小米一面：

1.自我介绍，平常怎样学习，最近看的一本书是什么书

2.C语言程序执行的流程（链接具体链接那些文件）

3.static在C语言中的作用以及在C++中的作用

4.栈的的用途，有哪些应用场景

5.C++（C11）的新特性（说了智能指针和模板类以及nullptr和NULL的区别）的实现

6.虚函数怎样实现的（可以用父类的指针访问子类自己定义的虚函数吗）

7.说说多线程的理解以及怎样用，线程间通信以及两个线程的消费者生产者模型

8.说一说网络中出现多个连接的请求怎样处理（select，poll，epoll的区别）

9.具体的问题：有一个公司，人员稳定有20人，设置门禁卡，用C语言，不适应库函数（只需要刷开门就可以，不用其他复杂的操作），最简单的实现（数组）；问题升级，可以使用C++和库函数，人员扩招，还是简单的门禁卡，需要怎样设计（set）；问题继续升级，除了刷开门还要显示个人信息该使用怎样的数据结构（map），人员有可能离职或者新增，那么假如有工号为10000的员工，需要查找几次?

10.有什么想要问我的吗?

小米二面：

1.自我介绍

2.Linux的相关命令：查找一个单词在十个文件中，快速定位到代码出错的行，怎样快速跳到第一行，Linux下用过哪些软件，用过什么IDE?

3.知道哪些预处理指令，都有什么作用，这些指令有什么共同点?

4.怎样计算一个结构体的大小，怎样设置默认对齐

sizeof(结构体名)，或者手动计算（考虑内存对齐问题）；#pragma pack(4)，设置对齐数为4，#pragma pack()，恢复编译器的默认对齐数

5.说一说栈的一些用途

（1）保存上下文（操作系统处理中断的时候，函数递归调用）

（2）传递参数（C语言进行函数调用的时候函数的参数使用寄存器进行传递，当参数多的时候就可以用栈来传递）

（3）临时变量的保存（非静态的局部变量以及编译器自动生成的临时变量）

（4）括号的匹配以及加减乘除运算时进行优先运算

6.说一说队列的用途

（1）操作系统的调度算法

（2）生产者消费者模型

（3）进程间通信（消息队列，匿名管道）

7.说一说STL的容器以及智能指针

容器：

（1）vector（向量）：是一种序列式容器，事实上和数组差不多，但它比数组更优越。一般来说数组不能动态拓展，因此在程序运行的时候不是浪费内存，就是造成越界。而

vector 正好弥补了这个缺陷，它的特征是相当于可拓展的数组（动态数组），它的随机访问快，在中间插入和删除慢，但在末端插入和删除快。

（2）List 由双向链表（doubly linked list）实现而成，元素也存放在堆中，每个元素都是放在一块内存中，他的内存空间可以是不连续的，通过指针来进行数据的访问，这个特点使得它的随机存取变得非常没有效率，因此它没有提供 [] 操作符的重载。但是由于链表的特点，它可以很有效率的支持任意地方的插入和删除操作。

（3）deque（double-ended queue）是由一段一段的定量连续空间构成。一旦要在 deque 的前端和尾端增加新空间，便配置一段定量连续空间，串在整个 deque 的头端或尾端。因此不论在尾部或头部安插元素都十分迅速。 在中间部分安插元素则比较费时，因为必须移动其它元素。deque 的最大任务就是在这些分段的连续空间上，维护其整体连续的假象，并提供随机存取的接口。

（4）set（集合）由红黑树实现，其内部元素依据其值自动排序，每个元素值只能出现一次，不允许重复。

（5）map 由红黑树实现，其元素都是 “键值/实值” 所形成的一个对组（key/value pairs)。每个元素有一个键，是排序准则的基础。每一个键只能出现一次，不允许重复。map 主要用于资料一对一映射的情况，map 内部自建一颗红黑树，这颗树具有对数据自动排序的功能，所以在 map 内部所有的数据都是有序的。

智能指针：

RAII是一种利用对象生命周期来控制程序资源（如内存、文件句柄、网络连接、互斥量等等）的简单技术。在对象构造时获取资源，接着控制对资源的访问使之在对象的生命周期内始终保持有效，最后在对象析构的时候释放资源。借此，我们实际上把管理一份资源的责任托管给了一个对象。这种做法有两大好处：

不需要显式地释放资源。

采用这种方式，对象所需的资源在其生命期内始终保持有效。

（1）auto_ptr：采用了管理权限转移，一旦有一个新的对象拷贝了原来对象的资源，那么原来的对象将被悬空，释放掉自身管理的资源，程序使用原来对象指针访问资源的时候会崩溃

