程序设计训练实验报告

职工工资信息管理系统

学院：物理与电子工程学院
班级：物联 171
学号：1719500097
姓名：黄业广
时间：2018/4/14

目录

程序设计训练实验报告

（一）题目：职工工资信息管理系统

（二）系统功能模块结构图

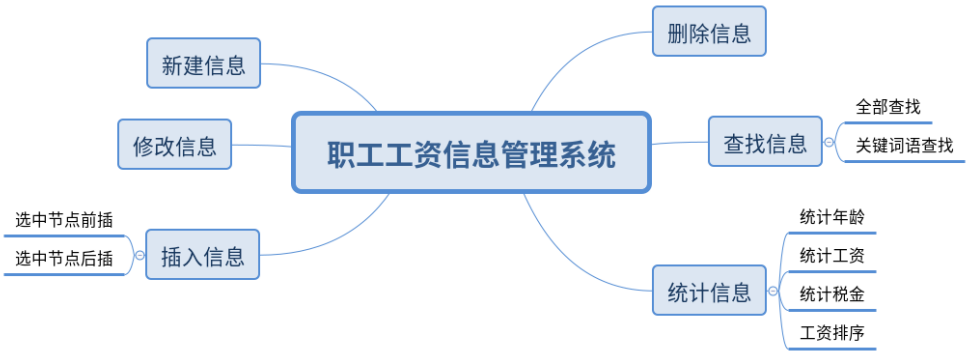

图 1　系统功能模块结构图

（三）各模块的功能

新建信息模块：检测是否已经存在文件，若没有存在文件则新建信息并保存到文件，若存在文件，则询问是否新建文件，若要新建文件则新建文件并保存，若不新建文件则退出。

修改信息模块：检测内存中是否存在数据链表，若已经存在则进行修改信息，若没有存在则引导至新建信息模块，修改信息时询问用户需要修改哪一部分的信息。

插入信息模块：检测内存中是否存在数据链表，若已经存在则进行插入信息，若没有存在则引导至新建信息模块，进行插入信息时询问用户要使用前插还是后插。

删除信息模块：检测内存中是否存在数据链表，若已经存在则进行删除信息，若没有存在则引导至新建信息模块。

查找信息模块：检测内存中是否存在数据链表，若已经存在则进行查找信息，若没有攒在则引导至新建信息模块，查找时询问用户需要全部查询还是关键词语

查询。

统计信息模块 :检测内存中是否存在数据链表，若已经存在则进行统计信息，若没有存在则引导至新建信息模块，统计时询问用户需要哪种统计方法，提供了 4 种统计方法，分别是统计年龄，统计工资，统计税金，按照应发工资排序，然后进行统计。

（四）数据结构设计及用法说明：

本程序用了双链表以及文件处理，在内存中的数据是以双链表的形式存在，然后保存在硬盘中是以二进制文件保存的，本程序之所以要用双链表的原因是因为双链表进行删除以及插入的操作比较简单，同时进行统计的操作相比单链表简单，所以使用双链表，使用二进制文件的原因是因为存入读取速度快，可以应付大量数据写读，同时二进制文件比较安全，不会导致员工信息泄露，所以使用二进制文件进行存储。

