プログラミング基礎

2023年度1Q　火曜日３,４時限(10:45~12:25)
金曜日1~４時限(8:50~12:20)

工学院　情報通信系

中山実,　渡辺義浩
伊藤泉,　小杉哲
TA: 小泊大輝,　千脇彰悟

5/9(火) 8:50～12:25
「関数2」

1. 関数の利用例、配列や繰り返し
2. 数値計算への応用

文字型配列の宣言と初期化

第5回の復習

```
int main(void) {

char str[6]={'H','e','l','l','o','¥0'};
//char str[]={'H','e','l','l','o','¥0'};
//char str[6]="Hello";
//char str[]="Hello";

//str =“Hello”;

printf("%c¥n", str[0]);
printf("%s¥n ", str);

return 0;
}
```

文字型配列の宣言.
一番最後に文字列の終わりを示す¥0（＼0）を入れる．それを考えた長さにする.
下の書き方でも可能.

このように後で代入は不可.

個別要素の出力(変換仕様%c) .

全体の出力． この場合は変換仕様%sを使う.

実行結果

```
H
Hello
```

文字型配列の宣言と初期化

第5回の復習

```
int main(void) {

    char str[100];

    printf("Input a word¥n");
    scanf("%s", str);
    printf("The word is: %s ¥n", str);

    return 0;
}
```

大きい配列を宣言.

文字の入力．変化仕様は%s,
変数(str)の前に＆をつけない.

実行結果

```
Input a word
  Happy(input)
The word is: Happy
```

```c
#include <stdio.h>
#include <stdlib.h>

int strlength(char s[]);

int main(void) {

	char str[100];

	printf("Input a word¥n");
	scanf("%s", str);
	printf("The word is: %s, and length=%d ¥n", str, strlength(str));

	return 0;
}

int strlength(char s[])
{
	int i;

	i=0;
	while (s[i]!='¥0')
		++i;
	return i;
}
```

今回はstring.hを使わない
関数のプロトタイプ宣言

文字列の終端を探して
文字数の確認

```
#include <stdio.h>
#include <stdlib.h>

int strcopym(char str_out[], char str_in[]);

int main(void) {

        char str[100], stro[100];

        printf("Input a word ¥n");
        scanf("%s", str);
        strcopym(stro, str);
        printf("The original string is: %s, and copied string is %s ¥n", str, stro);

        return 0;
}

int strcopym(char str_out[], char str_in[])
{
        int i;

        i=0;
        while (str_in[i]!='¥0'){
                str_out[i]=str_in[i];
                i++;
        }
        str_out[i]='¥0';
        return i;
}
```

関数のプロトタイプ宣言

文字列の終端が来るまで
文字をコピー
コピー文字数を関数の戻り値

5/9(火) 10:45～12:25
「関数2」

1. 関数の利用例、配列や繰り返し
2. 数値計算への応用

プログラムによる数値計算

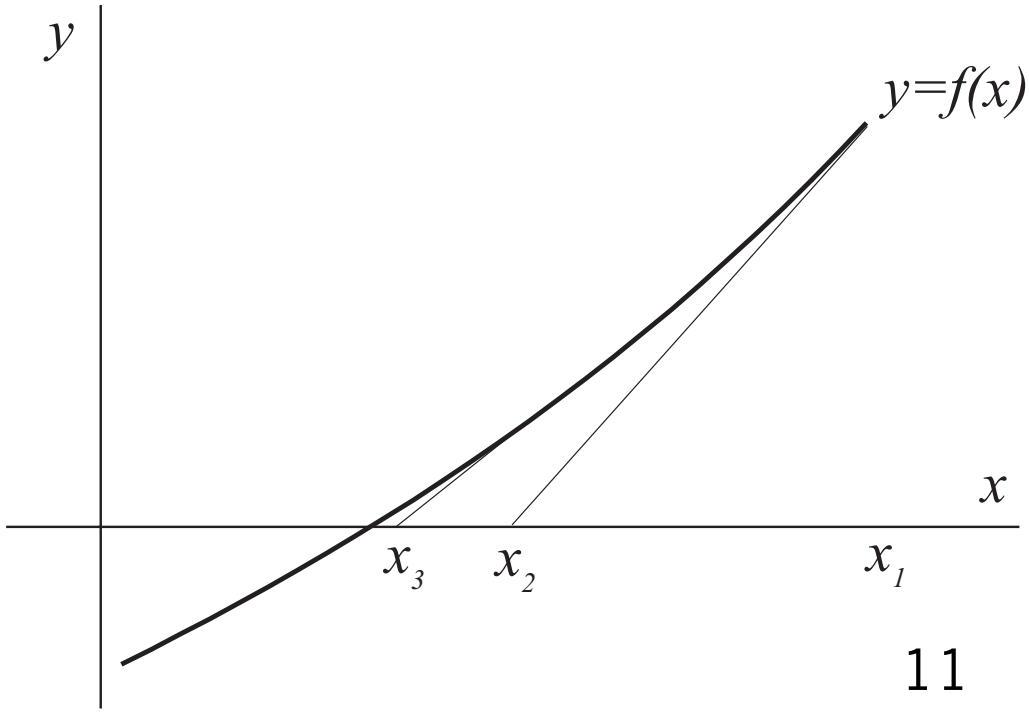
- 数値計算課題の例
  - 解の算出 (*f*は微分可能)：$f(x) = 0$
  - 複雑な式の場合は、近似的解法：Newton法

