

《数据结构》课程实验报告

2021-2022 学年第 1 学期

| | |
|------|------------|
| 姓名 | Banban |
| 学号 | |
| 班级 | |
| 专业 | 计算机科学与技术 |
| 手机 | |
| 实验地点 | |
| 任课教师 | |
| 学院 | 计算机科学与技术学院 |

实验一 线性表

一. 实验目的

1. 掌握线性表的两类存储结构（顺序存储结构和链式存储结构）的描述方法。
2. 掌握在顺序结构中实现查找、插入、删除操作的基本方法。
3. 掌握在各种链表结构中实现查找、插入、删除操作的基本方法。

二. 实验内容.

用三种存储结构（顺序表，链表，静态链表）求解 josuphu 问题；设有 n 个人围坐在一圈，现从指定的第 i 个人开始报数，数到第 m 个人出列，然后从出列的下一个人重新开始报数，数到第 m 个人又出列，如此重复，直到所有的人全部出列为止。

三. 实验要求

1. 上机前，针对实验内容，认真设计算法。
2. 上机过程中，能够熟练运用高级语言的程序调试器 **DEBUG** 调试程序。
3. 上机后，认真整理源程序及其注释，完成实验报告（包括源程序、实验结果、算法分析、心得体会等）。

四、顺序表，链表，静态链表 c 语言数据结构定义

顺序表：

```
typedef struct node{
    datatype data[MAXSIZE];
    int length;
}seqlist, *pseqlist;
```

静态链表：

```
typedef struct {
    datatype data;
    int next;
}component;
```

链表：

```
typedef struct node{
    datatype data;
    struct node *next;
}lnode,*linklist;
```

双向链表：

```
typedef struct line {
    struct line *prior,*next;
    int data;
}line;
```

五、源程序及运行结果（截图）

顺序表：

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```

#include<stdio.h>
#include<stdlib.h>
# define MAXSIZE 100
typedef int datatype;

typedef struct node{
    datatype data[MAXSIZE];
    int length;
}seqlist, *pseqlist;

pseqlist init(int num)
{//创建顺序表
    pseqlist p;
    p = (pseqlist)malloc(sizeof(seqlist));
    if(!p)
    {
        printf("分配失败! \n");
        exit(-1);
    }
    p->length = 0;
    for(int i=0;i<num;i++)
    {
        p->data[i]=i+1;
        p->length++;
    }
    return p;
}

int delete(pseqlist p, int d)
{//删除顺序表元素
    int i;
    if(!p->length)
    {
        printf("顺序表为空，无法删除! \n");
        return -1;
    }
    if(d<1||d>p->length)
    {
        printf("删除位置有错，无法删除! \n");
        return 0;
    }
    for(i=d;i<p->length;i++)
        p->data[i-1]=p->data[i];
    p->length--;
    return 1;
}

int josephus(pseqlist p, int s, int m)
{//约瑟夫环
    int a,i;
    if(!p->length)
    {
        printf("表中无元素\n");
        return 0;
    }
    a=s-1;//a下标从0开始
    printf("输出约瑟夫序列:");
    for(i=p->length;i>1;i--)
    {
        a=(a+m-1)%p->length;//下一个序列下标
        printf(" %d",p->data[a]);
        delete(p,a+1);
    }
    a=(a+m-1)%p->length;
    printf("\n最后剩下的数是: %d\n",p->data[a]);
    return 1;
}

int main()
{
    int s,m,num;
    pseqlist p;
    printf("请输入你要生成的顺序表长度:");
    scanf("%d", &num);
    p=init(num);
    printf("请输入从s处开始，第m个去除:");
    scanf("%d%d",&s,&m);
    josephus(p,s,m);
    return 0;
}

```

```
请输入你要生成的顺序表长度:8
请输入从s处开始,第m个去除:1 3
输出约瑟夫序列: 3 6 1 5 2 8 4
最后剩下的数是: 7
Program ended with exit code: 0
```

All Output

单链表:

