

**Лекция 1. Методы численного
интегрирования. Методы
интегрирования. Алгоритмы
методов: прямоугольника, трапеций
и Симпсона.**

Определенный интеграл

$$\int_a^b f(x)dx$$

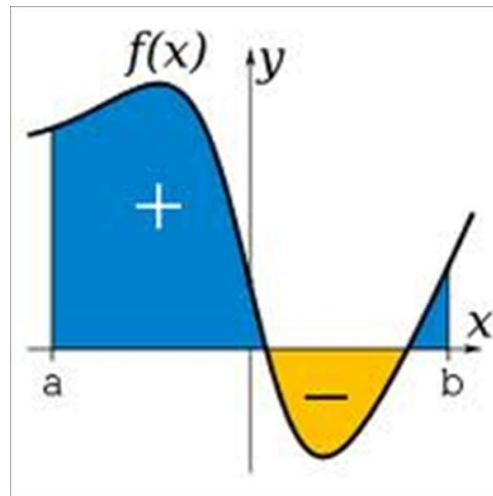
$f(x)$ – подынтегральная функция,
непрерывная на отрезке $[a,b]$

Формула Ньютона-Лейбница

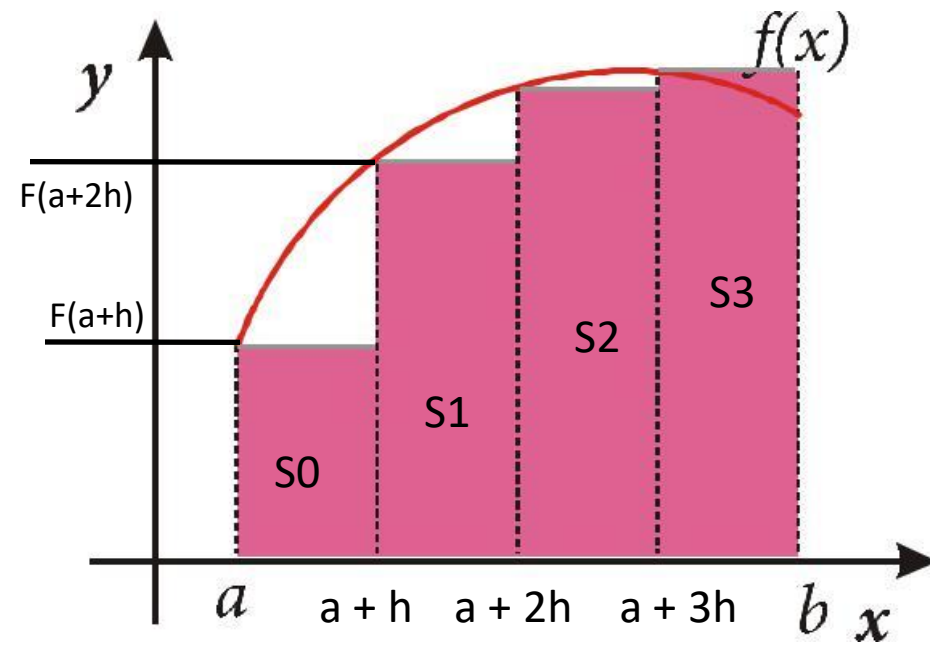
$$\int_a^b f(x)dx = f(b) - f(a)$$

Интеграл функции

- Интеграл функции – является площадью части графика функции (в пределах интегрирования), то есть площадью криволинейно трапеции
- Процесс нахождения интеграла называется интегрированием



Метод левых прямоугольников



Алгоритм:

1. Задать отрезок $[a;b]$
2. Делим отрезок на n частей и получаем шаг $h=(b-a)/n$
3. Площадь каждого получившегося прямоугольника: $S_i=h*f(a+i*h)$, где h – ширина, $f(a+i*h)$ – высота
4. Считаем сумму площадей $S=\sum S_i$
5. Увеличиваем число разбиений $n*=2$ и снова находим шаг и сумму площадей S_1
6. Повторяем пункты 2-5 пока $|S-S_1| \geq E$, где $E=1e-4$