（五）实验结果：

初始界面

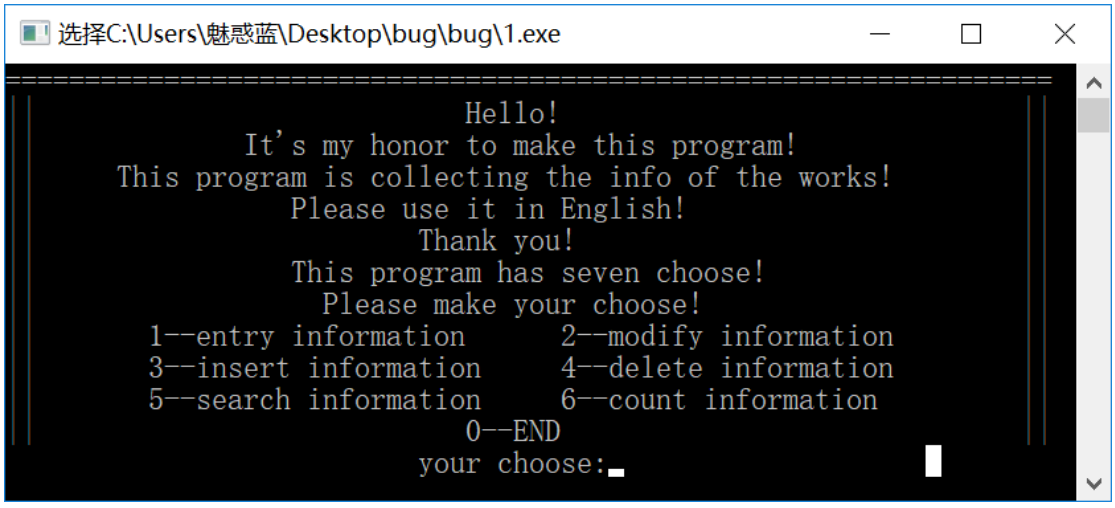

图 2　程序运行初始界面

进行修改功能

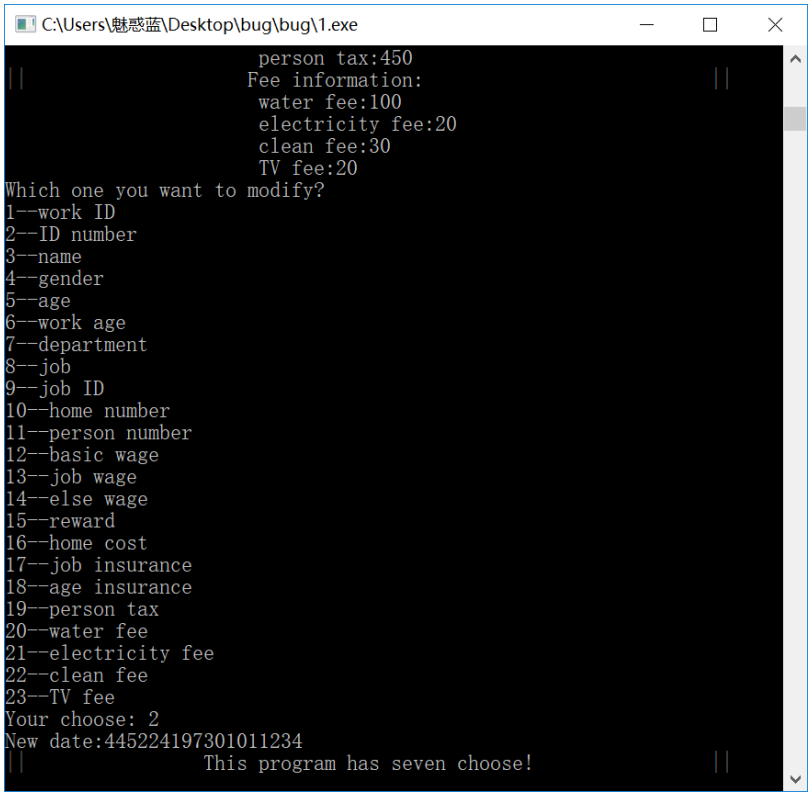

图 3　修改功能界面

进行插入功能

