новый отрезок локализации: [a, x2] или [x1, x2] или [x2, b].

Длина нового отрезка , т. е. равно в 2 раза меньше исходного.



Общее число экспериментов после k шагов: N = 2k + 1. N – нечётное число. После каждого шага длина отрезка локализации уменьшается ровно в 2 раза. Поэтому после k шагов метода длина последнего отрезка будет ровно в 2х раз меньше длины исходного отрезка, равной b – a.

Поскольку гарантированная точность решения будет равна половине длины последнего отрезка локализации, то гарантированная точность определения по формуле:



Достоинства:

* не нужно знать интервал, только начальное приближение
* применим для функций нескольких переменных
* быстрая сходимость

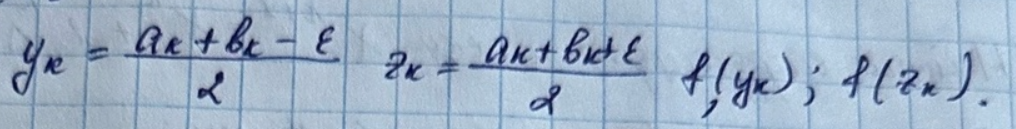
Недостатки:

* Может зацикливаться
* Ограниченность применения

**23.** Метод дихотомии. Достоинства и недостатки метода. Сходимость метода.

Метод дихотомии является процедурой последовательного поиска.

Алгоритм:

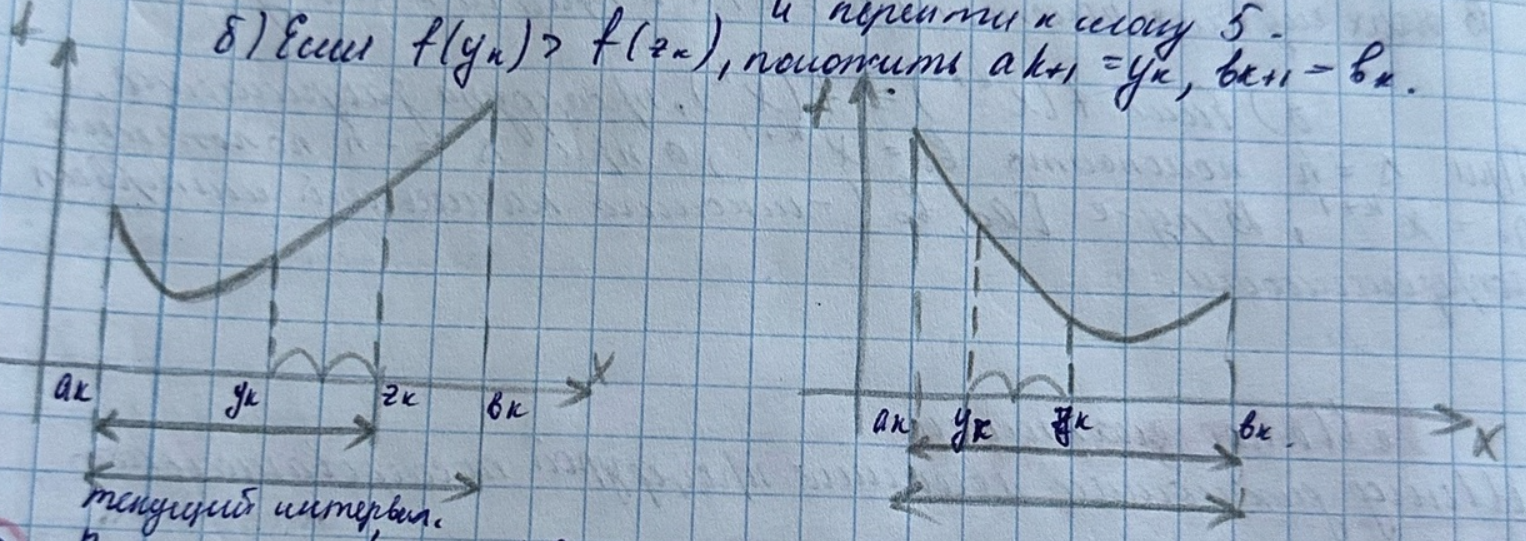
1. Задать начальный интервал неопределённости: L0= [a0; b0], ε>0 – малое число, I > 0 – точность (или N – количество вычислений функции)
2. Положить k=0
3. Вычислить 

Вычислить значение функции.

