**2018笔试C语言及面试回忆版**

**By-飞飞**

**复试完后的第二天就被老师拉过去干活了。。。（往届的很容易被直接拉过去）成绩还没出来，心里总是不太安心，希望一切都是稳稳的。现在在去北京的路上，想着做一下笔试和面试的回忆，留给学弟学妹们，就像以前的学长学姐帮助我们一样，很感激他们，这种传统一定要延续下去。。。**

**第一部分面试题**

**面试的时候需要抽三个题：数据库、网络、操作系统**

**我当时抽中的是数据库：关系R有m个属性，关系S有n个属性，他们做笛卡尔积的话，总共有多少对匹配？答案：m\*n个**

**网络：NAT指的是什么，有什么作用**

**操作系统：生产者与消费者问题中，同步与互斥问题**

**我抽的这三个问题挺简单的，这些题库中都有。说到题库，由于我面试题准备的比较晚，一看题库辣么多。。。想死的心都有了，但后来我发现其实花个三到四天把它弄熟还是绰绰有余的。**

**简单说说英语面试：面试之前各种背自我介绍，是真的慌，很怕到时候尬聊。但后面进去面试的时候我就笑了，感觉这老师的口语发音还不如我。。。其实英语面试一般不会挂的**

**注：我感触最深的复试阶段预面试找导师环节，很多高分的同学可能还没找到导师，而一些分不高的却找到了，这里告诉你们预面试导师选人真的就是看你的能力，而不是单纯的分数！当然分数固然重要，但能力更重要!每个导师看重的方面也不尽相同，有的导师真的就看你数学能力，你如果数学考得突出，其他方面一般，导师就是想要你；再者你参加过很多竞赛，做过很多项目，能力很突出只要你分数达线了，导师会有理由不要你嘛！有些导师确实不喜欢跨考的，包括本校土著跨院的（我认识的一个本校土著就是这样的），其实大家都知道原因，跨考的专业知识方面或多或少欠缺些，一些导师喜欢那些上来就能上手的（比如某个导师直接说我们这就是一公司。。。）。预面试找导师过程真的好痛苦，这里有一点我觉得很重要，那就是一定要提前！过来！最好就是老师过年上班了（今年老师28号上班）就立马过来，来的越早预面试的机会越多，就像相亲一样，最后总能相中一个。好在我认识的人最后好像都有着落，但这个过程真的让人崩溃。。。**

**找到导师了，其实只要笔试不挂基本就是稳了，之前听学长说综合面试环节有导师的就是走过场了，当时心里还是有些慌（面试题都还没背完。。），面试完后发现真的差不多就是这样的。**

**成绩没有下来之前，我到现在都还是慌慌得，希望一切都稳稳当当的，不管怎么样我们真的尽力了。。**

**不说废话了，来点干货。。。**

**第二部分笔试题**

**注：感觉除了最后一个填空题在历年真题都有见过，记得不全，希望大家帮忙补充回忆。**

一、选择题部分（只记得部分了）

1、void main(){

int a[]={1,7,12,15};

int \*p1=a,\*p2 = p1++;

\*p1 += \*p2++;

printf(“%d %d”,\*p1,\*p2);

}

**答案：首先定义指针p1指向数组a首地址，然后定义指针p2，也指向数组首地址，然后p1自加，也就指向了数组第二个元素。\*p1 += \*p2++;语句先将p2指向的第一个元素的值加到p1指向的第二个元素的值上，也就是第二个元素值为8，然后p2自加，指向第二个元素（\*p2++，\*与++优先级相同，从右自左结合，先与++结合，表示语句执行完后p2指向下一个元素，然后与\*结合，表示p2现在所指向的第一个元素的值），所以最后输出结果为：8 8**

2、下面程序的输出结果是（ ）

long fib(int n)

{

if(n > 2) return (fib(n - 1) + fib(n - 2));

else return 2; //16年这里返回1，注意区别

}

void main(){

printf(“%d\n”,fib(3));

