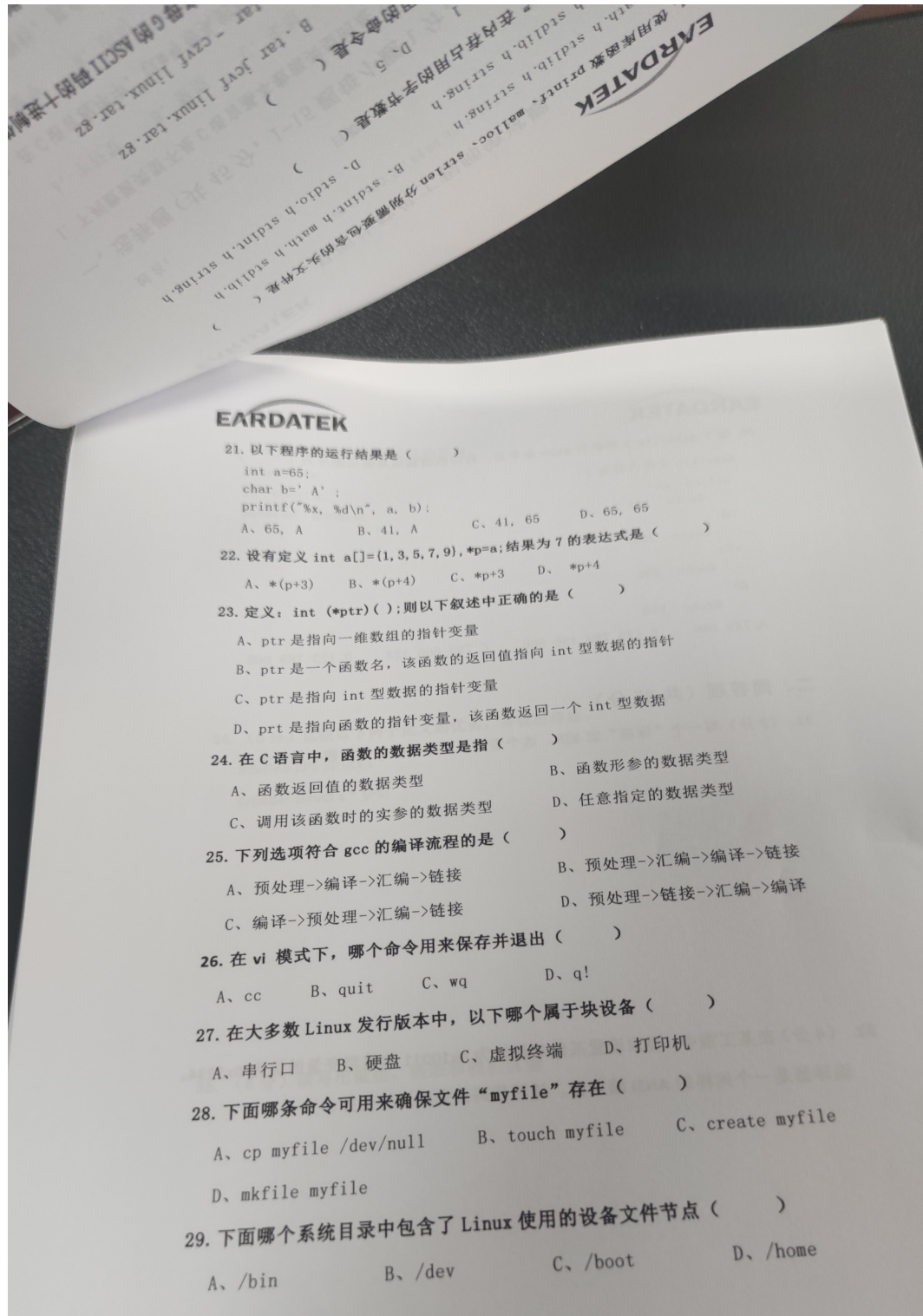


嵌入式软件工程师笔试面试经验分享(应届毕业生)

发布: 2022年7月6日 16:44:44

一、笔试部分

(一) 技术测试题 (拍了部分内容)



第三部分

- 24、 弗雷德、麦克、约翰三个人各有两部不同的汽车，其中一个人没有福特，麦克是唯一拥有法拉利的人，约翰有一部福特，而弗雷德和麦克则分别拥有一部别克。

问：谁可能拥有一辆劳斯莱斯？（ ）

- A 弗雷德 B 麦克 C 约翰

- 25、 一家博物馆里，你可以在水彩画下面的那层楼里找到现代雕塑，希腊瓷器画在油画那层楼的上面，顶层则有意大利展品，水彩画和南美艺术放在同一层楼上，而油画在现代雕塑下面的那层楼里。