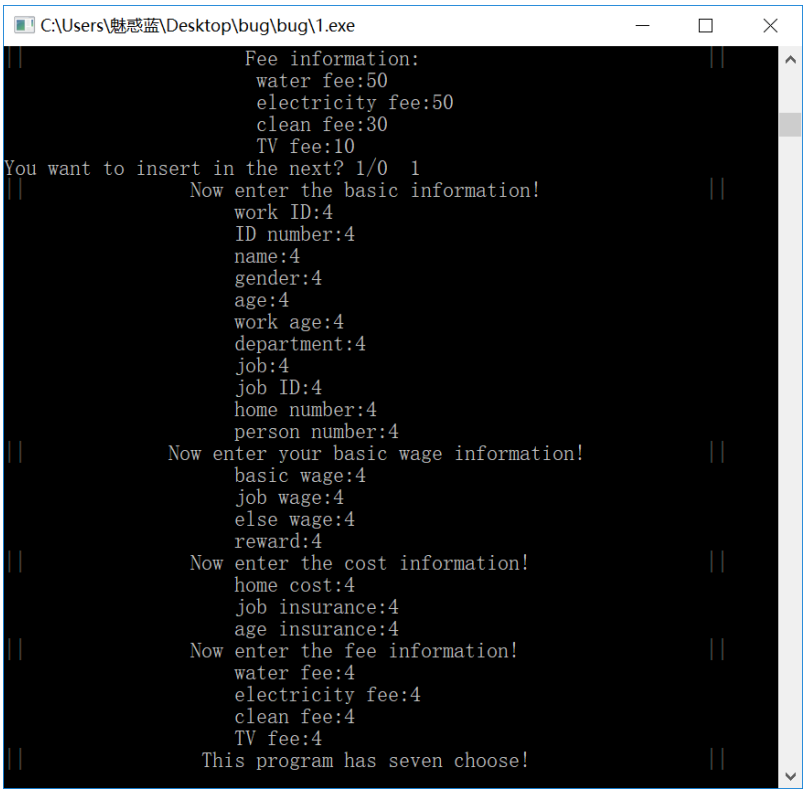

图 4　插入功能界面

进行删除功能

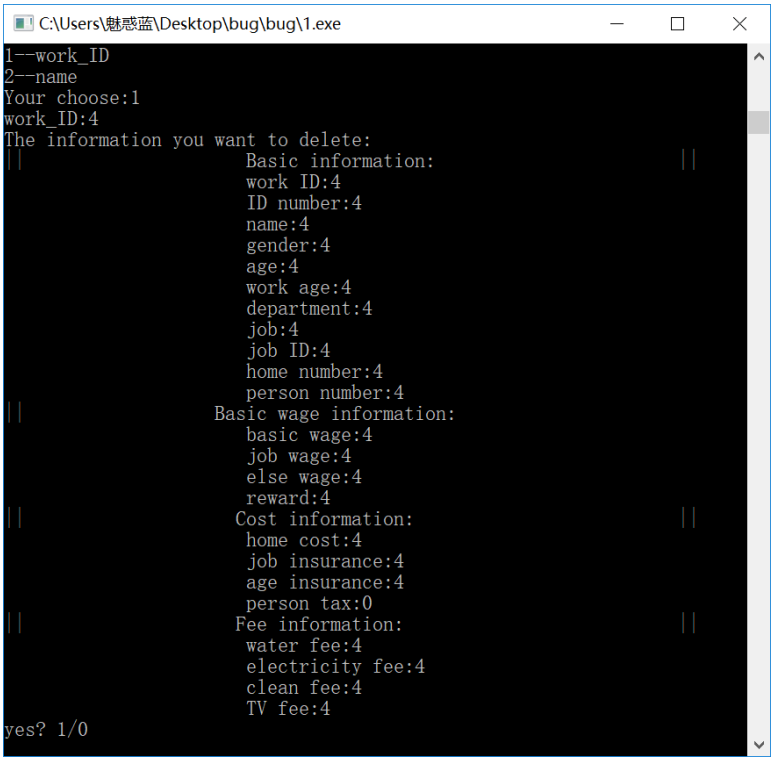

图 5　删除功能界面

进行查找功能

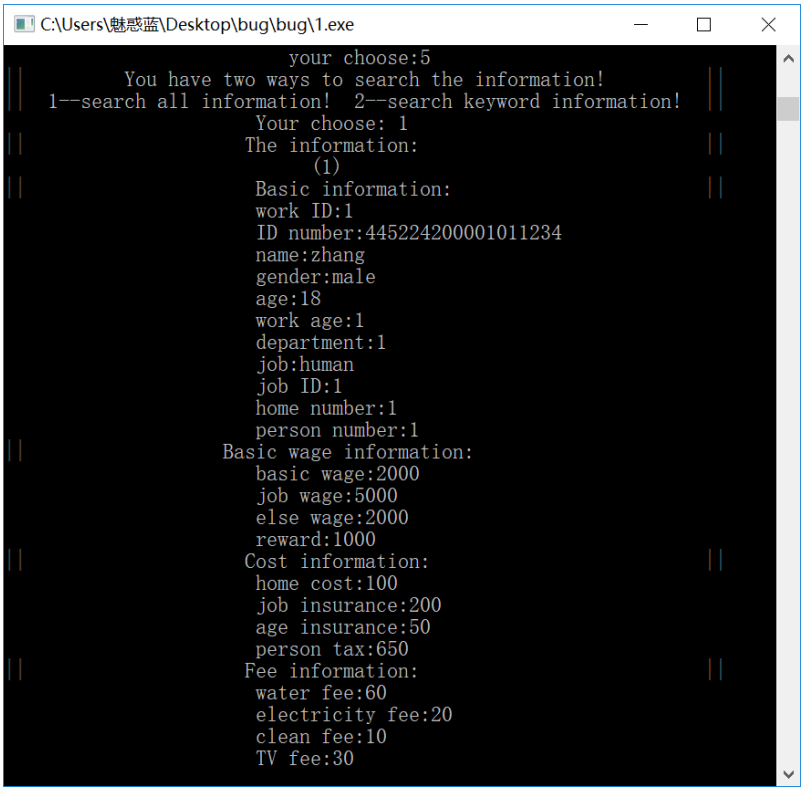

