

15. IPv4 data type

Write a class that represents an IPv4 address. Implement the functions required to be able to read and write such addresses from or to the console. The user should be able to input values in dotted form, such as 127.0.0.1 or 168.192.0.100. This is also the form in which IPv4 addresses should be formatted to an output stream

15. IPv4 데이터 타입

IPv4 주소를 나타내는 클래스를 작성하라. 콘솔로부터 IPv4 주소들을 읽고 쓸 수 있기 위해 필요한 함수를 구현하라. 사용자는 값을 127.0.0.1 이나 168.192.0.100과 같이 점이 포함된 형태 (dotted form)로 입력할 수 있어야 한다. 또한 IPv4 형태의 주소들을 출력 스트림으로 만들 수 있어야 한다.

16. Enumerating IPv4 addresses in a range

Write a program that allows the user to input two IPv4 addresses representing a range and list all the addresses in that range. Extend the structure defined for the previous problem to implement the requested functionality.

16. 일정 범위 안에서 IPv4 주소들을 열거

사용자로부터 범위를 나타내는 두 IPv4 주소들을 입력받아서 범위안의 모든 주소들을 열거하는 프로그램을 작성하라. 본 문제에서 요청된 기능을 구현하기 위해서, 이전 문제에서 정의된 구조를 확장하라.

17. Creating a 2D array with basic operations

Write a class template that represents a two-dimensional array container with methods for element access (at() and data()), capacity querying, iterators, filling, and swapping. It should be possible to move objects of this type.

17. 2D 배열과 기본적인 연산들을 만들기

엘리먼트에 접근하기 위한 메소드들(at()과 data()), 용량, 쿼리, iterator들, 채우기 그리고 바꾸기 메소드들을 포함한, 2차원 배열 컨테이너를 나타내는 클래스 템플릿을 작성하라.

18. Minimum function with any number of arguments

Write a function template that can take any number of arguments and returns the minimum value of them all, using operator < for comparison. Write a variant of this function template that can be parameterized with a binary comparison function to use instead of operator <.

18. 어떤 숫자도 인수로 받을 수 있는 최소(minimum) 함수

연산자 <를 비교에 사용하고 어떠한 형태의 숫자도 인수로 받고, 그 것들의 최소값을 반환하는 함수 템플릿을 작성하라. 연산자 <를 대신하여 이진 비교 함수로 parameterize될 수 있는, 이 함수의 다른 형태를 작성하라.

19. Adding a range of values to a container

Write a general-purpose function that can add any number of elements to the end of a container that has a method `push_back(T&& value)`.

19. 컨테이너에 어떤 값의 범위를 추가하기

`pushpack(T&& value)` 메소드를 가진 컨테이너의 끝에 원하는 수의 요소를 추가할 수 있는 General-purpose 함수를 작성하라.

20. Container any, all, none

Write a set of general-purpose functions that enable checking whether any, all, or none of the specified arguments are present in a given container. These functions should make it possible to write code as follows:

20. Container any, all, none

어떤, 모든 특정한 인수들이 받은 컨테이너에서 제공되는 지, 또는 전혀 제공되지 않는 지, 체크할 수 있는 General-purpose 함수들의 세트를 작성하라. 이 함수들은 다음과 같은 코드를 작성 가능하도록 해야 한다:

21. System handle wrapper

Consider an operating system handle, such as a file handle. Write a wrapper that handles the acquisition and release of the handle, as well as other operations such as verifying the validity of the handle and moving handle ownership from one object to another.

21. 시스템 handle wrapper

파일 핸들과 같은 운영 체제의 핸들을 고려하고, 핸들의 유효성 검증, 어떤 객체에서 다른 객체로 핸들의 ownership 이동과 같은 동작 뿐만 아니라, 핸들의 획득과 해제를 관리하는 wrapper를 작성하라.

22. Literals of various temperature scales

Write a small library that enables expressing temperatures in the three most used scales, Celsius, Fahrenheit, and Kelvin, and converting between them. The library must enable you to write temperature literals in all these scales, such as `36.5_deg` for Celsius, `97.7_f` for Fahrenheit, and `309.65_K` for Kelvin; perform operations with these values; and convert between them.

23. 다양한 온도 스케일 표기

가장 많이 사용되는 섭씨, 화씨, Kelvin의 세 가지 온도 스케일을 표현하고 변환할 수 있도록 작은 라이브러리를 작성하라. 이 라이브러리는 당신이 `36.5_deg`로 섭씨를, `97.7f`로 화씨를, 그리고 `309.65_K`로 Kelvin을 온도 문자 표현으로 작성할 수 있게 해야 하며, 이 값들로 연산을 수행하고 변환을 한다.