(1) 在最下面那层楼里能找到什么？（ ）

- A 现代艺术 B 水彩画 C 希腊瓷器 D 油画 E 意大利展品 F 南美艺术

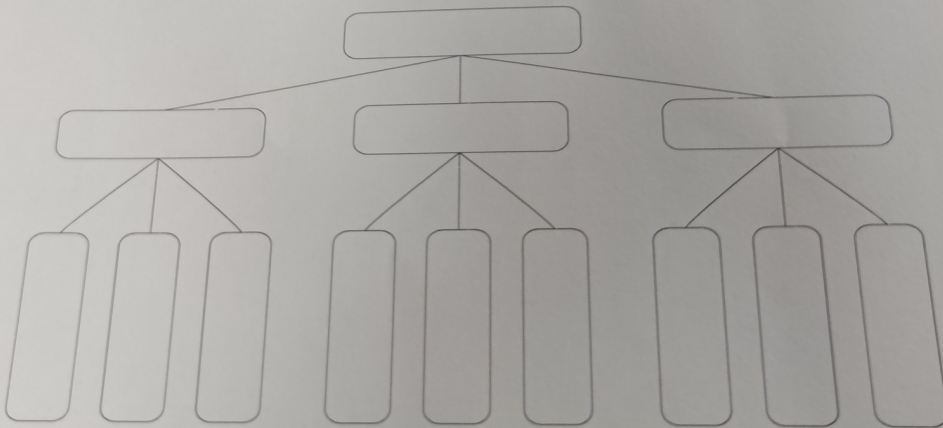
(2) 哪两种展览放在同一层？（ ）

- A 现代雕塑和希腊瓷器 B 水彩画和油画 C 油画和南美艺术 D 以上没有一个
是正确答案

- 26、 请用金字塔结构排列问题，并完成论点

题目：卖旧车前的准备

- | | |
|------------|--------------|
| 1) 把锈斑喷漆 | 6) 写后窗售车广告 |
| 2) 买新的车内地毯 | 7) 写公告板的售车海报 |
| 3) 写报纸广告稿 | 8) 修理车顶的破洞 |
| 4) 买新的顶灯 | 9) 将车内吸尘 |
| 5) 修理车身凹陷处 | |



姓名:

日期:

分数:

第一部分

请在每小题中圈上最符合你情况的选项(答案没有正确错误之分,仅作为参考)

1、你更关心或更重视:

A 人的感情

B 人的权利

2、当你必须会见一个陌生人时,你觉得:

A 这是很费力的事

B 这是愉快的,至少不费力的事

3、常与 A 还是与 B 类人相处的好些?

A 富于想象力的人

B 讲究实际的人或现实的人

4、和别人在一起时很自然地显得:

A 文静、不多说话

B 善于交际

5、以下两种人,你认为哪一种人更值得人尊重?

A 感情真挚的人

B 一贯负责的人

6、你认为你自己:

A 比一般人更加热情

B 不像一些人那样激动或兴奋

7、你和许多人一起工作时喜欢:

A 按既定的方式办事

B 自己想办法去做

8、以下两点,你对哪一点更恼火?

二、面试部分

(一) 技术面试题

面试百问:

问:你自己做的一个项目,问你在做项目的过程中遇到的困难以及怎么解决

问:你项目中用到的数据结构,单片机项目我只用了数组,面试官没有继续

问:单片机的资源大小。

问:通信方式: IIC SPI ,时序原理以及区别

问:网络协议: 建立连接的步骤 以及用的函数

问:怎么同时接入多个客户端

问:进程与线程的区别

问:进程的通信方式

问:使用共享内存的注意事项

问:如果一个进程长时间占用共享内存

问:有没有具体写的程序或者项目

问:为什么要学习Linux系统

问:有没有移植过Linux系统

问:移植中的问题以及解决方法

问:如果在公司做的项目要用其他语言去编写愿不愿意学

问:你了解过我们单位吗?

问:你倾向于做公司哪一类项目

答:智能硬件

问:你觉得智能硬件用到了什么关键技术

答:(这个问题有点回答不好,不理解)

