Numerical integration

$$I = \int_a^b f(x)\,dx \approx \widetilde{I}$$
$$\widetilde{I} = \sum_{q=1}^{Q} \omega_q\, f(x_q)\,,$$

where $x_q \in [a, b]$ are called nodes and ω_q weights.
Q number of nodes/weights.

Newton-Cotes rules

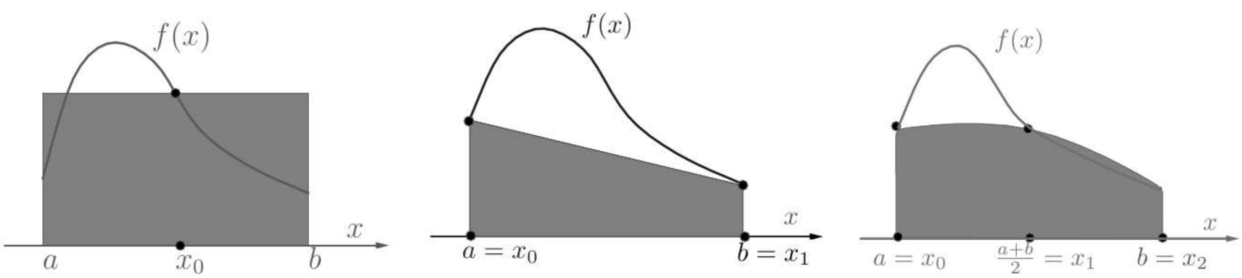

Figure 18.1: Midpoint, trapezoid and Simpson quadrature rules

Midpoint rectangle method

$$I_{r,m} = (b-a)f(\frac{a+b}{2})$$

1 node: $x_0 = \frac{a+b}{2}$
1 weight: $\omega_0 = (b-a)$
Exact for polynomials of degree at most 1.

Trapezoid method

$$I_{tr} = (b-a)\frac{f(a)+f(b)}{2}$$

2 nodes: $x_0 = a$, $x_1 = b$
2 weights: $\omega_0 = \omega_1 = \frac{b-a}{2}$
Exact for polynomials of degree at most 1.

Simpson method

$$I_{Sim} = (b-a)\frac{f(a) + 4f(\frac{a+b}{2}) + f(b)}{6}$$

3 nodes: $x_0 = a$, $x_1 = \frac{a+b}{2}$, $x_2 = b$
3 weights: $\omega_0 = \omega_2 = \frac{(b-a)}{6}$, $\omega_1 = 4\frac{(b-a)}{6}$
Exact for polynomials of degree at most 2.

Implementation

Implement functions for the above methods. For instance

```
 double rectangle(const double a, const double b, double  (* myf_ptr)( double )) {

double integral = 0.;

 ...

 return integral;

}
```

Composite rules

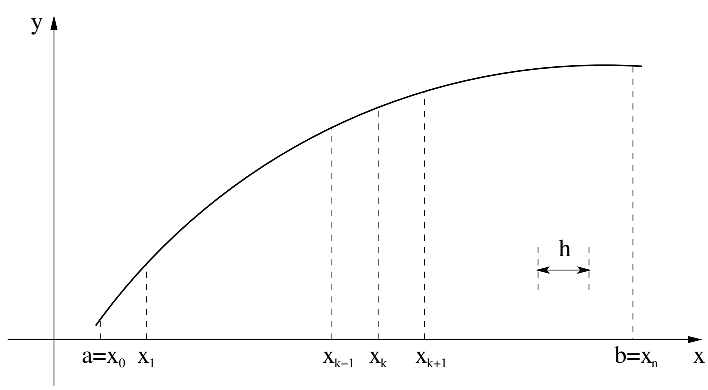

Figure 18.2: Partition of $[a, b]$ into n subintervals of length $h = (b-a)/n$

Partition $[a, b]$ into subintervals $[x_{k-1}, x_k]$, $k = 1, 2, \ldots, n$.

$$I = \int_a^b f(x)\,dx = \sum_{k=1}^{n} \int_{x_{k-1}}^{x_k} f(x)\,dx = \sum_{k=1}^{n} I_k \approx \sum_{k=1}^{n} \widetilde{I}_k\,.$$

Approximate each I_k with $\widetilde{I}_k$ obtained with a quadrature method.
If all $\widetilde{I}_k$ are approximated using a rectangle/trapezoid/Simpson rule,
you obtain a *composite rectangle/trapezoid/Simpson rule*.