

## 第4回 C++プログラミング実験II 実験課題

課題実施日:2022 年 5 月 2 日. 締切 5月5日(木)24:00

### 実験課題 4-1

以下の実行例を参考にし、vectorを用いて、一文字ずつ配列に格納し、その後に配列を操作して、例のように単語を表示するプログラム(スケルトンファイル名: ex04-1\_main.cpp)を作成せよ。(文字入力数は7文字のみを前提とするが、文字を配列に格納後は、size関数を用いて配列長を取得せよ。)

```
//実行例 1
% ./ex04-1
d e l i v e r
deliver
reviled
live
```

```
//実行例 2
% ./ex04-1
g a t e m a n
gateman
nametag
tema
```

```
//実行例 3
% ./ex04-1
r e p a p e r
repaper
repaper
pape
```

## 実験課題 4-2

2つの vector  $x$ ,  $y$  を用意したのちに各要素に整数値を格納し,  $x$  と  $y$  の等価性および大小関係を判定するプログラムを作成する. 実行例を参考にし, 以下 ex04-2-main.cpp の main 関数を適当に埋めプログラムを完成せよ.

- ① 実行例のように vector の要素を出力する print 関数を作成する
- ② vector の要素(整数値)となる一桁の数字(文字)の並びを入力する処理(ただし, #が入力されたら格納を終了する)
- ③ キーボード入力した値を  $x[1]$  に挿入する.  
vector  $y$  の先頭に vector  $x$  の最後の要素を挿入する.  
 $y[2]$ の要素を削除する.

```
// ex04-2-main.cpp
#include <vector>
#include <iostream>
using std::cin, std::cout, std::vector, std::string;

//(1) 実行例のように vector の要素を出力する print 関数を作成する

int main()
{
    vector<int> x;
    vector<int> y;

    cout << "Input elements of a vector x.¥n";
    //(2) vector x の要素(整数値)となる一桁の数字(文字)の並びを入力する処理
    //(ただし, #が入力されたら格納を終了する)
    cout << "¥n";
    cout << "Input elements of a vector y.¥n";

    //(2) vector y の要素(整数値)となる一桁の数字(文字)の並びを入力する処理
    //(ただし, #が入力されたら格納を終了する)
    cout << "Input number: ";
    int t{};    cin >> t; //x[1]に挿入するための変数 t

    cout << "¥n--original--¥n";
    cout << "x: ";
    print(x);
    cout << "¥n";
    cout << "y: ";
    print(y);
    cout << "¥n";
```

```
// ex04-2-main.cpp 続き
```

```
cout << std::boolalpha;
cout << "x == y " << (x == y) << "¥n";
cout << "x != y " << (x != y) << "¥n";
cout << "x < y " << (x < y) << "¥n";
cout << "x <= y " << (x <= y) << "¥n";
cout << "x > y " << (x > y) << "¥n";
cout << "x >= y " << (x >= y) << "¥n";
```

```
/*(3)
```

- ・キーボード入力した値を x[1]に挿入する
- ・vector y の先頭に vector x の最後の要素を挿入する
- ・y[2]の要素を削除する

```
*/
```

```
cout << "¥n-----¥n";
cout << "x: ";
print(x);
cout << "¥n";
cout << "y: ";
print(y);
cout << "¥n";

cout << std::boolalpha;
cout << "x == y " << (x == y) << "¥n";
cout << "x != y " << (x != y) << "¥n";
cout << "x < y " << (x < y) << "¥n";
cout << "x <= y " << (x <= y) << "¥n";
cout << "x > y " << (x > y) << "¥n";
cout << "x >= y " << (x >= y) << "¥n";
```

```
}
```

- ✓ 実行例①のように, original では x と y は等しいが要素を操作後に x[0] < y[0] となるため, ベクトル x はベクトル y より小さいと判定される(ベクトル x の方が要素数が大きい, そのことは無関係である).
- ✓ 2つのベクトルの等価性と大小関係を判定するための各演算子が非メンバ関数として提供されていることを様々な値を用いて実行し, 確認せよ.

//実行例①(下線部はキーボード入力)

\$ g++ -std=c++17 ex04-2-main.cpp -o ex04-2

\$ ./ex04-2

Input elements of a vector x.

x[0] = 1

x[1] = 2

x[2] = 3

x[3] = #

Input elements of a vector y.

y[0] = 1

y[1] = 2

y[2] = 3

y[3] = #

Input number: 9

--original--

x: 1 2 3

y: 1 2 3

x == y true

x != y false

x < y false

x <= y true

x > y false

x >= y true

-----  
x: 1 9 2 3

y: 3 2

x == y false

x != y true

x < y true

x <= y true

x > y false

x >= y false

//実行例②

\$ g++ -std=c++17 ex04-2-main.cpp -o ex04-2

\$ ./ex04-2

Input elements of a vector x.

x[0] = 1

x[1] = 2

x[2] = 3

x[3] = 4

x[4] = 5

x[5] = #

Input elements of a vector y.

y[0] = 3

y[1] = 2

y[2] = 1

y[3] = #

Input number: -1

--original--

x: 1 2 3 4 5

y: 3 2 1

x == y false

x != y true

x < y true

x <= y true

x > y false

x >= y false

-----  
x: 1 -1 2 3 4 5

y: 5 2

x == y false

x != y true

x < y true

x <= y true

x > y false

x >= y false

### 実験課題 4-3

文字列が格納された MyVec x, y について, 文字の欠損や順序逆転が生じた y を訂正して x と等価にするプログラムを作成したい。そこでまず, MyVec クラスを myvec.hpp 内に実装し, 課題 2 を参考に(1)print 関数や(2, 3, 6)main 関数内の文字入力を実装しない。(4) 次に, MyVec に operator==を実装しない。(5) 最後に, MyVec に要素を逆順にするメンバ関数 reverse を実装しない。ただし, 最初の正誤判定の際に一部でも文字の欠損があった場合でも、逆順への並べ替え処理を行うものとする。

```
// ex04-3-main.cpp
//
#include <iostream>
#include "myvec.hpp"
using std::cin, std::cout, std::vector;

//(1) 実行例のように MyVec の要素を出力する print 関数を作成する

int main() {
    MyVec<char> x, y;

    cout << "Input elements of a vector x.¥n";
    //(2) MyVec x の要素(文字)を入力する処理(ただし, #が入力されたら格納を終了する)

    cout << "Input elements of a vector y.¥n";
    //(3) MyVec y の要素(文字)を入力する処理(ただし, #が入力されたら格納を終了する)

    cout << std::boolalpha;
    //(4) x と y の等価性を判定
    cout << "x == y " << (x == y) << "¥n";

    cout << "Reversal of order.¥n";
    //(5) y を逆順にする
    y.reverse();

    print(y);
    cout << "¥n";

    cout << "Add elements.¥n";
    //(6) MyVec y の要素(文字)を追加する処理(ただし, #が入力されたら格納を終了する)
    print(y);
    cout << "¥n";

    cout << "x == y " << (x == y) << "¥n";

    return 0;
}
```

```
// 実行例 1
% ./ex04-3
Input elements of a vector x.
capacity#
Input elements of a vector y.
apac#
x == y false
Reversal of order.
capa
Add elements.
city#
capacity
x == y true
```

```
// 実行例 2
% ./ex04-3
Input elements of a vector x.
capacity#
Input elements of a vector y.
ticapac#
x == y false
Reversal of order.
capacit
Add elements.
y#
capacity
x == y true
```