

第 11 回演習

演習 1:accumulate, ex11-1.cpp

1 万円の予算に対して、プラン A とプラン B の二つの使い方を考えた。std::accumulate() と std::minus<> を使って二つの残額を計算せよ。

ヒント: (((((10000-1500)-2000)-1800)-780)-650)

```
int total {10000};
std::map<std::string, std::vector<int>> plans
{"plan A", {1500, 2000, 1800, 780, 650}},
 {"plan B", {5500, 1000, 320, 1580, 1650}}};
for (auto& [n,p]: plans) {
    std::cout << n << ": "
    << std::accumulate( /* write here */ )
    << "\n";
}
```

plan A: 3270

plan B: -50

演習 2: ラムダ式, ex11-2.cpp

- ▶ 以下のプログラム (第9回資料 p.27, 一部省略) の pred 関数をやめて実引数をラムダ式にせよ。

```
bool pred(int x, int y) { return x % y == 0; }

int main()
{
    std::vector a{3,2,4,6,8,1,2,3,4}, s{1,2,3};
    auto it { std::search(a.begin(), a.end(),
                          s.begin(), s.end()) };
    if (it != a.end())
        std::cout << it-a.begin() <<"\n"; // 5

    it = std::search(a.begin(), a.end(),
                    s.begin(), s.end(), pred);
    if (it != a.end())
        std::cout << it-a.begin() <<"\n"; // 1
}
```

演習 3: accumulate とラムダ式, ex11-3.cpp

$$\sum_{k=1}^n k^4 = \frac{1}{30}n(n+1)(2n+1)(3n^2+3n-1)$$

を確認したい。以下を埋めてプログラムを完成させよ。

```
int main(int argc, char *argv[]) {
    int n { argc>1 ? std::stoi(argv[1]):10 };
    std::vector<int> x(n);
    auto b{x.begin()}, e{x.end()};
    // 1 から n までの値を x に設定
    std::cout << std::accumulate(b, e, 0, /* ... */ )
    <<" " << /* 確認する式 */ <<"\n";
}
```

```
$ ./a.out
25333 25333
$ ./a.out 20
722666 722666
$ ./a.out 30
5273999 5273999
```

演習共通クラス

続く演習では, 説明で使った `print.hpp` と以下のヘッダファイルを共通に使用する

```
// sales.hpp
#include <iostream>
class Sales {
    std::string item;
    int num;
public:
    Sales(std::string i, int n):item(i),num(n){}
    auto geti() const { return item; }
    auto getn() const { return num; }
    friend auto&
    operator<<(std::ostream& o, const Sales& s) {
        return o <<"["<< s.item <<":"<< s.num <<"]";
    }
};
```

演習 4:std::sort とクラス, ex11-4.cpp

Sale クラスの vector 配列を item 順と num 順でそれぞれ整列させて出力するプログラムを完成させよ。

```
#include <algorithm>
#include <vector>
#include "print.hpp"
#include "sales.hpp"
int main() {
    std::vector<Sales> v {{ "xyz",3},{ "abc",2},
                        { "ijk",1},{ "opq",4}};

    auto b{v.begin()}, e{v.end()};
    std::sort(b, e, /* ... */ );
    print(v);
    std::sort(b, e, /* ... */ );
    print(v);
}
```

```
[abc:2] [ijk:1] [opq:4] [xyz:3]
[ijk:1] [abc:2] [xyz:3] [opq:4]
```

演習 5:std::accumulate とクラス, ex11-5.cpp

Sale クラスの vector 配列から num の合計を得るプログラムを std::accumulate により完成させよ。

```
#include <numeric>
#include <vector>
#include "sales.hpp"
int main()
{
    std::vector<Sales> v {{"xyz",3},{"abc",2},
                        {"ijk",1},{"opq",4}};
    auto b{v.begin()}, e{v.end()};
    std::cout <<"total = "
    << std::accumulate(b, e, 0, /* ... */)
    <<"\n"; // total = 10
}
```

演習 6:std::find_if とクラス, ex11-6.cpp

Sale クラスの vector 配列からコマンド引数で指定した item を探して出力するプログラムを std::find_if により完成させよ。

```
#include <algorithm>
#include <vector>
#include "sales.hpp"
int main(int argc, char *argv[])
{
    std::string item { argc>1 ? argv[1]:"" };
    std::vector<Sales> v {{ "xyz",3},{ "abc",2},
                        { "ijk",1},{ "opq",4}};
    auto it { std::find_if(v.begin(), v.end(),
                          /* ... */ ) };
    if (it != v.end())
        std::cout << *it << "\n";
    else
        std::cout << item << " : not found\n";
}
```



```
$ ./a.out  
: not found  
$ ./a.out abc  
[abc:2]  
$ ./a.out lmn  
lmn : not found  
$ ./a.out xyz  
[xyz:3]
```