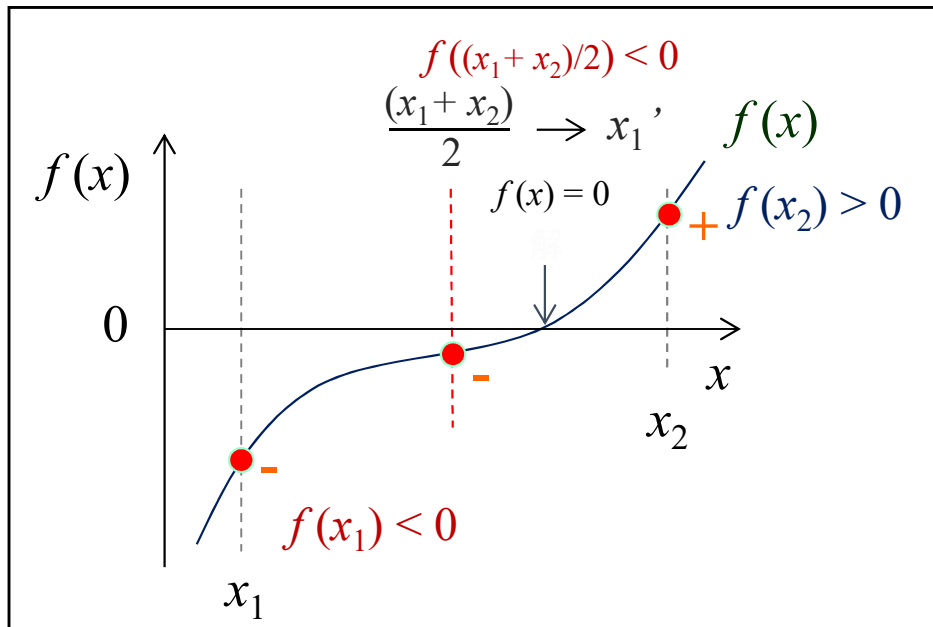


非線形方程式の解法

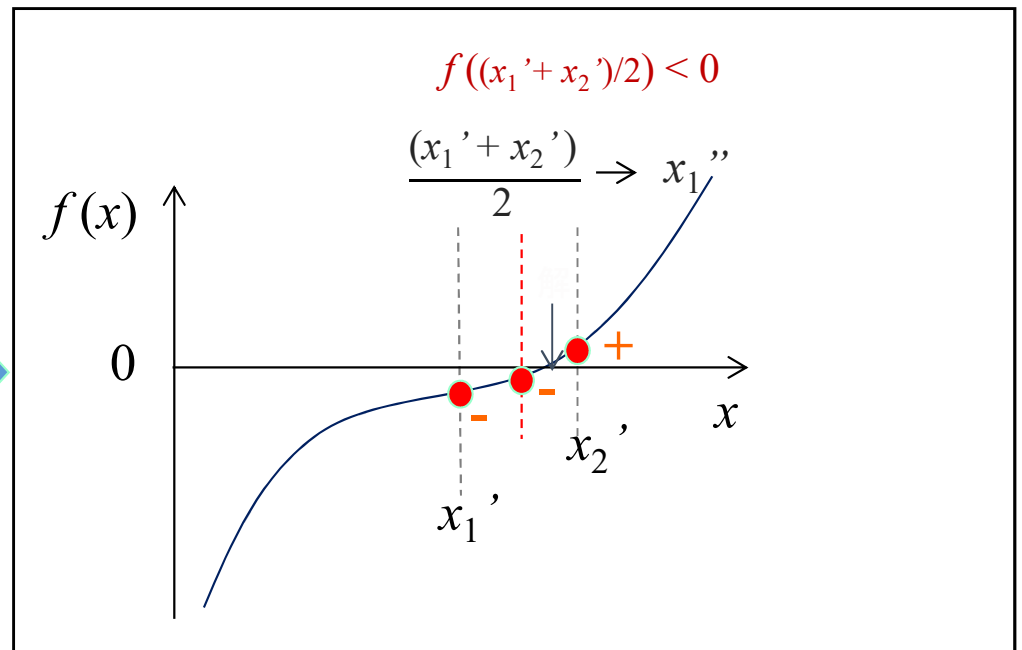