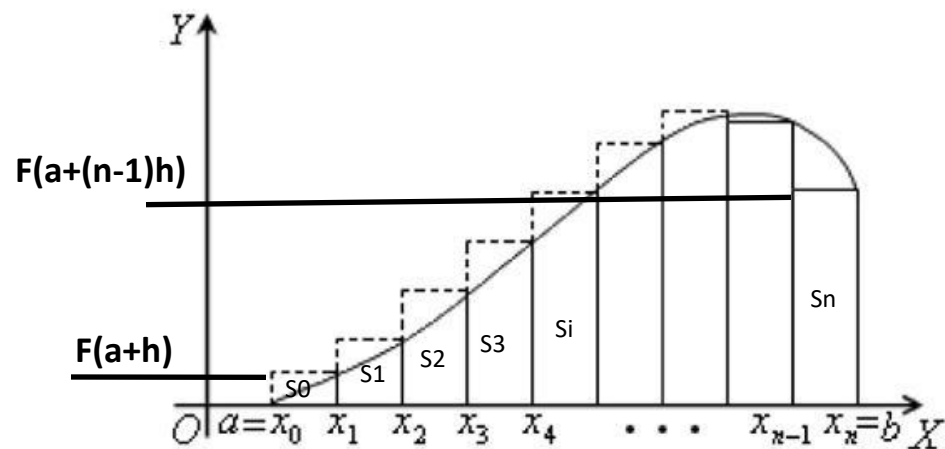
Метод правых и средних прямоугольников

Метод правых прямоугольников

$$S = h * f(a + (i+1) * h)$$

Ширина – h

Высота – значение функции справа
 $f(a + ih)$

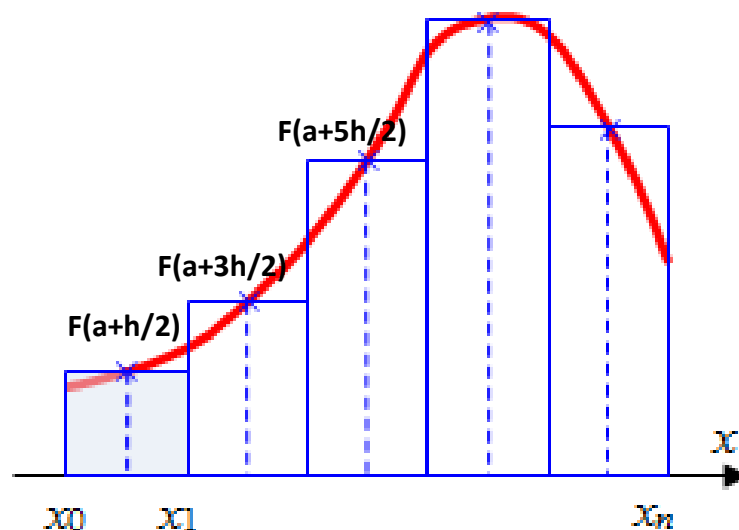


Метод средних прямоугольников

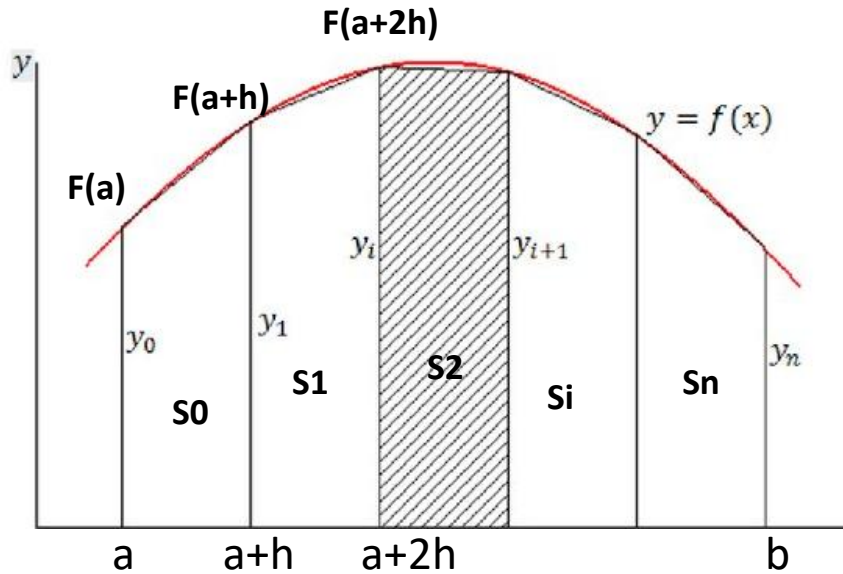
$$S = h * f(a + (2i+1)/2 * h)$$

Ширина – h

Высота – значение функции на
половине шага $f(a + h/2)$, $f(a + 3h/2)$,
 $f(a + 5h/2)$...



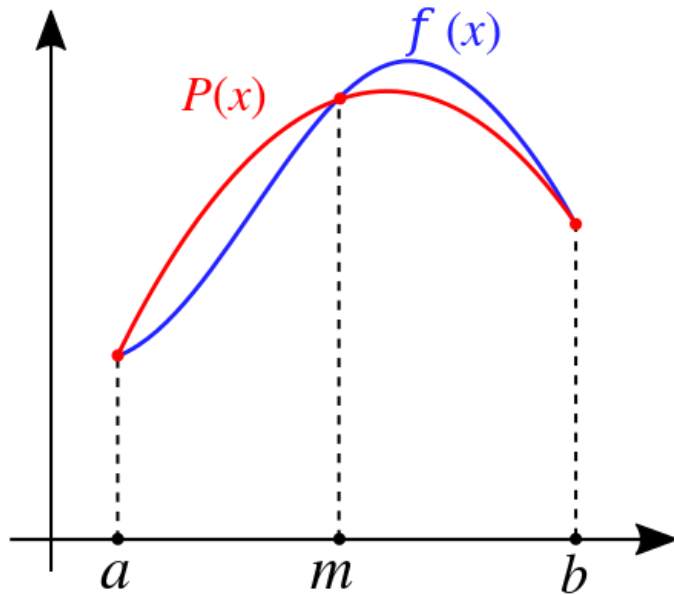
Метод трапеций



Алгоритм:

1. Применяем пункты 3-4 из алгоритма для прямоугольников
2. Рассчитываем площадь трапеций
$$S_i = h \cdot (f(a+ih) + f(a(i+1)h)) / 2$$
3. Считаем сумму площадей трапеций S

Метод симпсона



Алгоритм:

1. Задать отрезок $[a;b]$ и количество разбиений n
2. Разбиваем отрезок $[a;b]$ на $2n$ частей
 $h=(b-a)/2n$
3. Рассчитываем площади сдвоенных криволинейных трапеций
 $S_0 = h/3(f(a) + 4f(a+h)+f(a+2h))$
 $S_1 = h/3(f(a+2h) + 4f(a+3h)+f(a+4h))$
 $S_2 = h/3(f(a+4h) + 4f(a+5h)+f(a+6h))$
4. Находим сумму S всех трапеций
5. Увеличиваем кол-во разбиений в два раза $n^*=2n$, $h=(b-a)/2n$ и находим сумму площадей всех трапеций S_1
6. Повторяем 2-6 пока $|S-S_1| \geq \epsilon$

Разработка программного решения

```
#define A -10
#define B 10
#define E 1e-4
n=10;
do{
    h=double(B-A)/n;
    S1=0;
    for(i=0; i < n; i++)
        S1+=[ФОРМУЛА];
    n*=2;
    h=(double)(B-A)/n;
    S2=0;
    for(i=0;i<n;i++)
        S2+=[ФОРМУЛА];
}while(fabs(S1-S2)>E);
```


План отчета

1. Титульный лист (ФИО, курс, группа, название лабораторной);
2. Теоретическая часть (Теория, алгоритм)
3. Практическая часть (Блок схема к программе написанной на языке Си, результаты выполнения программы в виде таблицы);
4. Выводы о проделанной работе;