

# 数据结构实验二

姓名学号成绩

## 1 实验目的

## 2 实验内容

## 3 数据结构及算法描述

## 4 详细设计

## 5 程序代码

```
\tiny
#include<stdio.h>
#include<stdlib.h>

typedef struct Num_score//定义了一个结构体用于储存每个学生的学号、各科成绩、总分、加
{
    int Num;
    int score[3];
    int sum;
    float aver;
} student;

void Input_score(int *p,student *stu){//读入已有学生信息
    int i;
    for(i=0;i<*p;i++){
        printf("请依次输入第%d个学生的学号，语文，数学，英语成绩（用空格分开）\n");
        scanf("%d%d%d",&(stu+i)->Num,&(stu+i)->score[0],&(stu+i)->score[1]);
        while ((stu+i)->score[0]>100||((stu+i)->score[0]<0||((stu+i)->score[1]>100||((stu+i)->score[1]<0||((stu+i)->score[2]>100||((stu+i)->score[2]<0)))
        {
            printf("请重新输入第%d个学生的信息：",i+1);
            scanf("%d%d%d",&(stu+i)->score[0],&(stu+i)->score[1],&(stu+i)->score[2]);
        }
        stu[i].sum=stu[i].score[0]+stu[i].score[1]+stu[i].score[2];
        stu[i].aver=stu[i].score[0]*0.3+stu[i].score[1]*0.5+stu[i].score[2]*0.2;
    }
}
```

```
}
```

```
void Output_Grade(int *p, student *stu){//输出已存入的学生信息
    int i;
    printf("各学生成绩如下：\n");
    for ( i = 0; i < *p; i++)
    {
        printf("第%d学生的信息为：\t学号%d\t语文%d\t数学%d\t英语%d\t总分%d\n", i, stu[i].Num, stu[i].score[0], stu[i].score[1], stu[i].score[2], stu[i].sum);
    }
}
```

```
student* Add_score(int *p, student *stu){//添加学生信息
    // student *p;
    stu=(student*)realloc(stu,(*p+1)*sizeof(student));
    if(!stu) exit(0);
    printf("请依次输入新加入学生成绩：\n");
    scanf("%d%d%d", &stu[*p].Num, &stu[*p].score[0], &stu[*p].score[1], &stu[*p].score[2]);
    while (stu[*p].score[0] < 0 || stu[*p].score[0] > 100 || stu[*p].score[1] < 0 || stu[*p].score[1] > 100 || stu[*p].score[2] < 0 || stu[*p].score[2] > 100)
    {
        printf("请重新依次输入新加入学生成绩：\n");
        scanf("%d%d%d", &stu[*p].score[0], &stu[*p].score[1], &stu[*p].score[2]);
    }
    stu[*p].sum=stu[*p].score[0]+stu[*p].score[1]+stu[*p].score[2];
    stu[*p].aver=stu[*p].score[0]*0.3+stu[*p].score[1]*0.5+stu[*p].score[2]*0.2;
    *p++;
    Output_Grade(p, stu);
    // return stu;
}
```

```
void Delete_score_Num(int *p, student *stu){
    int i, j, Num;
    printf("请输入要删除学生信息的学号：");
    scanf("%d", &Num);
    for (i=0; i<*p; i++){
        if (stu[i].Num==Num){
            for (j=i; j<*p; j++) stu[j]=stu[j+1];
            *p-=1; i--;
        }
    }
    Output_Grade(p, stu);
}
```

```
void Sort_Score_Num(int *p, student *stu){//根据学号升序排列
    int i, j, k;
```

```

        student temp;
        for ( i=0;i<*p-1;i++){
            k=i;
            for ( j=i+1;j<*p;j++){
                if ( stu [ k ].Num>stu [ j ].Num) k=j;
            }
            temp=stu [ i ]; stu [ i ]=stu [ k ]; stu [ k ]=temp;
        }
        printf(" 根据学号排序如下 : \n");
        Output_Grade(p, stu );
    }

void Sort_Score_Sum(int *p, student *stu){//根据总分升序排列
    int i, j, k;
    student temp;
    for ( i=0;i<*p-1;i++){
        k=i;
        for ( j=i ; j<*p; j++){
            if (stu [ k ].sum>stu [ j ].sum) k=j;
        }
        temp=stu [ i ]; stu [ i ]=stu [ k ]; stu [ k ]=temp;
    }
    printf(" 根据总成绩排序如下 : \n");
    Output_Grade(p, stu );
}

int main(){
    int n, Num, c, *p;
    p=&n;
    student *stu;
    printf(" 请输入数组大小n (1-100) : ");
    scanf("%d", p);
    while (n<0||n>100)
    {
        printf(" 请重新输入数组大小 (1-100) : ");
        scanf("%d", p);
    }
    stu=(student*) malloc(n*sizeof(student));
    if (!stu) exit(0);
    Input_score(p, stu);
    // Sort_Score_Num(n, stu);
    Output_Grade(p, stu);
    printf(" 执行增加学生信息请输入1, 按学号排序请输入2, 按总成绩排序输入3, 删除学生输入4, 退出请输入5\n");
    scanf("%d",&c);
}

```

```
while (c)
{
    if(c==1)
        Add_score(p,stu);
    else if (c==2)
        Sort_Score_Num(p,stu);
    else if (c==3)
        Sort_Score_Sum(p,stu);
    else if (c==4)
        Delete_score_Num(p,stu);
    else printf("输入有误, 请重新输入!\n");
    printf("执行增加学生信息请输入1, 按学号排序请输入2, 按总成绩排序输入");
    scanf("%d",&c);
}
}
```

## 6 测试和结果

## 7 用户手册