安徽大学 2017 年计算机学院考研复试上机真题

1、由键盘任意输入 10 个正整数（有奇数也有偶数），要求输出其中的每个奇数，并输出奇数的个数，所有奇数之和。（10 分）

例如:若输入数据为:11，4，3，2，7，6，8，5，10，9，

则输出为: 11，3，7，5，9

NUM＝5

SUM＝35

源程序:

```
#include<stdio.h>
int judge(int number){
    if(1 == number%2){
        return 1;
    }
    return 0;
}
void main(){
    int num, num_odd = 0, sum_odd = 0;
    printf("请输入 10 个正整数:\n");
    for(int i = 0; i < 10; i++){
        scanf("%d,", &num);
        if(1 == judge(num)){
            printf("%d ", num);
            num_odd++;
            sum_odd += num;
        }
    }
    printf("\nNUM = %d\nSUM = %d\n", num_odd, sum_odd);
}
```

2、编程序找出 1000 以内的所有“完数”，并输出该完数和它的所有因子，（一个数恰好等于他的因子之和，这个数成为“完数”， 例如：6 的因子是 1，2，3，并且 6 = 1 + 2 + 3，所以 6 是一个“完数”）。（10 分）

源程序:


```
#include<stdio.h>
int judge(int num){
    int sum = 0, temp[30], j = 0;
    for(int i = 1; i <= num/2; i++){
        if(num % i == 0){
            sum = sum + i;
            temp[j++] = i;//temp[]数组存储 num 的所有约数
        }
    }
    if(sum == num){
        printf("完数 %d 的所有完数为:", num);
```

```
            for(int i = 0; i < j; i++){
                printf("%d ", temp[i]);
            }
            printf("\n");
            return 1;
        }else{
            return 0;
        }
}
void main(void){
        for(int number = 1; number <= 1000; number++){
            judge(number);
        }
        printf("\n");
}
```

3、由键盘任意输入一行字符（仅由英文字母及空格组成）编程实现：（1）输入每个单词及长度；（2）输出最长的单词（假设：相邻的两个单词之间用一个和多个空格隔开）（10 分）

例如：

若输入为：the benefits it can offer

则输出为：

the	3
benefits	8
it	2
can	3
offer	5

the longest word is：benefit

若输入为： this is a book

则输出为：

this	4
is	2
a	1
book	4

the longest word is：this book

源程序：

```
#include<stdio.h>
#include<string.h>
int po[20], poj = 0;
int le[20], lek = 0;
int cache[10] = {}, cac_len = 0;
int alphabetic(char c){
        if((c >= 'a' && c <= 'z') || (c >= 'A' && c<='Z'))
            return 1;
        else
            return 0;
        }
int longest(char string[]){
        int len = 0, i, length = 0, flag = 1, place , point;
        for(i = 0; i <= strlen(string); i++){
            if(alphabetic(string[i])){
                if(flag){
                    point = i;
```

```
                po[poj++] = point;
                flag = 0;
            }
            len++;
        }else{
            flag = 1;
            le[lek++] = len;
            if(len >= length){
                if(len > length){
                    for(int c = 0; c < 10; c++){
                        cache[c] = 0;
                    }
                    cac_len = 0;
                }
                length = len;
                place = point;
                cache[cac_len++] = place;
            }
            len = 0;
        }
    }
    return place;
}
void main(void){
    int i;
    char line[100];
    printf("输入一行文本\n");
    gets(line);
    longest(line);
    for(i = 0; i < poj; i++){
        for(; alphabetic(line[po[i]]); po[i] = po[i] + 1){
            printf("%c", line[po[i]]);
        }
        printf("%6d\n", le[i]);
    }
    printf("\nthe longest word is: ");
    for(i = 0; i < cac_len; i++){
        for(; alphabetic(line[cache[i]]); cache[i] = cache[i] + 1){
            printf("%c", line[cache[i]]);
        }
        printf(" ");
    }
    printf("\n");
}
```