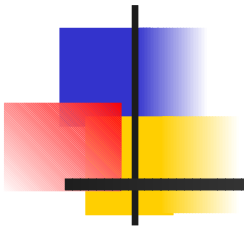
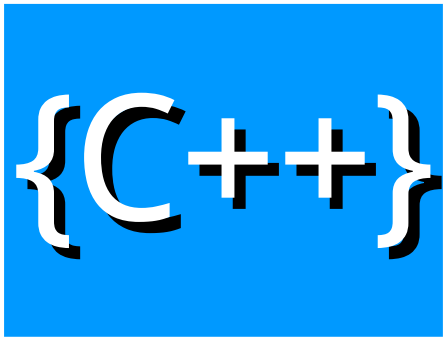


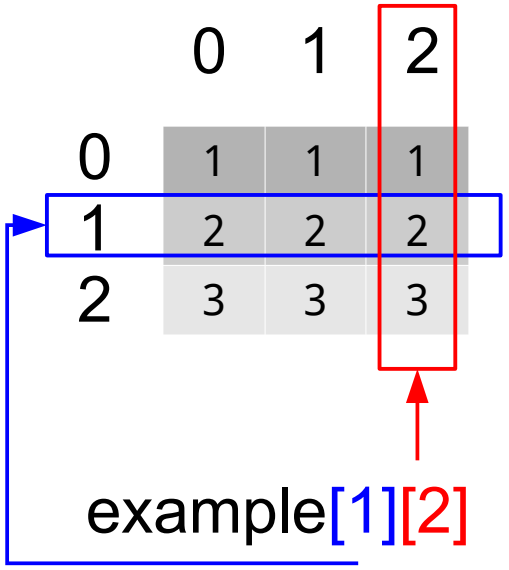
Arrays

Week 7



Yang-Cheng Chang
Yuan-Ze University
yczhang@saturn.yzu.edu.tw

Two dimension Array



Declaration and initialization

```
int example[3][3];
int example[3][3] = {{1,1,1},{2,2,2},{3,3,3}};
int example[][3] = {{1,1,1},{2,2,2},{3,3,3}};
int example[][3] = {{0},{0},{0}};
int example[3][3] = { 0 };
```

Access the array

```
for (int i = 0; i <= 2; i++){
    for(int j = 0; j <= 2; j++){
        cout << example[i][j] << " ";
    }
    cout << endl;
}
```

Generate a random number

- The include files
  - To use function srand(), should include <cstdlib>
  - To use function time(), should include <ctime>
- Should set different seed to srand() when you restart the program

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;
int main(int argc, const char *argv[])
{
    srand(time(NULL));
    cout << rand() << endl;
    return 0;
}
```

Set the seed of this random number generator

Don't set the seed in for-loop statements

- The random table will be the same and reset when you set the same seed

```
srand(0);
for (int i = 0; i < 10; i++) {
   cout << rand() << endl;
}

// The random table will be the same and reset when you set the same seed
cout << "----------------------------"<< endl;
srand(0);
for (int i = 0; i < 10; i++) {
   cout << rand() << endl;
}
```

```
// Don't set the seed in for-loop statements
for (int i = 0; i < 10; i++) {
    srand(time(NULL));
    cout << rand() << endl;
}
return 0;
```

Week 7 Assignment

- Write a program that simulates the rolling of two dice. The program should use rand to roll the first die and should use rand again to roll the second die. The sum of the two values should then be calculated.

3 + 2 = 5

Week 7 Assignment

- The following figure shows the 36 possible combinations of the two dice.

The value of Dice 1

The value of Dice 2	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

← The sum of two dices

Each die can show an integer value from 1 to 6, so the sum of the two values will vary from 2 to 12, with 7 being the most frequent sum and 2 and 12 being the least frequent sums.

Output

- Your program should roll the two dice 36,000 times. Use a two-dimensional array to tally the numbers of times each possible sum appears. Print the results in a tabular format.

	1	2	3	4	5	6
1	1031	1001	985	971	1012	1013
2	996	989	988	1032	988	1002
3	1011	984	1018	994	1026	969
4	973	989	977	997	1008	997
5	1045	1018	964	1016	1043	950
6	959	1039	1039	996	973	1007

← The times of each possible sum

Output

- Print the probability of each possible sum which vary from 2 to 12.

The probability of each possible sum

P(2) = P(1+1)= 0.026528
P(3) = P(1+2)+P(2+1)=0.027194+0.028806=0.056000
…..
P(7) = P(1+6)+P(2+5)+P(3+4)+P(4+3)+P(5+2)+P(6+1) =
0.028306+0.028111+0.028361+0.028417+0.026694+0.027333=0.167222
…..
P(12) =P(6+6)=0.027333