

# プログラミング基礎

## 1119 課題(1) 解説

### 平均と標準偏差の計算

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

```
void data(int data[], int n);
double average(int data[], int n);
double deviation(int* data, int n);
```

プロトタイプ宣言

```
int main(void){//main関数
    double ave,dev; // 平均, 標準偏差
    int a[100];//アクセスはa[0]~a[99]
    int n;//要素数
```

```
    scanf("%d", &n);
    data(a, n);
```

```
    ave = average(a,n);//平均の計算
    dev = deviation(a,n);//標準偏差の計算
```

```
    printf("average = %lf\n", ave);
    printf("deviation = %lf\n", dev);
```

データの出力

```
    return 0;
}
```

• データの読み込み

```
void data(int* data, int n){
    int i;

    for(i=0; i<n; i++)
        scanf("%d", &(data[i]));

    for(i=0; i<n; i++)
        printf("%d ",data[i]);

    printf("\n");
}
```

# 平均と標準偏差の計算

```
double average(int data[], int n){
    int i; /* ループのためのカウンタ */
    int sum = 0; //和の初期化
    for(i=0; i<n; i++)
        sum += data[i];
    return (double)sum/(double)n;
}

double deviation(int* data, int n){
    int i; //ループのためのカウンタ
    double ave; // 平均
    int sum = 0; //二乗和の初期化
    for(i=0; i<n; i++)
        sum += data[i]*data[i];
    ave = average(data, n); //平均値の計算
    return sqrt((double)sum/(double)n - ave*ave);
}
```

配列を関数の引数にするとき  
int data[]または int\* data

data[i]でi番目の値にアクセス

関数の中から別の関数  
を呼び出すこともできる

# 配列の値を累積値への更新

```
int main(void){
    int i; /* ループのためのカウンタ */
    int a[100];
    int n;

    scanf("%d", &n);
    data(a, n);

    for(i=1; i<n; i++)
        a[i] += a[i-1];

    for(i=0; i<n; i++)
        printf("%d ",a[i]);

    printf("¥n");

    return 0;
}
```

データの読み込み

カウントは1から

データの出力

# 最小値とそのインデックスを求める

```
int main(void){  
    int index,i,n;  
    int a[100];
```

```
    scanf("%d", &n);  
    data(a, n);
```

 } データの読み込み

```
    index=0;  
    for(i=1;i<n;i++){  
        if(a[i] < a[index])  
            index = i;  
    }
```



1. とりあえずindex=0とし, a[0]が最小値であるとする
2. a[i] (a[1] ~ a[n-1]) までの値とa[index]を比較
3. a[i]の方が小さければindexをiに更新

```
    printf("Index: %d¥n", index);  
    printf("Min: %d¥n", a[index]);
```

 } データの出力

```
    return 0;  
}
```

# 配列を昇順へ並べ替える

```
int main(void){  
    int i,n;  
    int a[100];
```

```
    scanf("%d", &n);  
    data(a, n);
```

 } データの読み込み

```
    sort(a, n); //ソート
```

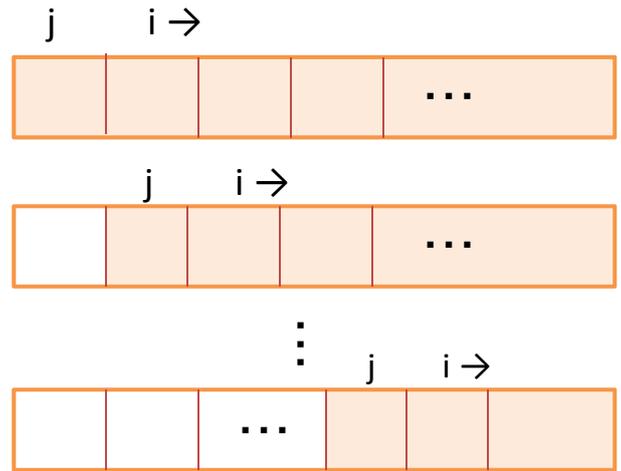
```
    for(i=0; i<n; i++){  
        printf("%d ",a[i]);
```

 } データの出力

```
    return 0;  
}
```

# 配列を昇順へ並べ替える

```
void sort(int data[], int n){  
    int index, i, j;  
    int temp;  
    for(j = 0; j < n-1; j++){  
        index = j;  
        for(i = j+1; i < n; i++){  
            if( data[i] < data[index] )  
                index = i;  
        }  
        temp = data[index];  
        data[index] = data[j];  
        data[j] = temp;  
    }  
}
```



j番目の値と最小値の  
入れ替え