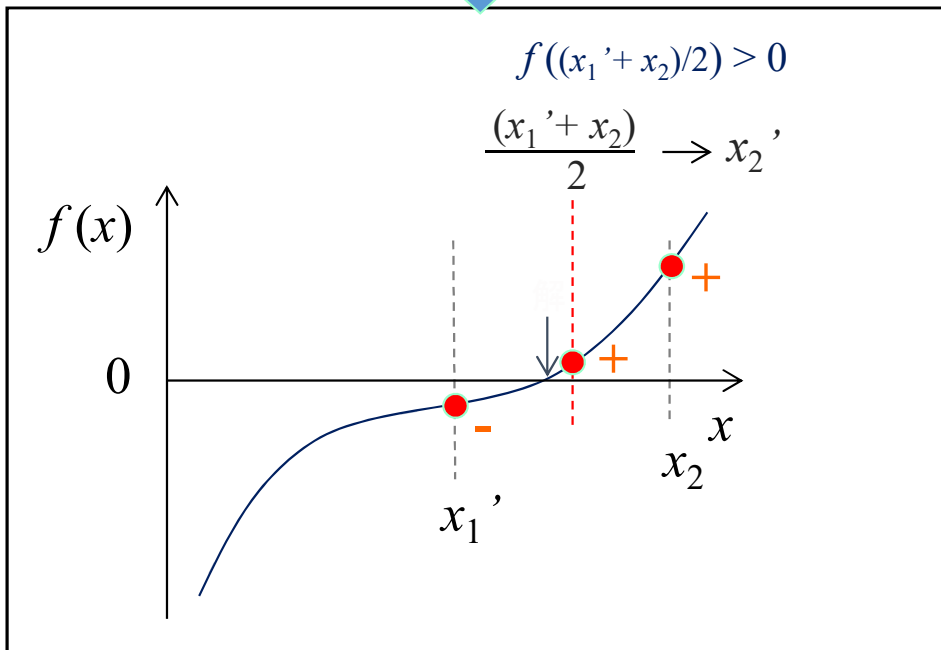
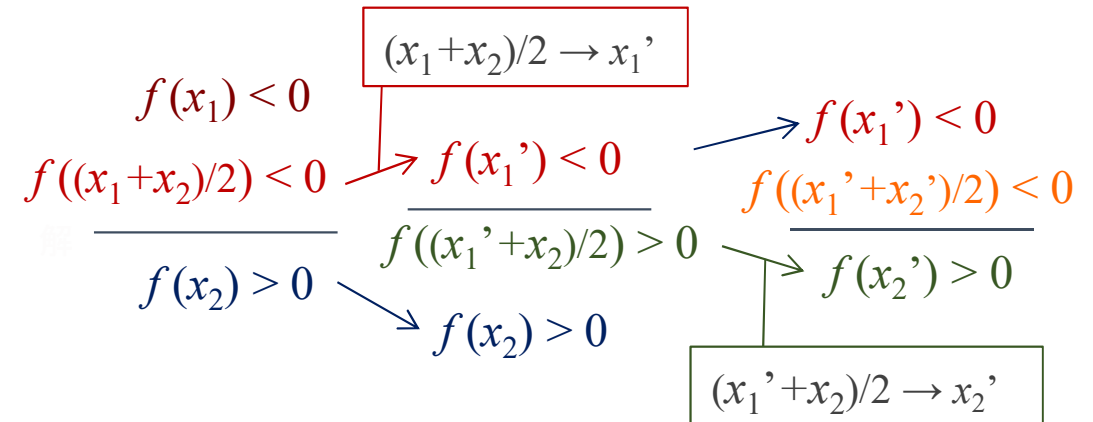
2分法



