Topic 6: Bitwise operations

Bitwise operation

- ~!&|^<< >>
    - Usually co-work with HEX operation

```
#include <stdio.h>

int main (void)
{
    unsigned char a = 0x3F;
    unsigned char b = 0xF3;

    a = ~a;
    printf ("~a: 0x%02x \n", a);
    a = ~a;
    printf ("!a: 0x%02x \n", !a);
    printf ("a & b: 0x%02x \n", a & b);
    printf ("a | b: 0x%02x \n", a | b);
    printf ("a ^ b: 0x%02x \n", a ^ b);
    a = a << 3;
    printf ("a << 3: 0x%02x \n", a);
    b = b >> 3;
    printf ("b >> 3: 0x%02x \n", b);

    return 0;
}
```

```
~a: 0xc0
!a: 0x00
a & b: 0x33
a | b: 0xff
a ^ b: 0xcc
a << 3: 0xf8
b >> 3: 0x1e
```

Check a specific bit

```
#include <stdio.h>
typedef unsigned char Int8;

int main (void)
{
    Int8 a = 0x3F;

    if (test_bit (a, 3))
    {
        printf ("test! \n");
    }
}
```

```
Int8 test_bit (Int8 target, Int8 bit)
{
    Int8 i;

    for (i = 0 ; i < bit ; i++)
    {
        target = target >> 1;
    }
    if (target & 0x01)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}
```

Set a bit

```
void set_bit (Int8 *target, Int8 bit)
{
    Int8 i;
    Int8 set = 0x01;

    for (i = 0; i < bit ; i++)
    {
        set = set << 1;
    }
    *target = *target | set;
}
```

Clear a bit

```
void clear_bit (Int8 *target, Int8 bit)
{
    Int8 i;
    Int8 set = xxx1;

    for (i = 0; i < bit ; i++)
    {
        xxx2;
    }
    *target = *target xxx3 (xxx4);
}
```

Topic 7: Macro

Macro

- Macro → #define XX OO
  - Macro is to replace XX to be OO !
  - No matter the original type. Macro will perform translation directly
- Example
  - #define PI (3.14159)
- Use macro to do operation
  - #define ADD(x, y) x+y

Macro

```
#include <stdio.h>

#define ADD(x,y) x+y
#define Add(x,y) (x+y)

int main (void)
{
    printf ("num1: %d \n", 3*ADD(5,2));   //xx1
    printf ("num2: %d \n", 3*Add(5,2));    //xx2

    return 0;
}
```

When writing marco, you have to add parenthesis !

Macro

- Use macro to write functions

```
int main (void)
{
    unsigned char a = 0x3F;

    if (test_bit (a, 3))
        printf ("test! \n");
}
```

```
#define   test_bit(target,bit)  \
    ((target >> bit) & (0x01))
```

```
Int8 test_bit (Int8 target, Int8 bit)
{
    Int8 i;

    for (i = 0 ; i < bit ; i++)
    {
        target = target >> 1;
    }
    if (target & 0x01)
    {     return 1;    }
    else
    {     return 0;    }
}
```

Macro

```
void set_bit (Int8 *target, Int8 bit)
{
    Int8 i;
    Int8 set = 0x01;

    for (i = 0; i < bit ; i++)
    {
        set = set << 1;
    }
    *target = *target | set;
}
```

```
#define SET_BIT(target,bit) \

??????
```

Use macro to concatenate keywords

- Use to define general data structure

- `#define CONN(x, y) x##y`
  - `CONN (go,od!)` → `good!`

Macro

```
#include <stdio.h>

#define DECLARE(x)         \
typedef struct type##x {   \
   int type##x##first;     \
   int type##x##second;    \
} tType##x

#define USE(x) tType##x

DECLARE(1);
DECLARE(2);
```

```
int main (void)
{
    USE(1) type1;
    type1.type1first = 20;

    return 0;
}
```