1. Сравнить f(yk) с f(zk).

а) Если f(yk) ≤ f(zk), положить ak+1 = ak, bk+1 = zk и перейти к шагу 5.

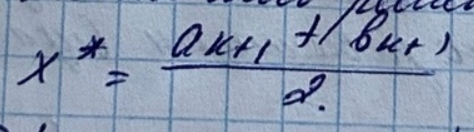
б) Если f(yk) > f(zk), положить ak+1 = yk, bk+1 = bk.



1. Вычисляем Lk+1 = | bk+1 - ak+1 | и проверяем условия окончания:

а) Если |Lk+1| ≤ 1 (или выполнено заданное N) процесс поиска завершается и x\* ∈ Lk+1 = [ak+1; bk+1].

В качестве приближенного решения берём середину последнего интервала:



б) Если |Lk+1| > 1, положить k = k+1 и перейти к шагу 3.

Достоинства:

* Простота
* Можно получить решение с заданной точностью (в пределах точности машинных вычислений)
* Основное достоинства метода - его скорость сходимости не зависит от вида функции f(x)

Недостатки:

* Нужно знать начальный интервал [a0; b0]
* На интервале [a0; b0] должно быть только одно решение
* Большее число шагов для достижения важной точности
* Только для функций одной переменной

Сходимость.

Для метода дихотомии характеристика относительного уменьшения начального интервала неопределенности находится по формуле: R(N) = 1/(2)^(N/2). N – количество вычислений функции.

**24.** Метод золотого сечения. Достоинства и недостатки метода. Сходимость метода.

Золотым сечением называется принцип деления отрезка на две части, при этом большая часть отрезка относится к меньшей части так же, как длина всего отрезка к большей части.

Алгоритм:

1. Задать начальный интервал неопределённости: L0= [a0; b0], I > 0 – точность (или N – количество вычислений функции)
2. Положить k=0
3. Вычислить y0 = a0 + 0,382(b0 - a0) и z0 = a0 + b0 – y0
4. Вычислить f(yk) и f(zk)
5. Сравнить f(yk) с f(zk):

а) Если f(yk) ≤ f(zk), положить ak+1 = ak, bk+1 = zk, yk+1 = ak+1 + bk+1 - yk,

zk+1 = yk

б) Если f(yk) > f(zk), положить ak+1 = yk, bk+1 = bk, yk+1 = zk, zk+1 = ak+1 + +bk+1 - zk

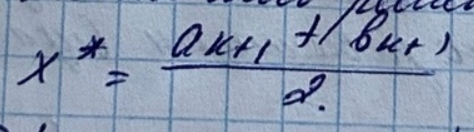
Перейти к шагу 6.

1. Вычислить Δ = | ak+1 - +bk+1| и проверить условие окончания:

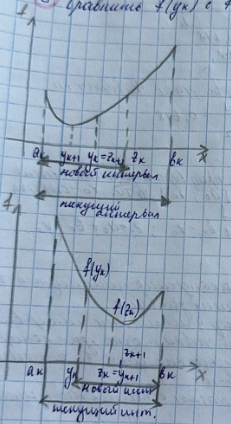
а) Если Δ ≤ 1 (или выполнено заданное N) процесс поиска завершается и

x\* ∈ Δ = [ak+1; bk+1].

В качестве приближенного решения берём середину последнего интервала:



б) Если Δ > 1, положить k = k+1 и перейти к шагу 4.



Достоинства:

* Гарантирует нахождение минимума в самых неблагоприятных условиях
* На каждом шагу уменьшения интервала вводится только одна новая точка, для которой требуется произвести только одно вычисление значения целевой функции
* Быстрее уменьшается область неопределённости

Недостатки:

* Нужно знать интервал [a; b]
* Только для функции одной переменной
* Неустойчивость относительно ошибок округления

Сходимость.

Для метода золотого сечения характеристика относительного уменьшения начального интервала неопределённости находится по формуле : R(N) = (0,618)N-1