

課題09

Q01: 配列の平均値と分散を求める

(プログラム名: [ex09_1.cpp](#))

関数 `calc_mean_var()` を作成し、配列中の要素の平均値(`mean`)と分散(`variance`)を計算しなさい。

※ここでは「母分散」(標本分散ではなく、単純に n で割った分散)を計算します。

`main()` 関数は変更してはいけません。

【プログラム例】

```
C/C++
#include <iostream>
#include <cmath>
using namespace std;

// 平均値と分散を求める関数
//   arr[] : 入力データ配列 (変更不可のため const)
//   n : 配列の要素数 (変更不可)
//   mean : 平均値 (参照渡しで返す)
//   var : 分散 (参照渡しで返す)
void calc_mean_var(const float arr[], const int n, float &mean, float &var)
{
    // ↓ ここを補完
}

int main()
{
    // --- テスト用配列1 ---
    float data1[] = {1.0f, 2.0f, 3.0f, 4.0f, 5.0f};
    int n1 = sizeof(data1) / sizeof(data1[0]);

    // --- テスト用配列2 ---
    float data2[] = {10.0f, 15.0f, 20.0f};
    int n2 = sizeof(data2) / sizeof(data2[0]);

    float mean, var;
    calc_mean_var(data1, n1, mean, var);
    cout << "配列1: 平均値 = " << mean << ", 分散 = " << var << endl;
    calc_mean_var(data2, n2, mean, var);
    cout << "配列2: 平均値 = " << mean << ", 分散 = " << var << endl;
}
```

【実行例】

```
None  
配列1: 平均値 = 3, 分散 = 2  
配列2: 平均値 = 15, 分散 = 16.6667
```

Q02:

(プログラム名 :[ex09_2.cpp](#))

関数 `count_call()` を作成しなさい。この関数の内部で 静的変数 (static 変数) を使い、関数が呼び出された回数を数えるようにします。呼び出すたびに、現在の呼び出し回数を画面に表示してください。

`main()` 関数は変更してはいけません。

【プログラム例】

```
C/C++  
#include <iostream>  
using namespace std;  
  
// TODO: 関数を定義する  
// 関数内で静的変数を使って呼び出し回数をカウントする  
void count_call()  
{  
    // ↓ ここを補完  
}  
  
int main()  
{  
    // 関数を3回呼び出す  
    count_call();  
    count_call();  
    count_call();  
    return 0;  
}
```

【実行例】

```
None  
count_call: 1 回目の呼び出しだす。  
count_call: 2 回目の呼び出しだす。  
count_call: 3 回目の呼び出しだす。
```

Q03: 関数ポインタを使って四則演算を行う

(プログラム名: [ex09_3.cpp](#))

ユーザーから次の3つの値を入力し、関数ポインタを使って四則演算(加算・減算・乗算・除算)を行うプログラムを作成しなさい。

- 第1の数(int)
- 演算子(+, -, *, /)
- 第2の数(int)

入力された演算子に応じて、対応する関数を関数ポインタ経由で呼び出し、計算結果を表示すること。

【ヒント】

- 加算、減算、乗算、除算の4つの関数をそれぞれ定義すること。
- 関数ポインタを定義して、入力された演算子に応じてどの関数を呼ぶかを切り替える。

【実行例1】

- 入力

None

```
第1の数を入力してください: 5
演算子 (+, -, *, /) を入力してください: *
第2の数を入力してください: 3
```

- 出力

None

```
5 * 3 = 15
```

```
第1の数を入力してください: 5
演算子 (+, -, *, /) を入力してください: *
第2の数を入力してください: 3
5 * 3 = 15
```

【実行例2】

```
第1の数を入力してください: 8
演算子 (+, -, *, /) を入力してください: -
第2の数を入力してください: -5
8 - -5 = 13
```