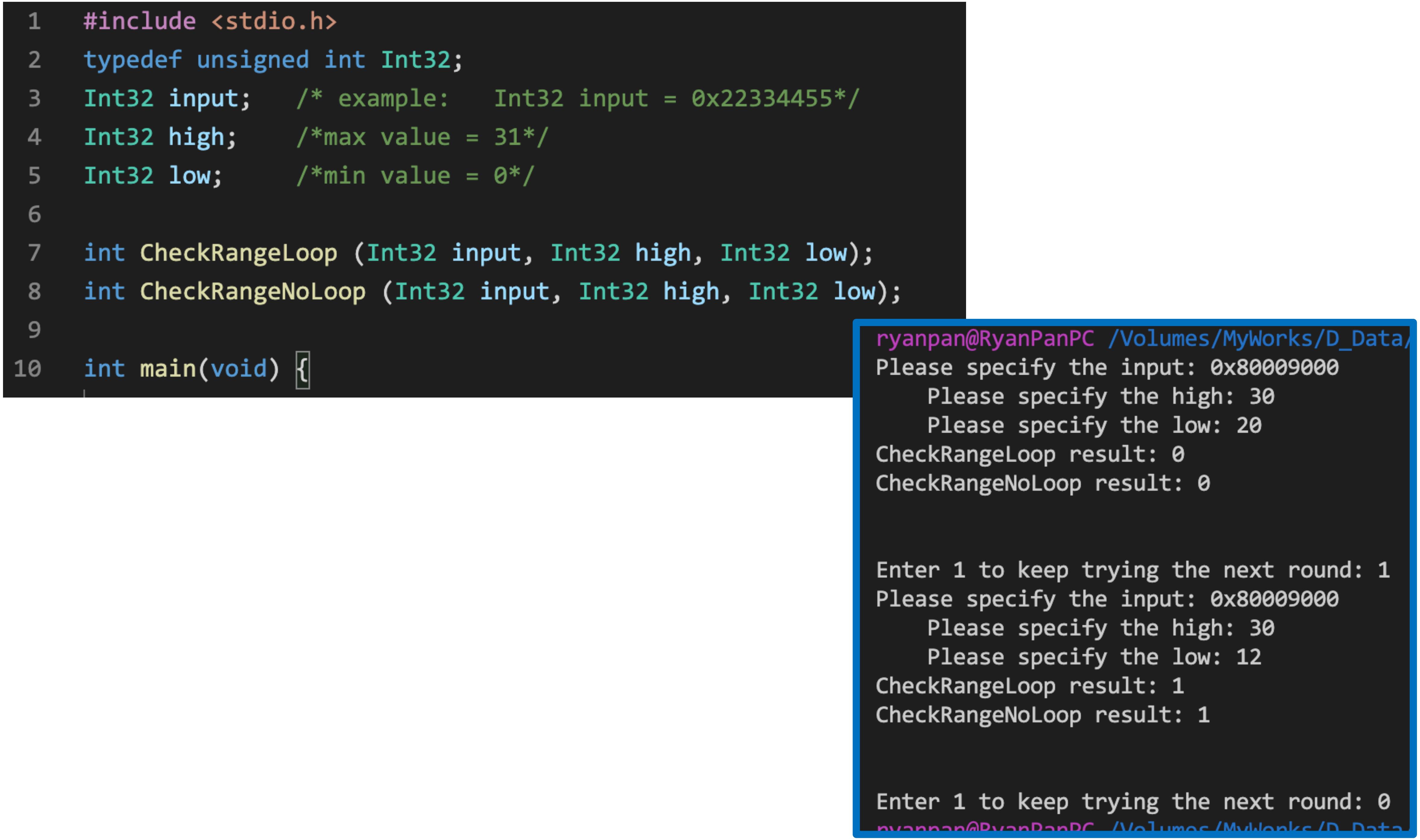
Special Macro

```
(i>=0 ? i : -i)   ← Add parenthesis!!

if (i>=0) return i;
else return –i;
```

W10-on site assignment

- Declare →
  - typedef unsigned int Int32;
  - Three global variables
    - Int32 input; /* example: Int32 input = 0x22334455*/
    - Int32 high; /*max value = 31*/
    - Int32 low; /*min value = 0*/
- Write two functions
  - int CheckRangeLoop (Int32 input, Int32 high, Int32 low); → can contain loop
  - int CheckRangeNoLoop (Int32 input, Int32 high, Int32 low); → cannot contain loop
- Check if the inputted 32 bit number have bits set to 1 between low and high
  - For example: input: 0x00000100
    - low: 0, high: 9 → return 1
    - low: 20, high: 31 → return 0
- Write a main function to call the above two functions



```c
#include <stdio.h>
typedef unsigned int Int32;
Int32 input;   /* example:   Int32 input = 0x22334455*/
Int32 high;    /*max value = 31*/
Int32 low;     /*min value = 0*/

int CheckRangeLoop (Int32 input, Int32 high, Int32 low);
int CheckRangeNoLoop (Int32 input, Int32 high, Int32 low);

int main(void) {
```

```
ryanpan@RyanPanPC /Volumes/MyWorks/D_Data/
Please specify the input: 0x80009000
    Please specify the high: 30
    Please specify the low: 20
CheckRangeLoop result: 0
CheckRangeNoLoop result: 0


Enter 1 to keep trying the next round: 1
Please specify the input: 0x80009000
    Please specify the high: 30
    Please specify the low: 12
CheckRangeLoop result: 1
CheckRangeNoLoop result: 1


Enter 1 to keep trying the next round: 0
ryanpan@RyanPanPC /Volumes/MyWorks/D_Data
```