

Rvalue References Solutions

- Describe what happens what the following operations are performed on two variables:
 - Copying
 - A copy of the source's data is made and used to populate the target
 - Swapping
 - The source and the target exchange their data without copying
 - Moving
 - The source's data is transferred to the target without being copied

- Under what circumstances is it safe to move data from a variable?
 - When the moved-from variable is no longer required (e.g. local variable at end of scope)

- What is an rvalue reference?
 - An rvalue reference is a reference that is bound to an rvalue
 - It has a name and a type, but its memory object is not accessible by the programmer
- What are rvalue references mainly used for?
 - Rvalue references are mainly used as a syntactic device to indicate that it is safe to move data from the variable

- Write down the prototypes for the move constructor and move assignment operator of a class called Type
 - `Type(Type&&) noexcept;`
 - `Type& operator=(Type&&) noexcept;`
- Apart from using rvalue references, how do these move prototypes differ from the copy equivalents?
 - Arguments are not const references
 - Declared `noexcept`

- Why do these differences exist?
 - The operators modify their argument (by moving data from it), so they cannot be const
 - If a copy operator throws an exception, we have a partially copied-to object, which can be discarded. The copied-from object is unaffected
 - If a move operator throws an exception, we have a partially moved-to object and a partially moved-from object
 - To provide the strong exception guarantee for move operations, the move operators must not throw unhandled exceptions