```
#include <stdio.h>
#include <stdlib.h>
typedef int datatype;

typedef struct node{
    datatype data;
    struct node *next;
}Node,*linklist;

linklist creat_linklist(int num)
{//创建新的空单链表
    linklist p = NULL,head,q;
    int i=1;
    head=(linklist)malloc(sizeof(Node));
    p = head;
    while(i<=num)
    {
        q=(linklist)malloc(sizeof(Node));
        q->data=i++;
        p->next=q;
        p=q;
    }
    p->next=head->next;
    free(head);
    return p->next;
}

int josephus(linklist h,int s,int m)
{//约瑟夫环,链表指针,从第s元素开始,删除m倍数的值
    linklist p=h,temp;
    if(!h)
    {
        printf("表中无元素");
        return 0;
    }
    for(int j=1;j<s;j++)
        p=p->next;
    printf("输出约瑟夫序列:");
    int i=1;
    while(p!=p->next)
    {
        if(i<m)
        {
            p=p->next;
            i++;
        }
        printf("%2d", p->next->data);
        temp=p->next;
        p->next=temp->next;
        free(temp);
        p=p->next;
        i=1;
    }
    printf("\n最后剩下的数是: %d\n", p->data);
    return 1;
}

int main() {
    linklist q;
    int s,m,x;
    printf("请输入你要生成的单链表长度: ");
    scanf("%d",&x);
    q=creat_linklist(x);
    printf("请输入从s处开始,删除第m个元素: ");
    scanf("%d%d",&s,&m);
    josephus(q, s, m);
}
```

注意不要雷同

banban

<https://github.com/dream4789/Computer-learning-resources.git>

```
return 1; }
```

```
请输入你要生成的单链表长度: 8
请输入从s处开始, 删除第m个元素: 1 3
输出约瑟夫序列: 3 6 1 5 2 8 4
最后剩下的数是: 7
Program ended with exit code: 1
```

静态链表:

```
#include<stdio.h>
#define maxsize 100
struct node
{
    int data;
    int next;
}str[maxsize];
void josephus(int num,int s,int m)
{
    int f,i,j,x,y;
    j=1;f=0;
    int rear=0;
    for(i=0;i<num;i++)
    {
        str[i].data=i+1;
        str[i].next=i+1;
    }
    str[num-1].next=0;
    x=num;
    for(i=0;i<(s-1);i++)
        rear=str[rear].next;
    while(f<x)
    {
        for(i=0;i<(m-1);i++)
            rear=str[rear].next;
        printf("%2d",str[rear].data);
        y=rear;
        for(i=rear+1;i<num;i++)
        {
            str[y].data=str[y+1].data;
            if(i==num-1)
                str[y].next=0;
            y++;
        }
        if(rear+1==num)
        {
            str[rear].next=0;
            rear=0;
        }
        num--;
        f++;
    }
}
int main()
{
    int num,s,m;
    printf("请输入你要生成的静态链表长度:");
    scanf("%d", &num);
    printf("请输入从s处开始, 第m个去除:");
    scanf("%d%d",&s,&m);
    printf("输出约瑟夫序列:");
    josephus(num,s,m);
    return 0;
}
```

```
请输入你要生成的静态链表长度:8
请输入从s处开始, 第m个去除:1 3
输出约瑟夫序列: 3 6 1 5 2 8 8 7Program
```

All Output ↕

六、小结与体会

顺序表、链表都属于线性表，都可以采用两种基本的存储结构：顺序存储结构和链式存储结构来存储。结构中的元素之间存在一对一的线性关系。

线性结构是最常用、最简单的一种数据结构。而线性表是一种典型的线性结构。其基本特点是线性表中的数据元素是有序且是有限的。

在这种结构中我发现一些规律：

- ① 存在一个唯一的被称为“第一个”的数据元素；
- ② 存在一个唯一的被称为“最后一个”的数据元素；
- ③ 除第一个元素外，每个元素均有唯一的一个直接前驱；
- ④ 除最后一个元素外，每个元素均有唯一的一个直接后继。

我对顺序表和链表的使用更加熟悉了，它的创建，取顶指针，插入，删除……在这次上机中更加熟练的操作。这次数据结构的上机实验，我做了顺序线性表的数值转换和约瑟夫问题。其中的顺序链表约瑟夫问题比较顺利，花的时间相对较少，因为比较熟悉；而静态链表却出错的比较多，可能是因为知识点掌握的还不够熟悉。在之后的学习中，我要巩固自己的基础知识，多学多问。