

3. 编写程序..

#include <stdio.h>

int main() {

int i, n, j;

int sum = 1, total = 0; //在此忽略越界情况, 题目没有给出大小

printf("请输入一个数 n: ", &n);

scanf("%d", &n);

for (i = 1; i <= n; i++) {

sum = 1;

for (j = 1; j <= i; j++) {

sum *= j;

//得到单个阶乘和

}

total += sum;

//阶乘和相加

}

printf("%d", total);

return 0;

}

姓名: 刘佳

装订线

2. 编写一个程序...

```
#include <stdio.h>
#include <math.h>
int main(){
    int x, i, j;
    printf("请输入取值范围");
    scanf("%d", &x);
    for (i = x; i > 1; i--) { // 从大到小, 保证效率
        for (j = 2; j <= sqrt(x); j++) {
            if (i % j == 0) {
                break;
            }
            if (j == sqrt(x)) {
                printf("范围内的最大素数为%d", i);
            }
        }
    }
    return 0;
}
```

硕士研究生招生复试

自命题科目答题纸

考试科目代码及名称: F0803 程序设计综合 成绩: _____

注意事项: 1、考生编号、姓名、报考专业须写在装订线内指定位置。

2、所有答案必须写在答题纸上, 做在试题纸或草稿纸上一律无效。

3、字迹要清晰, 保持卷面清洁, 一律使用蓝色或黑色钢笔或签字笔。

4、考生答完试题后, 在“共 页”处填写答卷的总页数。

5、禁止做任何与考试无关的标记。

1. 输入2个正整数

#include <stdio.h>

int main()

{
int m, n, i, j;

scanf("%d %d", &m, &n);

~~for (i = m; i <= n; i++) {~~

if (m < 1 || m > 1000 || n < 1 || n > 1000) { //判断输入是否满足要求
return 0;

}

for (i = m; i <= n; i++) {

for (j = 1; j <= 10; j++) {

if (i == j * j * j) {

printf("%d", i);

}

}

}

return 0;

}

考生编号: 105360431505714

姓名: 姚俊杰

报考专业: 电子信息

装订线

姓名: 张松

装订线

6. 假设一整形数组...
#include <stdio.h>

```
int main() {  
    int i, Z, F = 0
```

```
    int str[N] = {...}; // 假设该数组为 str, 里面有 N 个数.
```

// 第一步, 求得里面正数和负数各有多少 Z: 正数个数, F: 负数个数

```
    for (i = 0; i < N; i++) {  
        if (str[i] >= 0) {  
            Z++;  
        } else {  
            F++;  
        }  
    }
```

用伪代码描述的
可能不太规范,
故没有严格按照 C 标
准语法. **/

// 第二步, 创建双指针, 进行比较 遍历交换.

```
    int *m, *n, j, i;  
    m = &str[0];  
    n = &str[N-1];  
    for (i = 0; i < F; i++) {  
        *m++;  
        *n++;  
    }
```

```
    *m++;  
    if (F == 0 || Z == N)  
        for (i = 0; i < N; i++) {  
            if (*m > *n) {  
                temp = *m;  
                *m = *n;  
                *n = temp;  
            }  
            *m--;  
            *n++;  
            F++;  
            Z++;  
        }
```

// 当一指针到达终点就停止.

// 如果左边大, 就交换

第 7 页 (共 7 页)

最后 printf 输出.

姓名: 刘俊豪

装订线

5. 编写一个程序, 输入一个长度不超过100的字符串。

```
#include <stdio.h>
#define Max 100
#include <string.h>
int main()
```

char str[Max], r, str2[Max] //用str2[]来存要输出的字符。

int i, len, j, num;

~~for (i=0;~~

~~scanf("%s",~~

gets(str);

len = strlen(str);

for (i=0; i < len; i++) {

r = str[i];

~~str2[i] = str[i];~~

~~if (r == str[i])~~

for (j = i+1; j < len; j++) {

if (r == str[j]) {

str[j] = '0'; //将所有相同的字符置为'0'

num++;

}

for (i=0; i < len-num; i++) {

if (str[i] == str2[i]) {

continue;

} else {

r = str[i]; //找到下一个不相同的字符。

}

}

for (i=0; str2[i] != '0'; i++) {

printf("%c", str2[i]);

}
return 0;

}

思路: 将所有相同的置为'0';
找到一个不相同的则放入新的数组。

姓名: 张俊杰

装订线

(接上一页) 第4题.

```
if (flag < str[i]) {  
    goto: sky;  
    count = i + 1;  
    goto: sky; }  
}
```

//这里迫不得已使用goto语句,重新找合适的大于0的数.

```
printf("%d", flag);  
return 0;
```

```
for (i = 0; N[i] != 0; i++) { //找到指定数组中最大的那个
```

```
    int temp;  
    temp = N[i];  
    if (N[i] > temp) {  
        temp = N[i];  
    }  
}
```

```
printf("%d", temp);  
return 0;
```


4. 输入一个长度为10的数组 --

```
#include <stdio.h>
```

```
#define Max 10
```

```
int main() {
```

```
    int str[Max], i, flag=0, j, M, count;
```

```
    for (i=0; i<Max; i++) {
```

```
        scanf("%d", str[i]);
```

```
    }
```

```
    while for goto sky;
```

```
    for (i=0; i<Max; i++) {
```

```
        if (str[i] > 0) {
```

```
            flag = str[i];
```

```
            count = i;
```

```
        }
```

```
        if (flag == 0) {
```

```
            flag M = str[0];
```

```
            for (j=1; j<Max; j++) {
```

```
                if (str[j] > M) {
```

```
                    M = str[j];
```

```
                }
```

```
            }
```

```
            printf("最大值为: %d", M);
```

```
            return 0;
```

```
    }
```

```
    for (i=count; i<Max; i++) {
```

```
        if (str[i] > 0) {
```

```
            flag += str[i];
```

```
        } else {
```

思路: 如果全小于0, 找到单个最大值, 否则, 从头开始, 将连续的正数相加, 如果遇到负数, 就比较, 如果负数大, 则从负数后面开始继续按相同的办法, 同时记录刚刚这一段的大小, 最后放入一个数组, 最后输出这个数组最大值

// goto 到这里.

(这里没有使用暴力法.)