非線形方程式 $f(x) = 0$

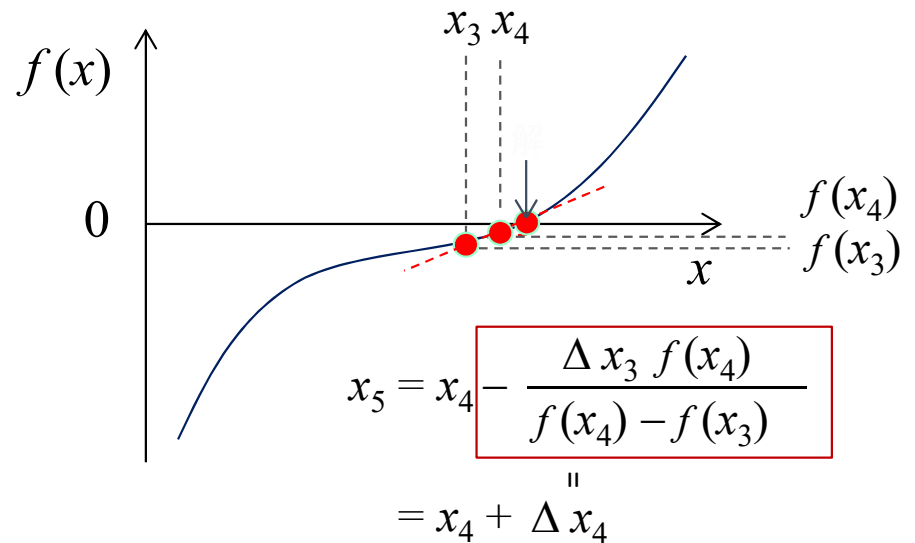
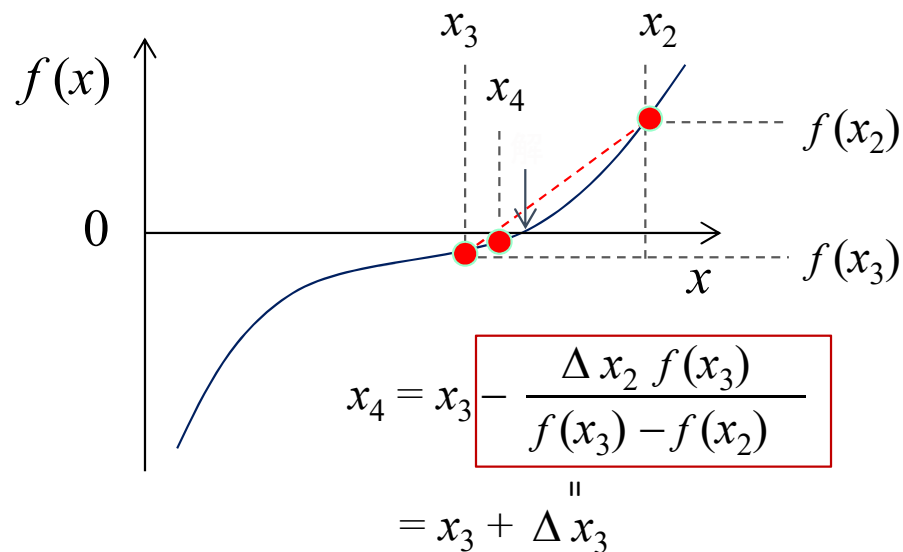
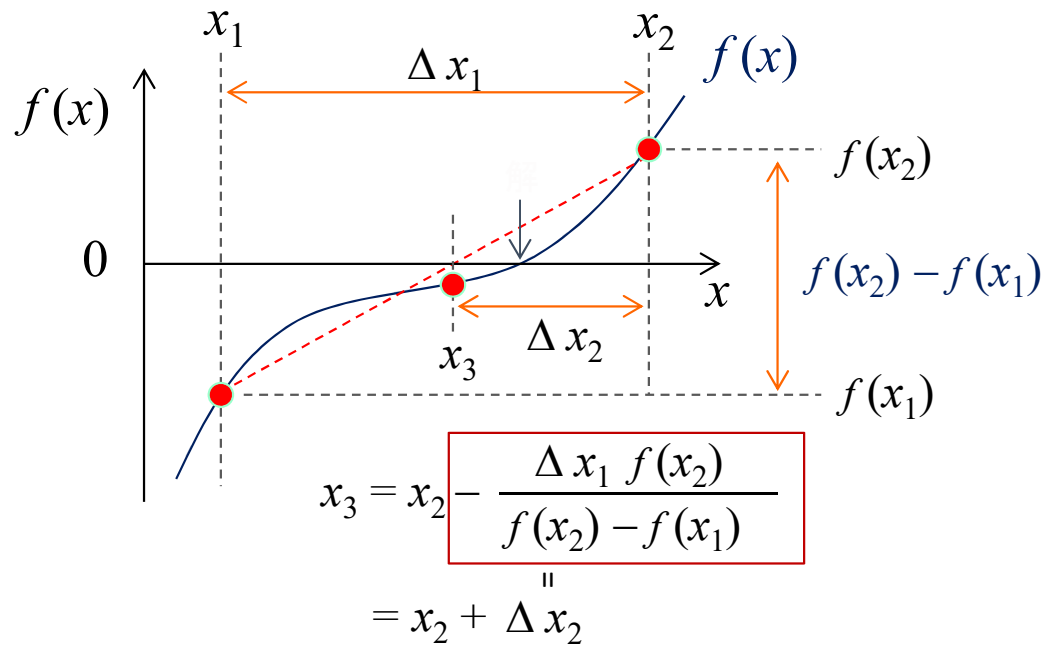
2分法

