# 强大的string

两大新神器

std::to\_string与std::stoi;

Int转化为string与string转化为int

注：std::stoi(“20 words”) 正确，转换为20

Std::stoi(“words 20”)错误

Push\_back(char c)

若a.push\_back(int ),则会加入一个奇怪的形状（1为正方形）

# 2.创建vector的二位数组

1. vector<vector<int>> array;*//定义二维数组*
2. vector<int>v;*//定义一维数组*
3. array.clear();*//将二维数组清空，或初始化（初始化代码很简单，不会可以百度）//也可不用这一步*
4. int temp = 0;
5. for (int i = 0; i < r; i++)*//输入r\*c的二维数组*
6. {
7. v.clear();*//子数组返回时要清除*
8. for (int j = 0; j < c; j++)
9. {
10. cin >> temp;
11. v.push\_back(temp);
12. }
13. array.push\_back(v);
14. }

# 3.new一个二位数组

const int M = 10, N = 5;//10行5列。  
int \*\* a;  
a = new int \*[M];  
for(int i = 0; i < M; i ++)  
a[i] = new int[N];  
所得到的a就是10行5列的数组。  
使用后需要释放，代码为

for(int i = 0; i < M; i ++)  
delete [] a[i];  
delete []a;

# 4指针与指针变量

指针的本意就是**内存地址**，我们可以通俗地理解成内存编号，既然计算机通过编号来操作内存单元，这就造就了指针的**高效率**。

那么什么是指针变量呢？指针变量可通俗地理解成存储指针的变量，也就是**存储内存地址（内存编号）的变量**。

对于一个32位的操作系统来说，内存空间中每一个字节的编号是一个32位二进制数，所以内存编号从0000 0000 0000 0000 0000 0000 0000 0000至1111 1111 1111 1111 1111 1111 1111 1111，转换成16进制也就是0x00000000至0xFFFFFFFF，所以指针变量占4个字节

# 5.size\_t与wchar\_t

size\_t是标准C库中定义的，在64位系统中为long long unsigned int，非64位系统中为long unsigned int。C语言中，此类型位于头文件[stddef.h](https://baike.baidu.com/item/stddef.h)中，而在C++中，则位于cstddef中。通常我们用sizeof(XXX)操作，这个操作所得到的结果就是size\_t类型。

wchar\_t就是wide char type，“一种用来记录一个宽字符的数据类型”。

wchar\_t wc=L’a’;//wc保存了宽字符a，占两个字节

//L‘a’表示宽字符a

# 6.int\* a,b;

与int \*a,b;同等，表示int \*a;int b;

# 7.栈和堆的区别

**一、程序的内存分配方式不同**

* 栈区（stack）：**编译器自动分配释放**，存放函数的参数值，局部变量的值等，其操作方式类似于数据结构的栈。
* 堆区（heap）：**一般是由程序员分配释放**，若程序员不释放的话，程序结束时可能由OS回收，值得注意的是他与数据结构的堆是两回事，分配方式倒是类似于数据结构的链表。

**二、申请方式不同**

* stack 由系统自动分配，heap 需要程序员自己申请。
* C 中用函数 malloc分配空间，用 free 释放，C++用 new 分配，用 delete 释放。