问:对WiFi与蓝牙了解多少,觉得这些技术

答:用过esp8266以及蓝牙模块,并用这些实现了一些功能

问:电子大赛3个人你是组长吗? 你觉得团队管理方面有什么要注意的

问：偏向于技术线还是管理线？

答：技术线，我技术方面更有优势

问：你觉得是上级分配任务还是自己做一些任务？

问：职业规划：

答：先学习，然后深入研究

问：公司的哪些方面想学习

问：公司给你一个你没有做过的项目，你会怎么入手

答：先从网上学习各个功能模块资料

问：如果实在网上也没有资料，自己没能力做怎么办？

答：向上级请示自己能力不足，需要更多的工程师完成这个任务。或者凭自己的能力先去试一试，能做出多少先做多少。

问：你有什么问题？

答：工作环境、是否是凭自己能力拿工资

（二）人事面试

（很多记不清了）只记得：

问：薪资情况、家庭情况、个人阅历、

答：期望实习在5000左右，转正6000-7000左右。

问：如果公司聘用了你会不会来公司工作

答：会，（然后拍了拍公司的马屁）

问：你对公司是否有了解？

答：了解，简述了公司的情况

面试官：（给你说了一下薪资情况，实习工资）

（三）第二次人事面试

软件工管人员说是想深入了解一下你的情况

问：近三年的职业规划

刚入门职业，先学习一些技能，打下一些基础，然后做更深入的研究，往高处发展

问：猜测在公司能学到什么技术

答：计算机语言、通信方式

问：你觉得什么是高处发展，高处是什么？

答：

问：区分技术深度是靠什么区分？

答：

问：靠什么能力区分，具体是那些能力？

答：

问：你想研究的方向？

答：智能家居，智能硬件

问：是否能别的方向开发？

答：可以

问：如果是你不喜欢的开发呢？

问：研究很多方向还是研究一个方向？

问：指定方向研究，需要研究多久？

问：你要开发一个东西一定要研究透3、5年之后才会研究其他东西吗？

问：你会在学校自学一些课程吗，比如软件类的课程？

答：有C C++语言

问：有没有学过软件工程这门课程

问：到你可以独立做项目或者你可以带人做项目之间你需要积累什么？

答：技术、管理

问：简历上你在在XXX写文章？

答：是，主要是学习作用

问：点击量多吗

问：你做过XX个毕业设计？

答：XXX

问：都做过什么项目

答： ...

问：做这些毕业设计最麻烦的是什么？

答：硬件需要调试，可能会出现线路不正常

问：有没有说你功能不正确？ 客户评价？

答：基本比较满意...

问：为什么会有这么多好评？

答：价格，态度，项目的功能

问：嵌入式软件是你的目标，为什么要做这些，喜欢这些。

答：成就感...

问：大学之前就接触过专业？

问：你画过最复杂的板子是什么？

答：XXX

问：你在广州住房学习？

问：希望到广州？ 会定居吗，为什么选择广州？

问：为什么住在XXX，有亲戚朋友吗？

问：今年毕业吗？

问：有提前了解过公司做什么的吗？

答：有的

问：你应聘我们公司的嵌入式软件，你觉得你的优势是什么

问：用5个词来描述一下你工作的

答：负责、自学能力强

问：沟通能力？

答：语文能力、交际能力方面的例子

问：电子大赛需要做文档，做论文，这个有什么用？

答：对自己项目做总结

问：做这个重要吗？

答：重要

问：项目非常急你还写吗？

答：先留一段时间，空闲了再完善。

三、总结常问问题

1、说说进程和线程的区别

答：

简单的来说,一个程序至少有一个进程,一个进程至少有一个线程。

第一点、进程有自己的独立地址空间，每启动一个进程，系统就会为它分配地址空间，建立数据表来维护代码段、堆栈段和数据段，这种操作非常昂贵。而线程是共享进程中的数据，使用相同的地址空间，因此CPU切换一个线程的花费远比进程要小很多，同时创建一个线程的开销也比进程要小很多。

进程则不会影响另外一个进程，它有自己的独立地址空间。

第二点、线程之间的通信更方便，同一进程下的线程共享全局变量、静态变量等数据，而进程之间的通信需要以通信的方式（IPC)进行。

2、进程间通信都有些什么？

- 答：
- 1、无名管道通信
 - 2、有名管道通信
 - 3、消息队列通信
 - 4、信号量通信
 - 5、信号
 - 6、共享内存通信
 - 7、套接字通信

3、进程有几种状态

一般来说，进程有三个状态，即就绪状态，运行状态，阻塞状态

总结：fork一个进程之后，进程加入队列，处在就绪态，等待被调用，此时如果得到运行就进入运行态，这时如果遇到阻塞状态就进入等待态等待再条件满足后再次进入就绪态，如果运气不好，在运行态时直接被结束了，就进入了僵尸态，如果被信号给终止了，就进入停止态。

4、对内核各种锁的了解

内核锁

- 1、原子操作(atomic):
- 2、自旋锁(spinlock/spinlock_irqsave):
- 3、读写自旋锁(rwlock):
- 4、顺序自旋锁(seqlock):
- 5、RCU(读-拷贝-更新):
- 6、信号量(semaphore):
- 7、读写信号量;

