C++程序设计(2)

1.数组操作

- C++以行优先存储，FORTRAN以列优先
- 数组N*N数组的 value[x][y] 偏移地址计算：x*N+y
- 一维数组快于多维数组

二维数组

格式2：

类型 数组名 [常量表达式1] [常量表达式2]

={ {初值表1} , {初值表2} ……}

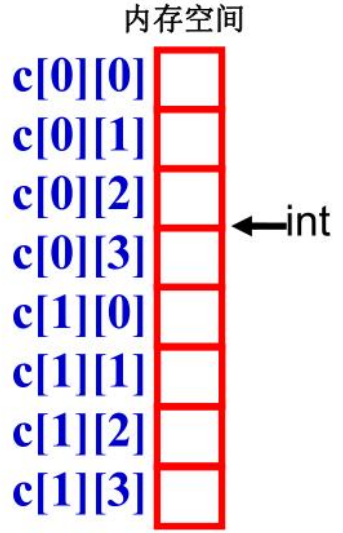
例

int c[2] [4]={ {1，3，5，7}，{2，4，6，8} };

右键有惊喜

1 c[0][0]	3 c[0][1]	5 c[0][2]	7 c[0][3]
2 c[1][0]	4 c[1][1]	6 c[1][2]	8 c[1][3]

二维数组在内存空间中存储



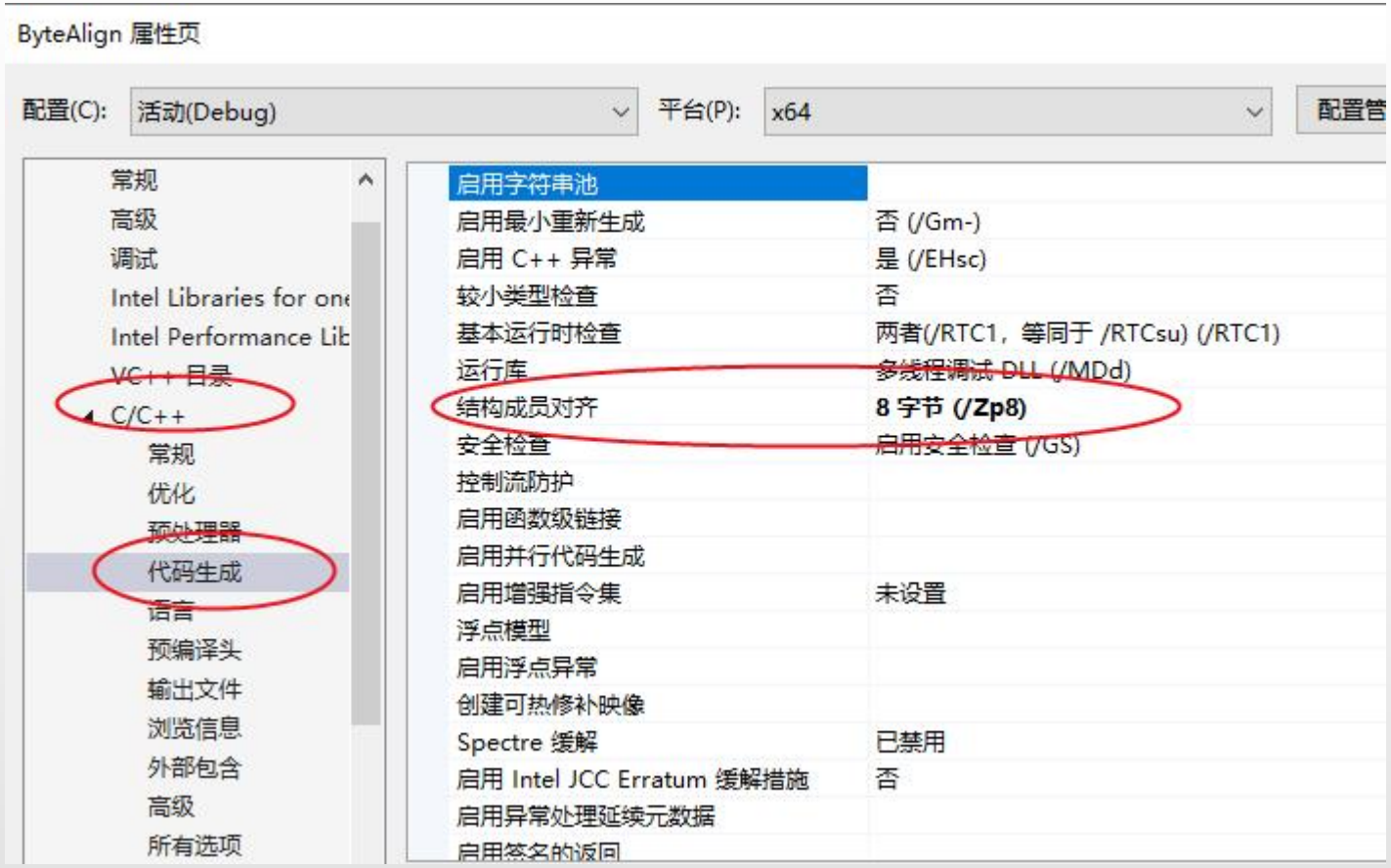
下标顺序

```
for (int i = 0; i < NUMSIZE; i++)
{
    for (int j = 0; j < NUMSIZE; j++)
    {
        value[i][j] = 1.0;
    }
}
```

```
for (int i = 0; i < NUMSIZE; i++)
{
    for (int j = 0; j < NUMSIZE; j++)
    {
        value[j][i] = 1.0;
    }
}
```

