

作业 11

必做题

第一题

源代码:

```
#include <stdio.h>
#include <stdlib.h>
/*数组输入*/
char cfInput(int *piList, int iLength) {
    int iRoundInput=0; /*循环变量*/
    printf("请输入数组: ");
    for (; iRoundInput<iLength; iRoundInput++) /*逐位输入*/
        scanf("%d", piList+iRoundInput);
    return 0;
}
/*数组输出*/
char cfOutput(int *piList, int iLength) {
    int iRoundInput=0; /*循环变量*/
    printf("整合排序数组为: \n");
    for (; iRoundInput<iLength/2; iRoundInput++) /*逐位输入*/
        printf("%d\t", *(piList+iRoundInput));
    printf("\n");
    for (iRoundInput=iLength/2; iRoundInput<iLength; iRoundInput++) /*逐位输入*/
        printf("%d\t", *(piList+iRoundInput));
    printf("\n");
    return 0;
}
/*单个数组选择排序*/
char cfRearrange(int *piList, int iLength) {
    int iRoundRA1=0, iRoundRA2; /*循环变量*/
    for (; iRoundRA1<iLength-1; iRoundRA1++) {
        for (iRoundRA2=iRoundRA1+1; iRoundRA2<iLength; iRoundRA2++) {
            if (*(piList+iRoundRA1)>*(piList+iRoundRA2)) /*大小反了交换*/
                int iTemp=*(piList+iRoundRA1);
                *(piList+iRoundRA1)=*(piList+iRoundRA2);
                *(piList+iRoundRA2)=iTemp;
        }
    }
    return 0;
}
/*两个数组按序组合*/
char cfCj(int *piList1, int *piList2, int *piListCj, int iLengthS1, int iLengthS2) {
    int iLocCj=0, iRound1=0, iRound2; /*位置和循环变量*/
    for (; iRound1<iLengthS1; iRound1++) /*不比第一个数组元素大的第二数组元素*/
```

```

        if ((iRound1==0) || (*(piList1+iRound1)!=*(piList1+iRound1-1))) {
            for (iRound2=0; iRound2<iLengthS2; iRound2++) { /*加入第二数组元素*/
                if ((*(piList2+iRound2)<*(piList1+iRound1)) && ((iRound1==0) ||
                    (*(piList2+iRound2)>*(piList1+iRound1-1))))
                    *(piListCj+iLocCj++)=*(piList2+iRound2);
            }
        }
        *(piListCj+iLocCj++)=*(piList1+iRound1); /*加入第一数组元素*/
    }
    for (iRound2=0; iRound2<iLengthS2; iRound2++) { /*比第一个数组元素大的第二数组
元素*/
        if (*(piList2+iRound2)>*(piList1+iLengthS1-1)) /*加入第二数组元素*/
            *(piListCj+iLocCj++)=*(piList2+iRound2);
    }
    return 0;
}
/*主函数*/
char main() {
    int iList1[10], iList2[10], iListCj[20]; /*定义数组*/
    cfInput(iList1, 10); /*输入第一数组*/
    cfInput(iList2, 10); /*输入第二数组*/
    cfRearrange(iList1, 10); /*第一数组排序*/
    cfRearrange(iList2, 10); /*第二数组排序*/
    cfCj(iList1, iList2, iListCj, 10, 10); /*数组合并*/
    cfOutput(iListCj, 20); /*数组输出*/
    system("pause");
    return 0;
}

```

运行结果：

```

请输入数组： 1   4   9   16   25   3   7   11   14   27
请输入数组： 0   2   5   8   15   28   18   17   19   29
整合排序数组为：
0   1   2   3   4   5   7   8   9   11
14   15   16   17   18   19   25   27   28   29
请按任意键继续： . . |

```

第二题

源代码：

```

#include <stdio.h>
#include <stdlib.h>
int del(int*p, int m) {
    char cFlag=1; /*轮到的报数*/
    while (m>1) {
        char cTemp=*p, cRound=0; /*报数者的序号和循环变量*/
        for (; cRound<m-1; cRound++) /*后续者向前移位*/
            *(p+cRound)=*(p+cRound+1);
    }
}

```

```

        if (cFlag==3) { /*报号到3消去该人*/
            cFlag=1;
            m--;
        }
        else { /*报号非3继续循环*/
            cFlag++;
            *(p+m-1)=cTemp;
        }
    }
    return *p;
}

int main() {
    short sLength, sRound=0; /*人数变量和循环变量*/
    int num[50]; /*保存序号的数组*/
    printf("请输入人数: ");
    scanf("%hd", &sLength); /*输入人数变量*/
    for (; sRound<sLength; sRound++)
        *(num+sRound)=(int) sRound;
    printf("剩余者的编号为 (初始编号为0): %d", del(num, (int) sLength));
    system("pause");
    return 0;
}

```

运行结果:

```

请输入人数: 40
剩余者的编号为 (初始编号为0): 27请按任意键继续. . .