$f(x)$ が正と負になる x_2, x_1 から始め、その中間点の $f(x)$ の正負により解を狭め込む



非線形方程式の解法

割線法



非線形方程式 $f(x) = 0$

割線法 $f(x)$ の差分から解の位置を推定して正しい解に近づく

$$\Delta x_1 = x_2 - x_1$$

$$\Delta x_k = - \frac{\Delta x_{k-1} f(x_k)}{f(x_k) - f(x_{k-1})}$$

$$x_{k+1} = x_k + \Delta x_k$$

$$\lim_{k \rightarrow \infty} f(x_k) = 0$$

C++ クラス (データメンバとメンバ関数)

C++クラスの概念

クラス : データと関数の構造体

非公開の変数を公開する関数のみで操作することを可能にする

↓ ↓
非公開のデータメンバ 公開するメンバ関数

class クラス名

```
{  
private: // 省略可 ---- 非公開部分の定義 ----  
        非公開のデータメンバの宣言
```

```
public: // ---- 公開部分の定義 ----  
        公開するメンバ関数、フレンド関数の宣言  
        (クラス名と同じ関数はデータメンバを生成し初期化するコンストラクタ)  
};
```

class complex

```
{  
private:          クラスcomplex のデータメンバーへの引数  
    double re, im;  
public:          非公開データメンバーへの入力
```

```
    complex(double r=0, double i=0){ re=r ; im=i ;}  
    コンストラクタ  
    complex型のデータの生成とそのデフォルトの初期値を指定
```

```
    void print();  
    friend complex operator +( complex, complex);  
    friend complex operator *( complex, complex);  
    クラスcomplex のデータメンバーにアクセスできる関数
```

```
};
```

クラスの変数 a のメンバ re の値

a.re

↑ ↑
クラスの変数名 クラスのメンバ名

コンストラクタの使用例

```
complex a;      a のメンバ a.re=0, a.im=0 を生成  
complex b(12,0); b のメンバ b.re=12, b.im=0 を生成  
complex c=12;   c のメンバ c.re=12, c.im=0 を生成
```

公開するメンバ関数、フレンド関数の定義

```
void complex::print(){  
    printf("(%5.2f, %5.2f )\n", re, im);  
}
```

演算子の多重定義 (演算子関数の定義)

```
complex operator +(complex a, complex b){  
    return complex( a.re +b.re, a.im+b.im); }
```

```
complex operator *(complex a, complex b){  
    return complex( a.re *b.re - a.im*b.im,  
                    a.re *b.im + a.im*b.re); }
```

クラスの利用例

```
main(){  
    complex a(1, 2), b(0, 1), c;  
  
    c = a*b + b;  
    c.print();  
}
```

レポート課題 No.2

提出期限 1月25日（金）

各自が興味を持っている研究分野に関連することで、数値計算を利用して解析することができる課題を自由に設定し、その解析方法と解析結果についてレポートとしてまとめ、提出してください。

重力多体問題、熱統計力学、分子動力学、2次元温度場・静電場解、量子化学計算、固体結晶のバンド計算、量子多体問題、並列計算法などが考えられます。

使用する計算プログラムの言語、解析に必要なとなるソフトウェアやライブラリについて、制限はありません。