图 6-1　查找功能界面

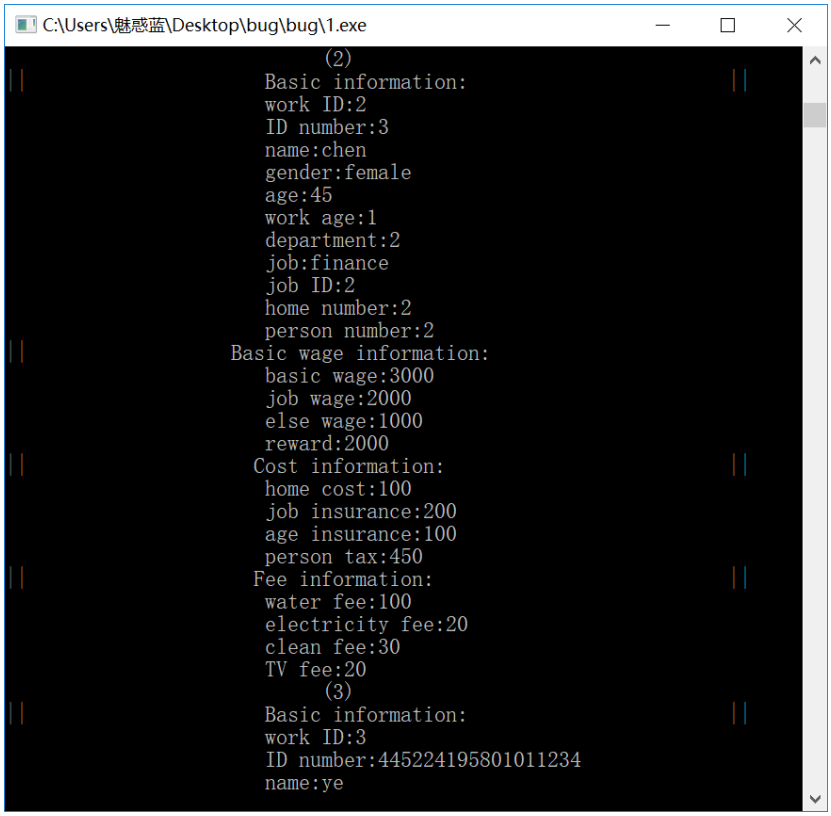

图 6-2　查找功能界面

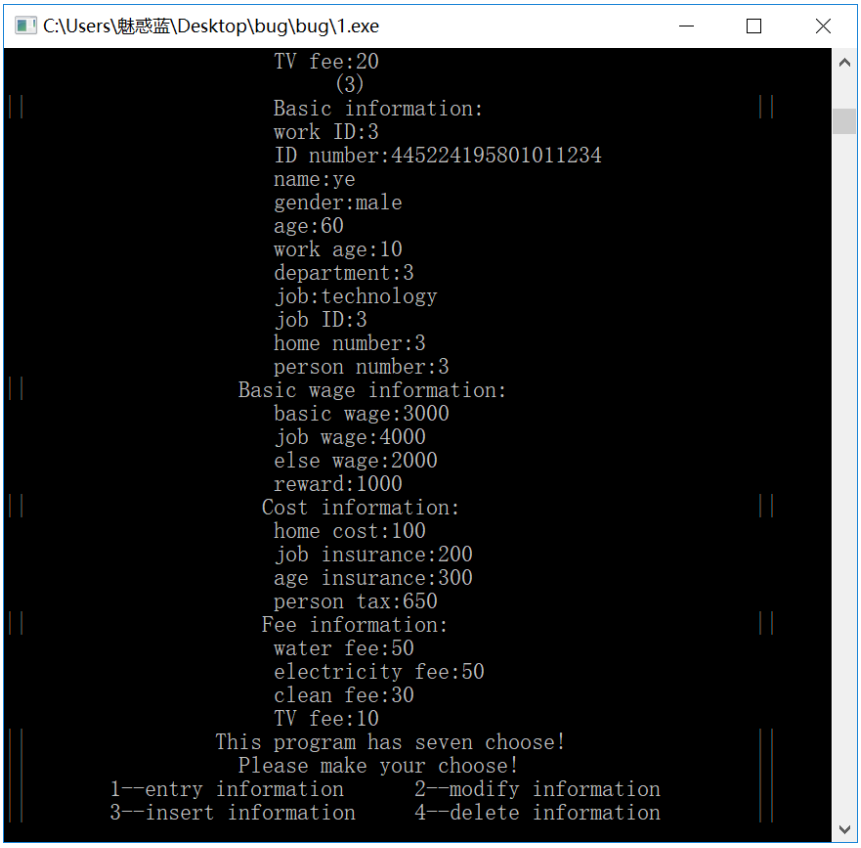

图 6-3　查找功能界面