NUMSIZE = 10000， 效果是否一样？

2.字节对齐



#pragma pack()

3.函数调用开销

哪种调用方式好，是否有改进的余地?

```
double GetN3(double x)
{
    double val = pow(x,3);
    return val;

    //
    double val = x * x * x;
}
```

4.内存管理

- 智能指针使用 unique_ptr（代码规范）
- 单一数据原则
- 指针数据原则上提供深拷贝函数
- 使用时分配，使用完释放
- 清楚指针生命周期和所有权（code review）
- 内存泄漏检查工具 VLD

5.矩阵类数据

```
class xMatrix
{
public:
    double getValue(int xIndex, int yIndex);
private:
    DataType value;
}
```

double 型矩阵数据存储，DataType 使用下列哪种合适?

1. double value[N][N];
2. std::map<std::pair<int, int>, double> value;
3. double* value;
4. std::vector<double> value;

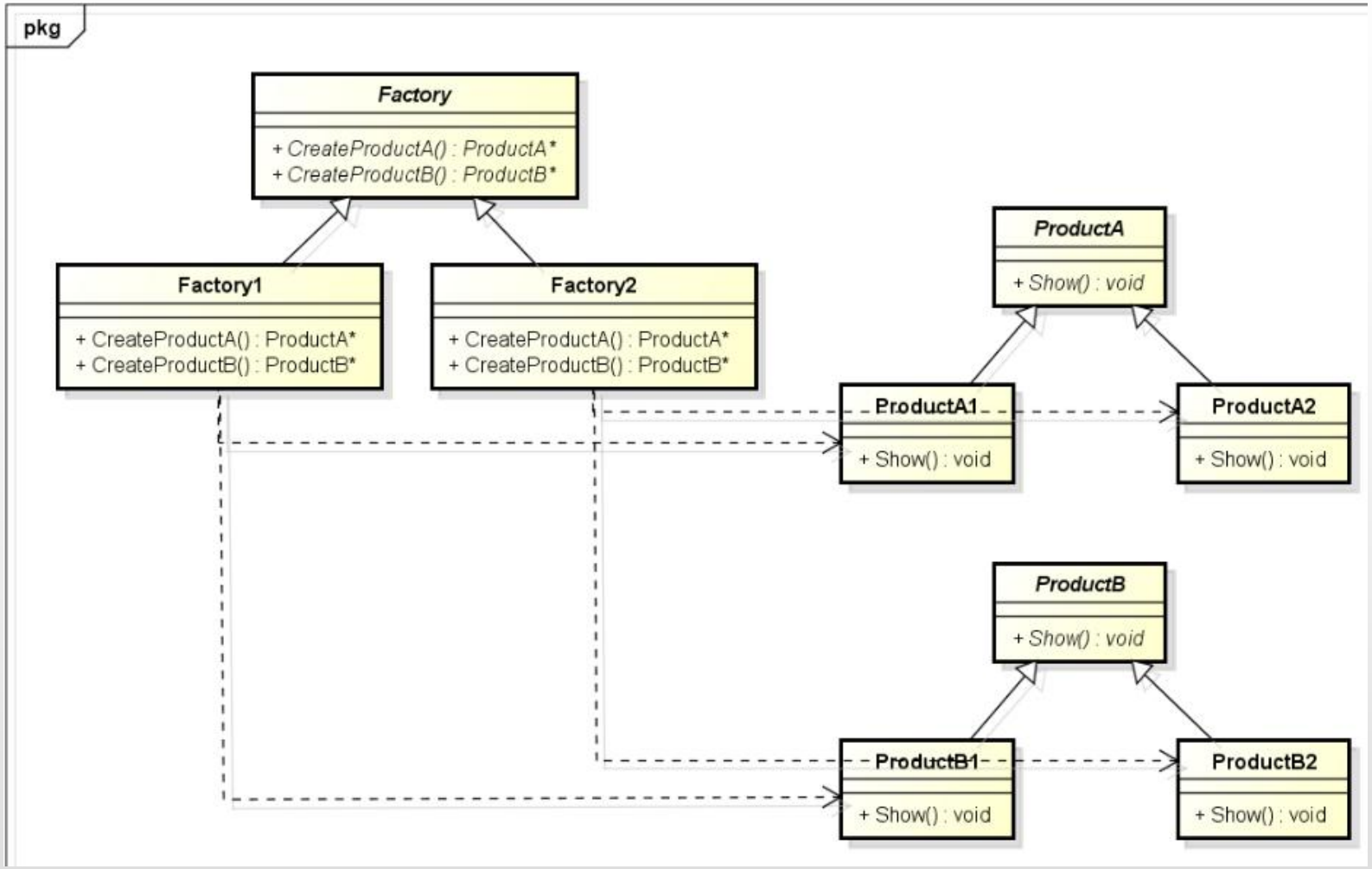
6.if..else和switch 替换

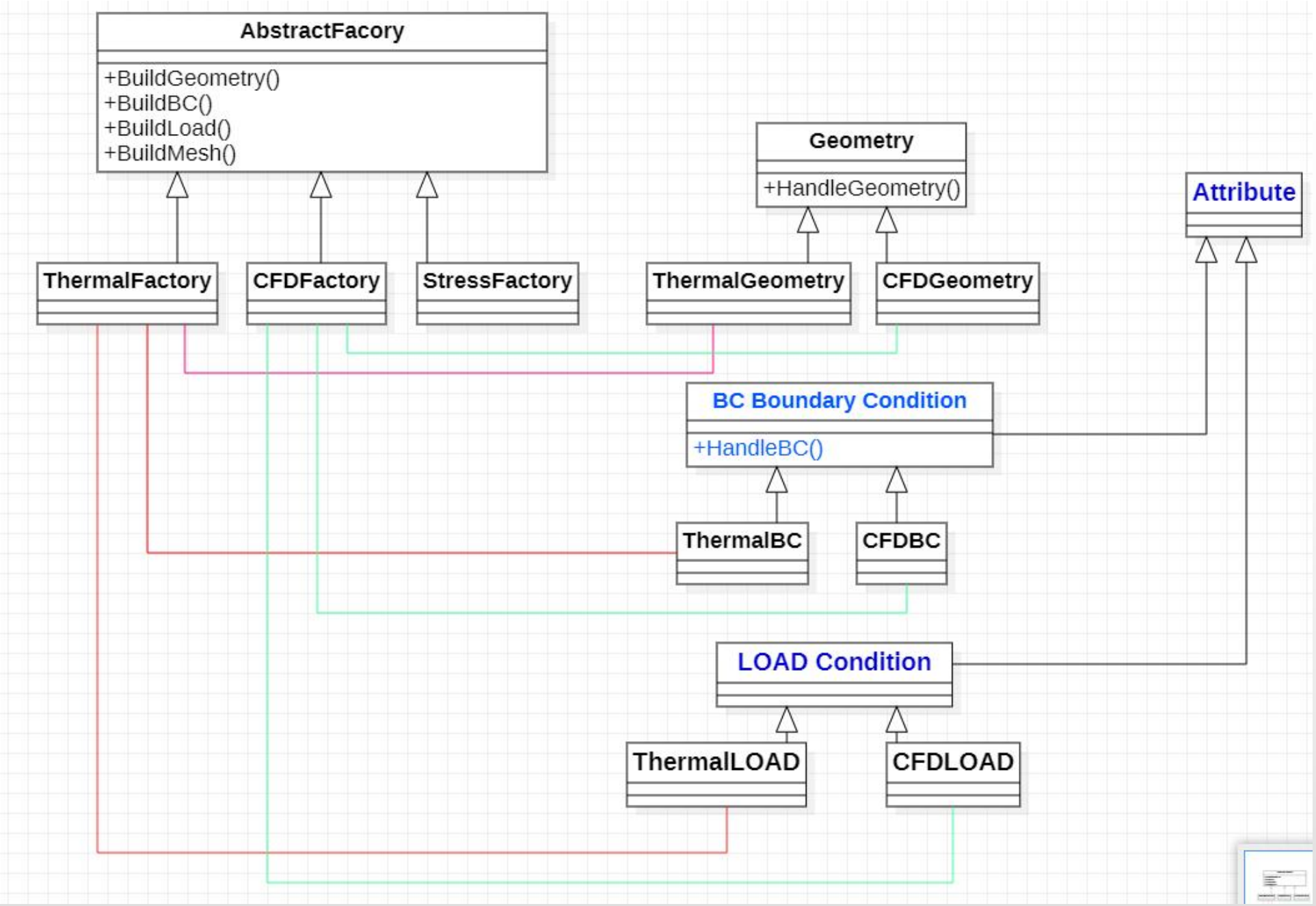
```cpp
#include <map>
#include <string>
void Demo() {
    int x = 4;
    if (x == 4)
    {
        std::cout << "Four" << std::endl;
    }
    else if (x == 5)
    {
        std::cout << "Five" << std::endl;
    }

    std::map<int, std::string> val;
    val[4] = "Four";
    val[5] = "Five";
}
```

2.工厂模式替换

工厂和抽象工厂


pkg
Factory
+ CreateProductA() : ProductA*
+ CreateProductB() : ProductB*
Factory1
+ CreateProductA() : ProductA*
+ CreateProductB() : ProductB*
Factory2
+ CreateProductA() : ProductA*
+ CreateProductB() : ProductB*
ProductA
+ Show() : void
ProductA1
+ Show() : void
ProductA2
+ Show() : void
ProductB
+ Show() : void
ProductB1
+ Show() : void
ProductB2
+ Show() : void


AbstractFacory
+BuildGeometry()
+BuildBC()
+BuildLoad()
+BuildMesh()
Geometry
+HandleGeometry()
Attribute
ThermalFactory
CFDFactory
StressFactory
ThermalGeometry
CFDGeometry
BC Boundary Condition
+HandleBC()
ThermalBC
CFDBC
LOAD Condition
ThermalLOAD
CFDLOAD

Next meeting topics

Q&A