hw0206 How Conversion Works?

./hw0206_example/中有範例檔案

1. float <-> int32_t

- 當有限的浮點值轉換為整數類型時，小數部分會被去除。
  ex. float 2.5 轉換成 int32_t 是 2
- 如果整數部分的值超出目標整數類型的表示範圍，則未定義。
  ex. float 10000000000.0 轉換成 int32_t 是 -2147483648
- 當整數類型的值被轉換為浮點類型時，如果該值能夠精確表示，則不會改變。
  ex. int32_t 100 轉換成 float 是 100.0
- 如果值不能精確表示，結果將是最接近的可表示值。
  ex. int32_t 16777217 轉換成 float 是 16777216.0
- 如果值超出可表示範圍，則未定義。

ISO/IEC 9899:201x
6.3.1.4 Real floating and integer

When a finite value of real floating type is converted to an integer type other than _Bool, the fractional part is discarded (i.e., the value is truncated toward zero). If the value of the integral part cannot be represented by the integer type, the behavior is undefined.

When a value of integer type is converted to a real floating type, if the value being converted can be represented exactly in the new type, it is unchanged. If the value being converted is in the range of values that can be represented but cannot be represented exactly, the result is either the nearest higher or nearest lower representable value, chosen in an implementation-defined manner. If the value being converted is outside the range of values that can be represented, the behavior is undefined. Results of some implicit conversions may be represented in greater range and precision than that required by the new type (see 6.3.1.8 and 6.8.6.4).

2. int32_t <-> uint32_t

- 如果值在新類型的範圍內，則轉換後不會改變
  ex. int32_t 100 轉換成 uint32_t 是 100
  ex. uint32_t 100 轉換成 int32_t 是 100
- 如果新類型是無號的(unsigned)，並且值超過新類型的範圍，值將反覆加或減去新類型的最大值+1，直到值在新類型的範圍內。
  ex. int32_t -1 轉換成 uint32_t 是 4294967295

ISO/IEC 9899:201x
6.3.1.3 Signed and unsigned integers

When a value with integer type is converted to another integer type other than _Bool, if the value can be represented by the new type, it is unchanged.

Otherwise, if the new type is unsigned, the value is converted by repeatedly adding or subtracting one more than the maximum value that can be represented in the new type until the value is in the range of the new type.

Otherwise, the new type is signed and the value cannot be represented in it; either the result is implementation-defined or an implementation-defined signal is raised.

3. double <-> float

- 當浮點數類型之間進行轉換時，如果值可以被新類型精確表示，則值保持不變。

  ex. double 1.5 轉換成 float 是 1.5
- 如果不能精確表示，結果是最接近的可表示值。

  ex. double 0.3 轉換成 float 大約是 0.30000001192092895508
- 如果值超出新類型的範圍，行為未定義。

  ex. double 308 轉換成 float 是 inf

ISO/IEC 9899:201x
6.3.1.5 Real floating types

When a value of real floating type is converted to a real floating type, if the value being converted can be represented exactly in the new type, it is unchanged. If the value being converted is in the range of values that can be represented but cannot be represented exactly, the result is either the nearest higher or nearest lower representable value, chosen in an implementation-defined manner. If the value being converted is outside the range of values that can be represented, the behavior is undefined. Results of some implicit conversions may be represented in greater range and precision than that required by the new type (see 6.3.1.8 and 6.8.6.4).