}

**答案：调用fib(3)，由于3>2，所以返回fib(2)+fib(1)，fib(2)和fib(1)都返回2，所以最后结果为：4**

3.struct Stu{

Int num;

Int score;

}Stu;

Void main(){

Stu stu[3]={“”100,100”,”200,200”,”300,300”};

Stu \*p;

P=stu;

}下列选项不正确的是()

A(p+1)->num=100 B p++ C \*p.num=100 D p=stu

**答案：选C，因为.的优先级高于\*，而.是直接取值，改为(\*p).num**

二．程序改错题

1下面程序的功能是将字符串src逆序输出，请将下面程序的错误改正，缺少的代码补全。

clude<stdio.h>

#include<stdlib.h>

#include<string.h>

void main()

{

char \*src = “hello,world”;

char \*dest,\*d,\*p;

int len,i;

len = strlen(src);

dest = (char \*)malloc(len);

p = src[len];

d = dest;

while(len-- != 0)

\*d ++ = \*p --;**//往年这里是d++=p--,但今年这里给的是正确的，\*（d ++） = \*（p --）也是可以的**

printf(“%s”,dest);

}

**答案：①dest = (char \*)malloc(len);改为dest = (char \*)malloc(len + 1);因为要多出一个存储字符串结束符’\0’（或者dest = (char \*)malloc(sizeof(char)\*(len + 1));这里sizeof(char)的值为1，所以写成len+1也行）**

**②p = src[len];改为p = &src[len - 1];因为src[len - 1]才是最后一个字符，数组下标从0开始，并且要取地址&赋给指针p**

**③在printf(“%s”,dest);语句前要增加一行语句\*d = ‘\0’;设置字符串结束标识**

三、读程序的功能

#include<stdio.h>

void main()

{

int i,j,k,d,flag1,flag2;

scanf("%d",&d);

printf("d=%d\n", d);

for(i=1;i<=100;i++)

{

j=i;

flag1 = 0;

while((j>0) && (!flag1))

{

k=j%10;

j=j/10;

if(k==d)

flag1=1;

}

if(flag1)

{

j=i\*i;

flag2 = 0;

while((j>0) && (!flag2))

{

k=j%10;

j=j/10 ;

if(k==d)

flag2=1;

}

if(flag2)

printf("%-5d %-5d\n", i, i\*i);

}

}

}

**答案：功能：从1-100中找出符合以下条件的数，该数和该数的平方中都含有输入的数字d**

四、简答题

1、请简述C语言的隐式类型转换发生的四种情况，并说明每种情况如何转换。（注：这里面还有个小题，float如何四舍五入转化成int）

**1、算术运算式中，低类型能够转换为高类型。**

**2、赋值表达式中，右边表达式的值自动隐式转换为左边变量的类型，并赋值给它。**

**3、函数调用中参数传递时，系统隐式地将实参转换为形参的类型后，赋给形参。**

**4、函数有返回值时，系统将隐式地将返回表达式类型转换为返回值类型，赋给调用函数。**

2、从C语言执行效率方便，简述下C语言除了指针、宏函数、位运算还采取了哪些措施提高执行效率。

**1、循环嵌套中将较长循环设为内存循环，较短循环设为外置循环，以减少cpu跨切循环层的次数，提高程序的运行效率。**

**2、让用户可以直接操作进程虚拟地址，给用户提供了高效的内存拷贝函数，变量类型是弱类型，可以各种强制转换。**

**3、API，可以调用系统API，接近底层**

1. **嵌入汇编语言，更贴近极限**

3、问数组越界会产生什么后果？

**数组越界，将会把数据存放到一个未知的区域，而这个未知的区域如果恰好是系统很重要的位置，可能了就会修改系统程序导致系统出错甚至是崩溃，如果这个位置不是和系统有关，可能结果并不明显或者没有产生影响，但为了系统安全需要进行数组下标越界检查。**