（2）unique_ptr：对拷贝构造函数和赋值运算符重载只申明不实现或者申明成私有，所以unique_ptr防止拷贝。

（3）shared_ptr：通过引用计数的方式实现多个shared_ptr对象之间的资源共享；shared_ptr内部为每个对象维护者一份计数，记录着该资源被几个对象共享，当对象被销毁时（调用析构函数时）说明自己不需要该资源了，引用计数-1，当引用计数为0时就说明自己是最后一个使用该资源的，就必须释放掉该资源，如果不为0，说明还有其他的对象在使用该资源，就不能释放，否则会造成野指针；shared_ptr可能会产生线程安全问题（++，--时），所以底层实现是加锁的；shared_ptr可能会产生循环引用问题（即双向链表中），所以引入weak_ptr弱引用来解决循环引用的问题。

8.说一说一个单链表的逆置，不能开辟额外的空间

```
ListNode* ListNodeReverse(ListNode* Head)
{
 if(Head==nullptr && Head->next==nullptr)
 {
```

```
return Head;
}
else
{
ListNode* cur=Head;
 ListNode* pre=nullptr;
 while(cur!=nullptr)
 {
 ListNode* p=cur;
 cur=cur->next;
 p->next=pre;
 pre=p;
 }
 }
 return pre;
}
```

9.说一说操作系统的进程调度算法

(1)先到先服务（FCFS）按照进程到达的顺序进行服务，非抢占式执行，相当于队列的先进

(2)最短进程优先（SPN）也是非抢占式执行，这种最短进程优先也不是完全的最短进程优先，会首先服务最先到达的，接下来服务等待队列中最短进程，相当于优先队列

(3)最短剩余时间优先（SRT）或者最短作业优先（SJF）这种算法是SPN的升级算法，抢占式执行的调度算法，如果等待的队列中有作业时间比当前正在执行的进程剩余作业时间短的，那么就先执行作业时间短的进程，绝对的最短进程优先服务，同样作业时间的进程是抢占式执行

(4)轮转（RR）也称时间片轮转技术，轮转最重要的是时间片长度，以一个周期产生中断，当中断发生时，当前进程置于就绪队列尾端，从就绪队列队首拿出一个作业继续一个周期的执行（相当于FCFS）。

(5)高响应比优先（HRRN）是一种兼容了FCFS和SPN调度算法的算法，每次进行下一个作业调度时先计算等待队列中每个作业的响应比（响应比=（服务时间+等待时间）/服务时间），每次取出响应比最高的作业进行服务；这种算法调度算法克服了饥饿状态，兼顾了长作业。

10.说一说生产者消费者模型

11.说一说单例模式

```
//懒汉模式（线程安全）内嵌垃圾回收
class Singleton
{
public:
```

```
5  static Singleton* GetIncestance()
6  {
7  if(_a==nullptr)
8  {
9  _mutex.lock();
10  if(_a==nullptr)
11  {
12  _a=new Singleton();
13  }
14  _mutex.unlock();
15  }
16  return _a;
17  }
18  class Garbage
19  {
20  public:
21  ~Garbage()
22  {
23  if(Singleton::_a!=nullptr)
24  delete Singkleton::_a;
25  }
26  };
27  Garbage ga;
28  private:
29  Singleton(){}
30  Singleton(const Singleton&)=delete;
31  Singleton& operator=(const Singleton&)=delete;
32  static Singleton* _a;
33  static mutex _mutex;
34 };
35 Singleton::Garbage ga;
36 Singleton* Singleton::_a=nullptr;
37 mutex Singleton::_mutex;
38 //饿汉模式
39 class Singleton
40 {
41 public:
42  static Singleton* GetIncetance()
43  {
44  return &_a;
```

```
45   }
46   private:
47   Sinleton(){}
48   Singleton(const Singleton&)=delete;
49   Singleton& operator=(const Singleton&)=delete;
50   static Singleton _a;
51  }
52 Singleton Singleton::_a;
```

12.说一说操作系统的组成

（1）进程管理

（2）内存管理

（3）文件系统

（4）网络通讯

（5）安全机制

（6）驱动程序

（7）用户界面

13.对自己这次面试表现的评价以及平时都看什么书籍

14.如果分配给你一个任务，需要用到你不熟悉的技术，你怎样去解决任务

15.帮我总结这次面试哪些回答的有问题

CVTE一面：

1.简单的自我介绍

2.介绍一下自己的项目

3.说一下STL的容器都有哪些

4.说一下TCP协议和UDP协议的区别

5.怎样保证UDP的可靠性

6.输入一个URL之后会发生什么事

7.map和unordered_map的区别

8.说一说粘包问题，怎样预防粘包问题

9.有什么想问我的吗