$$f'(x_1) = \frac{f(x_1)}{(x_1 - x_2)}$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$



- 例題：次式の正の解を算出する

$$2\sin x = x$$

$$f(x) = 2\sin x - x = 0$$

$$x_{n+1} = x_n - \frac{x_n - 2\sin x_n}{1 - 2\cos x_n} = \frac{2(\sin x_n - x_n \cos x_n)}{1 - 2\cos x_n}$$

n	x_n	分子	分母	x_{n+1}
1	2	3.48	1.83	1.90
2	1.90	3.1211	1.6466	1.8955
3	1.8955	3.1049	1.6380	1.89549

関数で、x_nを引数として、x_{n+1}を返り値とすれば、
関数を繰り返し実行することで、x_nが更新される。

収束の判定：　x_n　x_{n+1}　収束の条件を0.001以下とした場合、n=3で終了

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

float ff(float x);

int main(void) {

	float c,x=2;

	while(fabsf((c=ff(x))-x)>0.001){
		x=c;
	}
	return EXIT_SUCCESS;
}

float ff(float x)
{
	float a,b,c;

	a = 2*(sin(x)-x*cos(x));
	b = 1-2*cos(x);
	c = a/b;
	printf("a=%f, b=%f, c=%f, d=%f ¥n", a,b,c,fabsf(c-x));
	return c;
}
```

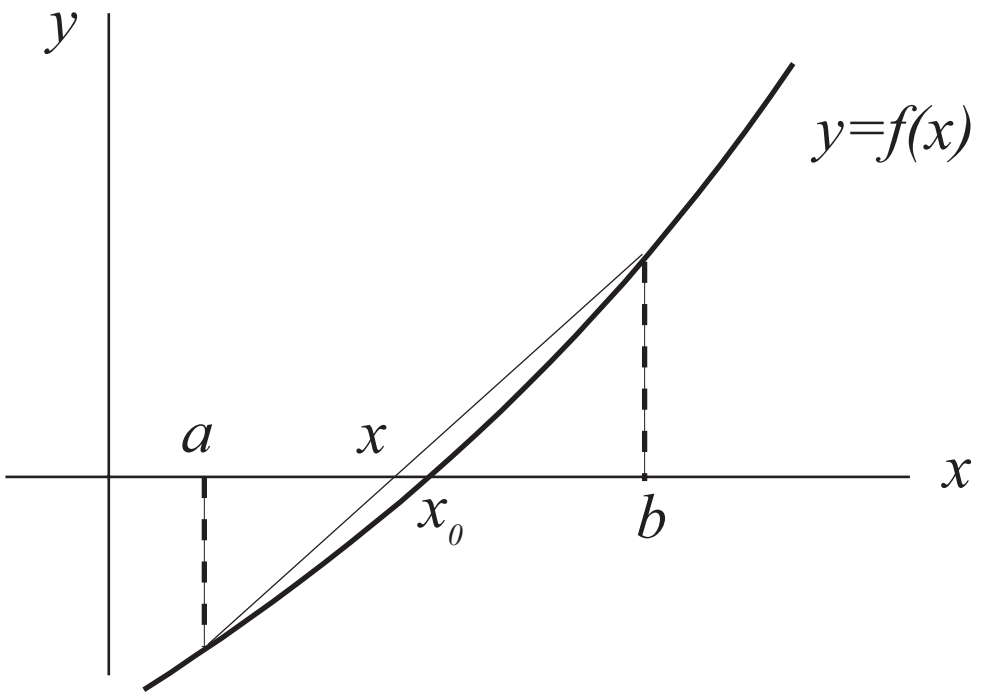
関数のプロトタイプ宣言

収束の判定

xの更新

- 逐次改良法

$$x = \frac{af(b) - bf(a)}{f(b) - f(a)}$$



x_0を含む区間を設定し、上式を用いて、a または b を更新する。
上図の場合は、aを更新し、更新差分で終了を判定する。
前述の方法が使えない時に利用する。