```

第三题

源代码:

```

#include <stdio.h>
#include <stdlib.h>
int a[10];
int *pa[10];
int ifInitPa(void) { /*初始化指针数组*/
    char cRound=0;
    for (; cRound<10; cRound++)
        pa[cRound]=a+cRound; /*每一元素指向数组对应位次元素*/
    return 0;
}

int ifInputA(void) { /*初始化整数数组*/
    char cRound=0;
    for (; cRound<10; cRound++) {
        printf("请输入第%d个数: ", cRound+1);
        scanf("%d", a+cRound); /*依次录入每个整数*/
    }
}

```

```

    }
    return 0;
}
int ifJudgeAndSwap(char cNorm, char cLoc) { /*具体位比较大小并调换顺序*/
    if (**(pa+cLoc)<**(pa+cNorm)) {
        char *cpTemp=(char*)(pa+cLoc);
        *(pa+cLoc)=*(pa+cNorm);
        *(pa+cNorm)=(int*)cpTemp;
        return 1;
    }
    return 0;
}
int ifArrange(char cLoc) { /*插入排序, cLoc表示正在插入的元素下标*/
    if (cLoc==0) return 0; /*起始条件, 即第一位*/
    else{
        char cNorm=0; /*被比较的位次下标*/
        ifArrange(cLoc-1); /*将前面的部分进行排序*/
        for (;cNorm<cLoc;cNorm++) {
            ifJudgeAndSwap(cNorm, cLoc); /*进行交换*/
        }
    }
    return 1;
}
int ifPrintNum() { /*按序打印a中整数*/
    char cRoundPN=0;
    for (;cRoundPN<10;cRoundPN++)
        printf("%d\t", *(pa+cRoundPN));
    return 0;
}
int ifPrintLoc() { /*按序打印对应序号*/
    char cRoundPN=0;
    for (;cRoundPN<10;cRoundPN++)
        printf("%d\t", *(pa+cRoundPN)-a);
    return 0;
}
int main() {
    ifInitPa();
    ifInputA();
    ifArrange(9);
    printf("排序结果为: ");
    ifPrintNum();
    printf("\n对应序号为: ");
    ifPrintLoc();
    system("pause");
}

```

```

        return -52;
    }

```

运行结果：

```

请输入第1个数：15
请输入第2个数：-33
请输入第3个数：22
请输入第4个数：123
请输入第5个数：-52
请输入第6个数：52
请输入第7个数：767
请输入第8个数：3314
请输入第9个数：8238
请输入第10个数：0
排序结果为：-52 -33    0    15    22    52    123    767    3314    8238
对应序号为：4  1    9    0    2    5    3    6    7    8    请按任意键继续. . .

```

选做题

第一题

HhnewyearNEWYEAR

YEAR

第二题

源代码：

```

#include <stdio.h>
#include <stdlib.h>
/*cFlag排序含义宏定义*/
#define MAJOR 1
#define MINOR 0
/*三个表示四位数的数组定义*/
char cOrigin[4]/*原始数*/, cMajor[4]/*降序数*/, cMinor[4]/*升序数*/;
/*四位数转化为数组*/
short sIToL(short sI, char cL[4]) {
    char cRound=3; /*循环变量*/
    for (; cRound>=0; cRound--) { /*从小到大逐位添加*/
        *(cL+cRound)=sI%10;
        sI/=10;
    }
    return 0;
}
/*数组转化为四位数*/
short sLToI(char cL[4]) {
    short sResult=*(cL); /*结果*/
    char cRound=1; /*循环变量*/
    for (; cRound<4; cRound++) {
        sResult*=10;
        sResult+=*(cL+cRound);
    }
    return sResult;
}
/*整理数组*/

```

```

char cArrange(char cOrigin[4],char cDest[4],char cFlag) {
    char cRound1=0,cRound2=0; /*循环变量*/
    for (;cRound2<4;cRound2++)
        *(cDest+cRound2)=*(cOrigin+cRound2);
    for (;cRound1<4;cRound1++) { /*选择排序*/
        for (cRound2=cRound1+1;cRound2<4;cRound2++) {
            if ((*(cDest+cRound1)<*(cDest+cRound2) && cFlag==MAJOR) || /*降序*/
                (*(cDest+cRound1)>*(cDest+cRound2) && cFlag==MINOR)) { /*升序*/
                /*交换位置*/
                char cTemp=*(cDest+cRound1);
                *(cDest+cRound1)=*(cDest+cRound2);
                *(cDest+cRound2)=cTemp;
            }
        }
    }
    return 0;
}

/*一次完整操作*/
short sOnce(short sOrigin) {
    int i;
    short sMajor,sMinor,sDifference; /*倒序数、正序数和差值变量定义*/
    sIToL(sOrigin,cOrigin); /*转化为数组*/
    cArrange(cOrigin,cMajor,MAJOR); /*倒序排序*/
    cArrange(cOrigin,cMinor,MINOR); /*正序排序*/
    sMajor=sLToI(cMajor);
    sMinor=sLToI(cMinor);
    sDifference=sMajor-sMinor;
    printf("%d-%d=%d\n",sMajor,sMinor,sDifference);
    return sDifference;
}

/*判断输入是否有效*/
short sJudge(int iInput) {
    /*符合要求*/
    if (iInput<10000 && iInput>=1000 && iInput%1111!=0)
        return (short)iInput;
    /*不符合要求*/
    else{
        printf("error\n");
        return -13108;
    }
}

/*主函数*/
int main() {
    int N;

```

```
short sProcess;  
printf("请输入一个非1111倍数的四位数: ");  
scanf("%d",&N);  
if((sProcess=sJudge(N))!=-13108) {  
    while ((sProcess=sOnce(sProcess))!=6174) {}  
}  
system("pause");  
return 0;  
}
```

运行结果:

```
请输入一个非1111倍数的四位数: 4211  
4211-1124=3087  
8730-378=8352  
8532-2358=6174  
请按任意键继续. . . |
```

```
请输入一个非1111倍数的四位数: 12511  
error  
请按任意键继续. . . |
```