找工作找了也有一段时间了，期间参加的笔试也不少，由此借最近比较得闲之际，将整理出有关 C++在笔试中常考到的知识点，这些都是我遇到的一些笔试题目或者是我自己以前搜集整理的资料，现在拿出来希望可以给大家一些帮助。如果有什么错误之处，请各位指出，谢谢！

1. C++跟 C 语言的异同之处？

答：C++语言来源于 C 语言。

同：规则简单，具有数据表示运算功能，可以直接对内存操作，运行效率高，

异：C 语言缺乏数据类型检查机制，代码重用性差；C++强化了数据的类型检查和语句的结构性，增加了面向对象的程序设计的支持。

2. 请问 C++的类和 C 里面的 struct 有什么区别？

答：struct 的成员的默认访问说明符为 Public，而 class 的成员的默认访问说明符为 Private。其他没有区别

3. 列举关键字 static 在 C++中的作用。

a) 在函数体，一个被声明为静态的变量在这一函数被调用过程中维持其值不变。

b) 在模块内（但在函数体外），一个被声明为静态的变量可以被模块内所用函数访问，但不能被模块外其它函数访问。它是一个本地的全局变量。

c) 在模块内，一个被声明为静态的函数只可被这一模块内的其它函数调用。那就是，这个函数被限制在声明它的模块的本地范围内使用。

4. 编程题: 用最有效率的方法算出 2 乘以 8 等於几? ————2 << 3，2 向左移 3 位。

5. 简述逻辑操作(&，|，^)与条件操作(&&，||)的区别。

答：区别主要在两点：

a) 条件操作只能操作布尔型的，而逻辑操作可以操作布尔型和数值型。

b) 逻辑操作不会产生短路。

6. &的意义：包括取址和引用。两者的区别如下所示：

```
int a;
int &pa;
int &ra=a; //ra 是 a 的别名，只能在定义时初始化
pa=&a; //pa 指向 a，这里的&是取址。
```

7. 关于指针变量

答：

- 变量的地址是一个十六进制整数。
- 能够存放地址的变量称为“指针类型变量”，简称“指针变量”。
- 计算机的 CPU 决定了内存寻址方式，所以不管指针所指的对象是什么类型，指针值本身的规格都是一样，例如，16 位或者 32 位的整数。关联类型的作用是控制对象的访问。如果关联类型为 int，通过指针变量访问对象时，读取从指针值指示的位置开始的连续 4 个字节，并按整型数据解释。
- Void *vp，void 指针变量能够存放任意内存地址。因为没有关联类型，编译器无法解释所指对象，程序中必须对其进行强制类型转换，才可以按指定类型数据使用。

8. 关于内存方面

答：

a) 关于内存的分配

内存单元由操作系统按字节编号，称为地址。一个变量的地址值是系统分配的，不能由高级程序设计语言决定，但 C++可以知道分配后的地址值。

b) 关于内存对齐的问题以及 sizof()的输出

编译器自动对齐的原因：为了提高程序的性能，数据结构（尤其是栈）应该尽可能地在自然边界上对齐。原因在于，为了访问未对齐的内存，处理器需要作两次内存访问；然而，对齐的内存访问仅需要一次访问。

c) 关于内存的访问

在内存建立一个对象之后，可以用名方式访问 和地址方式访问。

- 名方式访问：操作对象的内容，也叫直接访问。访问形式分为“读”和“写”。
- 地址方式访问：一个对象的地址用于指示对象的存储位置，称为对象的“指针”。指针所指的物理存储空间称为“指针所指对象”。通过地址访问对象又称为“指针访问”。指针访问运算符“*”，取址运算符是“&”。

9. 请讲一讲析构函数和虚函数的用法和作用。

答：析构函数是在对象生存期结束时自动调用的函数，用来释放在构造函数分配的内存。虚函数是指被关键字 virtual 说明的函数，作用是使用 C++语言的多态特性

10. 关于关键字 extern 和 Static 的说明：

答：

a) 关键字 extern 和 static 声明静态存储变量和函数标识符。全局声明的标识符默认为 extern。

b) 当这两个关键字用于说明变量时，程序开始执行时就分配和初始化存储空间，用于说明函数，表示从程序执行开始就存在这个函数名。

c) 用 static 声明的局部变量只能在定义该变量的函数体中使用。

d) 与自动变量不同的是，static 在第一次使用时进行初始化（默认初始化值为 0）。函数退出时，系统保持其存储空间和数值，下次调用时，static 变量还是上次退出函数时的值。

e) 程序的其他文件也能够访问全局变量，但必须在使用该全局变量的每个文件中用关键字 extern 予以声明。

11. 关于变量与常量：

答：变量的赋值运算是在程序运行时才执行的操作。常量定义在编译阶段就被确定下来。

12. 关于数组元素：

a) 只有定义静态数组，C++才会自动把各元素值初始化为 0。（static）

b) 数组由 CONST 约束为常量的，必须在定义的时候初始化，不能在程序代码中对它的元素重新赋值。

13. 关于全局变量和局部变量：

答：具有文件作用域的变量称为全局变量。具有函数作用域或块作用域的变量称为局部变量。全局变量声明默认初始值为 0。当局部变量和全局变量同名，在块内屏蔽全局变量。在块内访问全局变量，可以用域运算符“：：”。

