Structured Bindings and Custom Types Exercises

Function which Returns Different Types

- Is it possible to have a C++ function which returns different types in different branches?

Structured Bindings and Custom Types

- Write a class
- Use constexpr if to implement a get<> member function for your class
- Write a function which returns an object of this class
- Write a program which uses a structured binding to unpack the elements of the returned class

Implementing tuple_size and tuple_element

- (Optional)
- Modify your solution so that it displays the last member (in declaration order) of the class
- Make sure that your solution safely handles the case of a class with no members