4、实参和形参都是数组元素；实参是数组地址，形参是指针；实参和形参都是数组地址问这三个都是什么传递方式，

**答案：值传递、地址传递、地址传递**

五、填空题

快排找第K小 ps:红色字体表示需要填写的空

注：今年这道题真的一下子就看出来科大老师出题的仓促。。。题目直接省略了一段关键代码，一开始一脸懵逼，下面原题差不多是这样的，学弟学妹不要惊讶，这道题出来估计被很多人吐槽。。

Low=0;high=n-1;

do{

do{

I=low;j=high;t=a[i];

}while(i<j);

a[i]=t;

If(i==k)

Return t;

If(i>k)

high=i-1;

if(i<k)

low=i+1;

}while(low<high);

Return a[low];//这个空我可能错了

其实这道题正确的打开方式是这样的：从往年题扒过来的

Low=0;high=n-1;

do{

I=low;j=high;t=a[i];

do{

While(a[j--]>t);

While(a[i++]<t);

If(i<j) swap(a[j],a[i]);//具体代码实在忘了，反正不是空空，我这样写运行是一样的

}while(i<j);

If(i==k)

Return t;

If(i>k)

high=i-1;

if(i<k)

low=i+1;

}while(low<high);

Return a[low];//这个空我可能错了

2、根据下面的代码，填写表格。整数算2字节，字符2字节，指针4字节。每个区域的起始地址都是0，内存按2字节编址。（注：哇这道题历年出现的概率也好大唉，今年表格给出了但没有给出内存的相对地址，并且fun函数里较以前多了一个static类型的变量,还有一个坑就是以前字符是1个字节，今年是字符是两个字节，如果没看到的话就。。）

int num=2;

void main()

{

char str1[2018]={"UESTC"};//不记得这里几个字符了，反正就是要乘2

char \*str2="CHENGDU";

char p;

}

void func(int m)

{

Static int sta;

int n=10;

}

|  |  |  |
| --- | --- | --- |
| **内存区域** | **常量或变量名** | **占用内存大小** |
| **常量区** | **2** | **2** |
| **2018** | **2** |
| **"UESTC"** | **12** |
| **"CHENGDU"** | **16** |
| **10** | **2** |
| **全局** | **num** | **2** |
| **静态数据区** | **Sta** | **2** |
| **main函数** | **str1** | **4036** |
| **str2** | **4** |
| **p** | **2** |
| **func函数** | **m** | **2** |
| **n** | **2** |

3、用Eratasthene筛选法求200以内的素数，修改代码使得效率变高

注：这道题考前幸好我看见群里大佬在群里发了图片，然后看了一眼，大致有个印象。

题目程序大致这样：

#include<stdio.h>

#include<math.h>

Void main()

{

int prime[201]={0};

int d;

int k;

int i;

for(d=2;d<=200;d++)//这里修改为for(d=2;d<=sqrt(200);d++)

{

if(prime==0){

for(k=1;k<=200;k++){// 这里修改为f(k=2\*d;k<=200;k+d)

If(k%d==0){//这里修改为prime[k]=1,把if条件判断去掉 Prime[k]=1;

}

}

}

}

k=0;

for(i=2;i<=200;i++)

{

if(prime[i]==0)

{

Printf(“%d\t”,i)

}

}

}

}

}

4、最后一道题是一道加密解密问题，大致的意思就是通过主函数输入两个字符串，第一个字符串作为加密解密的目标字符串，第二个字符串通过函数转换为整型数据作为解密加密的key

解密加密规则是：

解密：将字符串按a-z字母表的顺序右移K位，注意这里是循环的，我当时做的时候忘记了…循环的意思就是移动到z之后一定要返回到a

解密：将字符串按a-z字母表的顺序左移K位，注意这里是循环的，我当时做的时候忘记了…循环的意思就是移动到z之后一定要返回到a

这样分析的话其实并不难，但是我错了，，，

代码忘了，但相信意思明白了就会了，这是我觉得我在历年真题中唯一没看到的题

六、程序设计题20分

输入若干个整数（以0结束）如何逆序构建双向链表，这道题很简单就不写了，王道数据结构都有