

Min and Max Algorithms

Solutions

- Describe the max() and min() algorithm functions
 - Return the larger (smaller) of their arguments
 - By default, the < operator of the elements is used
- What arguments do they take?
 - The arguments to be compared, and an optional predicate function to perform the comparison

- Write a simple program to demonstrate their use
- Alter your program to use a predicate function
- Write a similar program, but this time use an initializer list as argument to the `max()` and `min()` calls

- Describe `std::pair`
 - `std::pair` is a data structure with two elements
- When is `std::pair` useful?
 - Passing or returning pairs of related data to and from functions
- How is this data accessed?
 - The elements are called “first” and “second”
- How is a variable of type `std::pair` created?
 - Direct construction, or by calling `std::make_pair()`

- Write a program which demonstrates the use of `std::pair`

- Describe the minmax() algorithm function
 - minmax() returns an std::pair in which the first element contains the smallest element and the second element is the largest
- Write a simple program to demonstrate its use

- Describe the `max_element()` and `min_element()` algorithm functions
 - `max_element()` and `min_element()` are versions of `max()` and `min()` which take an iterator range
 - They return an iterator to the largest or smallest element
- Write a simple program to demonstrate their use

- Describe the `minmax_element()` algorithm function
 - `minmax_element()` takes an iterator range
 - It returns a pair of iterators
 - The first element is the smallest, the second element is the largest
- Write a simple program to demonstrate its use