



北京中软笔试题

- 1.不许用中间变量,把 String ABCDE 倒转
- 2.10000 个数求第 2 大的数,不许用排序算法.
- 3.排序算法的测试用例

我的答案:

1.

```
1. include "stdafx.h"
```

```
1. include <iostream>
```

```
1. include <string>
```

```
using namespace std;
```

```
int _tmain(int argc, _TCHAR* argv[])
```

```
{
```

```

char *ps = new char[15];
strcpy_s(ps,15,"I am yuchifang");
cout<<"before reverse:"<<endl;
cout<<ps<<endl;
int i = 0;
int j = 13;
while(i<j)
{
    ps[i] = ps[i]+ps[j];
    ps[j] = ps[i]-ps[j];
    ps[i] = ps[i]-ps[j];
    i++;
    j--;
}
cout<<"after reverse"<<endl;
cout<<ps<<endl;
return 0;
}

```

.....

这是我在[中软](#)国际面试的一些题目，看你会吗？

- 1、**Java** 中的属性和字段有什么区别？ --这个是当时考我的一道题，我不会啊！！！！！！！！
- 2、**C/C++**你更熟悉那个？ **C++**中的虚函数用在什么地方？ **JAVA** 中的虚函数呢？（我没记得 **Java** 中还有这个）
- 3、你对数据库中存储过程了解多少？说说。
- 4、假如微软给你 500 万美元项目经费，你怎么利用它？

5、你对考研是什么态度？

6、你对加班是什么态度？ 这个答案项目经理说出来后我感觉太现实了，你猜他怎么说的？和 IT 人有关的

7、Why do you ICS&S? （我忘具体怎么说的了，大意就是问你为何要来中软国际？人家可是用全英语问的，说的挺快的，幸好我听懂一点）

问题太多了，大多是技术问题，不过都和知识要点有关，不考太难的太复杂的，不过我还是大部分不会啊!!! 英语会让你就自己喜欢的方向作简介呢。难，要我说就完了。

先想出这多了，下次再写，技术问题好多，有空把笔试题给大家看看，唉！

.....

题目为写算法生成随机 4 位数,要求每位数字不同.

Solution:

有个隐含条件,即第一个数字不能为 0. 以下是我初步的 javascript 解法:

```
function rand4(){
    var arr = new Array();
    var index = 0;
    var tempNum;
    while(index<4){
        tempNum = Math.round(Math.random()*9);
        if(!(index==0 && tempNum==0)){
            //hold point
            if(arr[tempNum]== null){
```

```

        arr[tempNum] = tempNum;
        ++index;
    }
}
}
var ret = 0;
for(var i in arr){
    if (ret != 0){
        ret *= 10;
    }
    ret += parseInt(i);
}    alert(ret);
return ret;
}

```

我相信会编程的人都会写出这个算法. 基本上是很普通的.这个算法有个不愉快的地方就是 **hold point** 那块,成功率为 100%(第一次),90%(第二次),80%(第三次),70%(第四次),越来越低,不过只要不低于 50%,程序效率还不会打折扣.

以下是我的第二种写法,用空间换了点时间,用 C++语言写的:

```
1. include <iostream>
```

```
1. include <stdlib.h>
```

```
using namespace std;
```

```
class Rand4{
```

```
    private:
```

```
        int a[10];
```

```

int getNum(int index){
    while(a[index]==-1){
        ++index;
        index %= 10;
    }
    int ret = a[index];
    a[index] = -1;
    return ret;
};

void init(){
    for(int i=0;i<10;++i){
        a[i] = i;
    }
};

public:
    int Next(){
        init();
        int ret = (rand()%10) *1000;
        int index = 100;
        int time = 3;
        while(time>0){
            ret += getNum((rand()%10)*index;
            index /= 10;
            --time;
        }
        return ret;
    };
};

int main(int argc, char *argv[])

```

```
{  
    Rand4 r4;  
    cout<<r4.Next()<<endl;  
    cout<<r4.Next()<<endl;  
    cout<<r4.Next()<<endl;  
    system("PAUSE");  
    return 0;  
}
```

此法的时间复杂度为正宗的 $O(n)$, 空间开销(不包含调用库函数的)就比前一个例子多了些.

此法在 n 取 m 的随机计算中, m 从 0 一直到趋近于 n , 效率都是不变的.

其他的高效方法暂时还没有想出.