14. 关于函数参数传递问题。

答：

a) 参数的定义：参数是调用函数与被调用函数之间交换数据的通道。函数定义首部的参数称为形式参数（形参），调用函数时使用的参数称为实际参数（实参）。实参必须与形参在类型，个数，位置上对应。

b) 函数调用前，形参没有存储空间。函数被调用时，系统建立与实参对应的形参的存储空间，函数通过形参与实参通信，进行操作。函数执行完毕，系统收回作为形参的临时存储空间。这个过程称为参数传递或参数的虚实结合。

c) C++有 3 种参数传递机制：值传递（值通用），指针传递（地址调用）和引用传递（引用调用）。实参和形参按照不同传递机制进行通信。

d) 三种传参方式的比较：

①□ 传值参数：在值传递机智中，作为实参的表达式的值被复制到由对应的形参名所标识的一个对象中作为形参的初试值。函数体对形参的访问，修改都是在这个标识对象上操作，与实参无关。

②□ 指针参数：函数定义中的形参被说明为指针类型时，形参指针对应的实际参数是地址表达式，即对象的指针。调用函数时，实参把对象的地址值赋给形式参数名标识的指针变量，被调用函数可以在函数体通过形式参数指针间接访问实参所指对象。这种参数传递方式称为指针传递或地址调用。特别注意的是：指针函数不能返回局部变量的指针。例如：

```
int *f()
{
    int temp;
    //……
    return &temp;//在函数结束之后，temp 已经被释放，失去了意义。
}
```

③□ 引用参数：当 C++函数的形参被定义为引用类型，称为引用参数。引用参数对应的实参应该是对象名。函数被调用时，形参不需要开辟新的存储空间，形式参数名作为引用（别名）绑定于实参标识的对象。执行函数体时，对形参的操作就是对实参对象操作。直到函数执行结束，撤消引用绑定。

15. 关于默认参数的规定说明：

答：函数的形式参数说明中可以设置一个或多个实参的默认值，默认参数必须是函数参数中的最右边的参数。调用具有多个默认参数的函数时，如果省略的参数不是参数表中最右边的参数，则该参数右边的所有参数也必须省略。

16. 全局变量和局部变量有什么区别？是怎么实现的？操作系统和编译器是怎么知道的？

答：一些变量在整个程序中都是可见的，它们称为全局变量。一些变量只能在一个函数中可知，称为局部变量。这就是他们的区别。

在任何函数外面定义的变量就是全局变量，在函数内部定义的变量是局部变量，这是它们在程序中的实现过程。操作系统和编译器是根据程序运行的内存区域知道他们的，程序的全局数据放在所分配内存的全局数据区，程序的局部数据放在栈区。

17. 函数模板与类模板有什么区别？

答：函数模板的实例化是由编译程序在处理函数调用时自动完成的，而类模板的实例化必须由程序员在程序中显式地指定。

18. MFC 中 CString 是类型安全类么？

答：不是，其它数据类型转换到 CString 可以使用 CString 的成员函数 Format 来转换

19. C++中为什么用模板类？

答： (1) 可用来创建动态增长和减小的数据结构

（2）它是类型无关的，因此具有很高的可复用性。

（3）它在编译时而不是运行时检查数据类型，保证了类型安全

（4）它是平台无关的，可移植性

（5）可用于基本数据类型

20. C++中什么数据分配在栈或堆中，New 分配数据是在近堆还是远堆中？

答：栈: 存放局部变量，函数调用参数,函数返回值，函数返回地址。由系统管理

堆: 程序运行时动态申请，new 和 malloc 申请的内存就在堆上

21. heap 和 stack 有什么区别？（堆和栈有什么区别？）

答：栈是一种线形集合，其添加和删除元素的操作应在同一段完成。栈按照后进先出的方式进行处理。堆是栈的一个组成元素

22. 关于宏定义指令：

答：宏定义指令用来指定正文替换程序中出现的标识符。形式：#define 标识符 文本

在 C 语言中，不带参数#DEFINE 常用于定义常量，带参数#DEFINE 用于定义简单函数。由于该指令是在程序正式编译前执行，所以不能对替换内容进行语法检查。C++的关键字 CONST 定义常量和 INLINE 定义的内联函数代替了#DEFINE 的作用。#DEFINE 的一个有效应用是在条件编译指令中。