5、TCP、UDP建立连接一般步骤

TCP编程的服务器端一般步骤是：

- 1、创建一个socket，用函数socket();
- 2、设置socket属性，用函数setsockopt();(可选)
- 3、绑定IP地址、端口等信息到socket上，用函数bind();
- 4、开启监听，用函数listen();
- 5、等待客户端连接上来，用函数accept();
- 6、收发数据，用函数send()和recv(), 者read()和write();
- 7、关闭网络连接;
- 8、关闭监听;

TCP编程的客户端一般步骤是：

- 1、创建一个socket，用函数socket();
- 2、设置socket属性，用函数setsockopt();可选
- 3、绑定IP地址、端口等信息到socket上，用函数bind();可选
- 4、设置要连接的对方的IP地址和端口等属性;
- 5、连接服务器，用函数connect();
- 6、收发数据，用函数send()和recv(), 或者read()和write();
- 7、关闭网络连接;

与之对应的UDP编程步骤要简单许多，分别如下：

UDP编程的服务器端一般步骤是：

- 1、创建一个socket，用函数socket();
- 2、设置socket属性，用函数setsockopt();可选
- 3、绑定IP地址、端口等信息到socket上，用函数bind();
- 4、循环接收数据，用函数recvfrom();
- 5、关闭网络连接;

UDP编程的客户端一般步骤是：

- 1、创建一个socket，用函数socket();
- 2、设置socket属性，用函数setsockopt();可选
- 3、绑定IP地址、端口等信息到socket上，用函数bind();可选
- 4、设置对方的IP地址和端口等属性;
- 5、发送数据，用函数sendto();
- 6、关闭网络连接;

6、网络分层

OSI七层网络模型	TCP/IP四层概念模型	对应网络协议
应用层(Application)	应用层	HTTP、TFTP, FTP, NFS, WAIS、 SMTP

OSI七层网络模型	TCP/IP四层概念模型	对应网络协议
表示层(Presentation)	应用层	Telnet, Rlogin, SNMP, Gopher
会话层(Session)	应用层	SMTP, DNS
传输层(Transport)	传输层	TCP, UDP
网络层(Network)	网络层	IP、ICMP、ARP、RARP、AKP、UUCP
数据链路层(Data Link)	数据链路层	FDDI, Ethernet, Arpanet, PDN, SLIP, PPP
物理层(Physical)	数据链路层	IEEE 802.1A IEEE 802.2到IEEE802.11

7、typedef和#define的用法与区别

一、typedef的用法

在C/C++语言中，typedef常用来定义一个标识符及关键字的别名，它是语言编译过程的一部分，但它并不实际分配内存空间，实例像：

```
typedef    int        INT;
typedef    int        ARRAY[10];
typedef    (int*)      pINT;
```

typedef可以增强程序的可读性，以及标识符的灵活性，但它也有“非直观性”等缺点。

二、#define的用法

#define为一宏定义语句，通常用它来定义常量(包括无参量与带参量)，以及用来实现那些“表面似和善、背后一长串”的宏，它本身并不在编

译过程中进行，而是在这之前(预处理过程)就已经完成了，但也因此难以发现潜在的错误及其它代码维护问题，它的实例像：

```
#define    INT            int
#define    TRUE           1
#define    Add(a,b)      ((a)+(b));
#define    Loop_10      for (int i=0; i<10; i++)
```

在Scott Meyer的Effective C++一书的条款1中有关于#define语句弊端的分析，以及好的替代方法，大家可参看。

三、typedef与#define的区别

从以上的概念便也能基本清楚，typedef只是为了增加可读性而为标识符另起的新名称(仅仅只是个别名)，而#define原本在C中是为了定义常量，到了C++，const、enum、inline的出现使它也渐渐成为了起别名的工具。有时很容易搞不清楚与typedef两者到底该用哪个好，如#define INT int这样的语句，用typedef一样可以完成，用哪个好呢？我主张用typedef，因为在早期的许多C编译器中这条语句是非法的，只是现今的

编译器又做了扩充。为了尽可能地兼容，一般都遵循#define定义“可读”的常量以及一些宏语句的任务，而typedef则常用来定义关键字、冗

长的类型的别名。

宏定义只是简单的字符串代换(原地扩展)，而typedef则不是原地扩展，它的新名字具有一定的封装性，以致于新命名的标识符具有更易定义变量的功能。请看上面第一大点代码的第三行：

```
typedef    (int*)        pINT;
以及下面这行：
```

```
#define    pINT2        int*
效果相同？实则不同！实践中见差别：pINT a,b;的效果同int *a; int *b;表示定义了两个整型指针变量。而pINT2 a,b;的效果同int *a, b; 表示定义了一个整型指针变量a和整型变量b。
```