进行统计功能

统计年龄

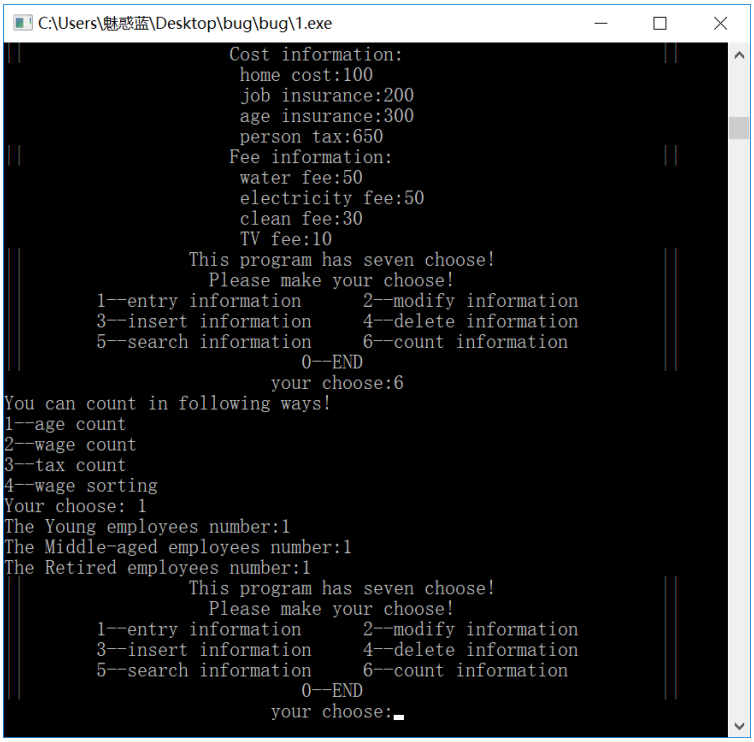

图 7-1　统计功能统计年龄界面

统计工资

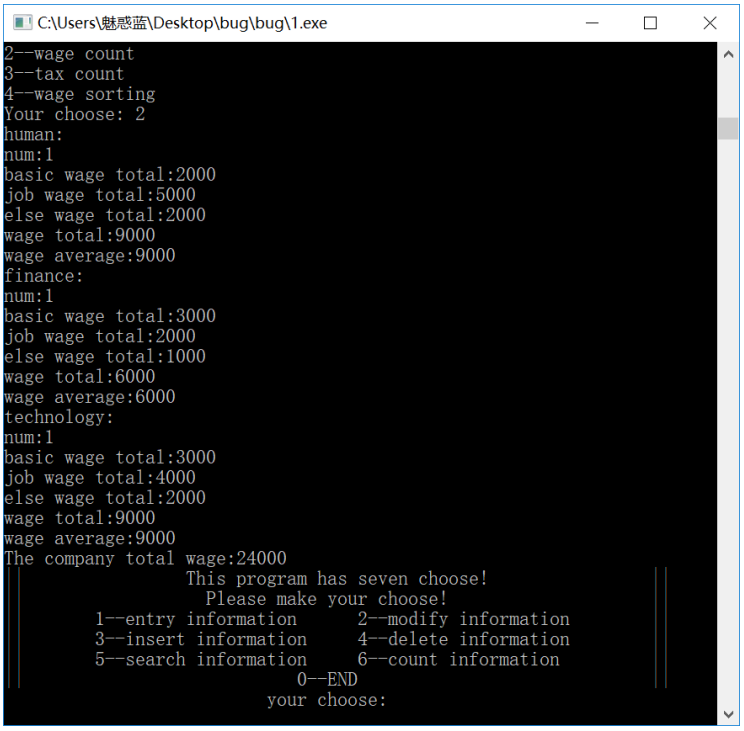

图 7-2　统计功能统计工资界面

统计税金

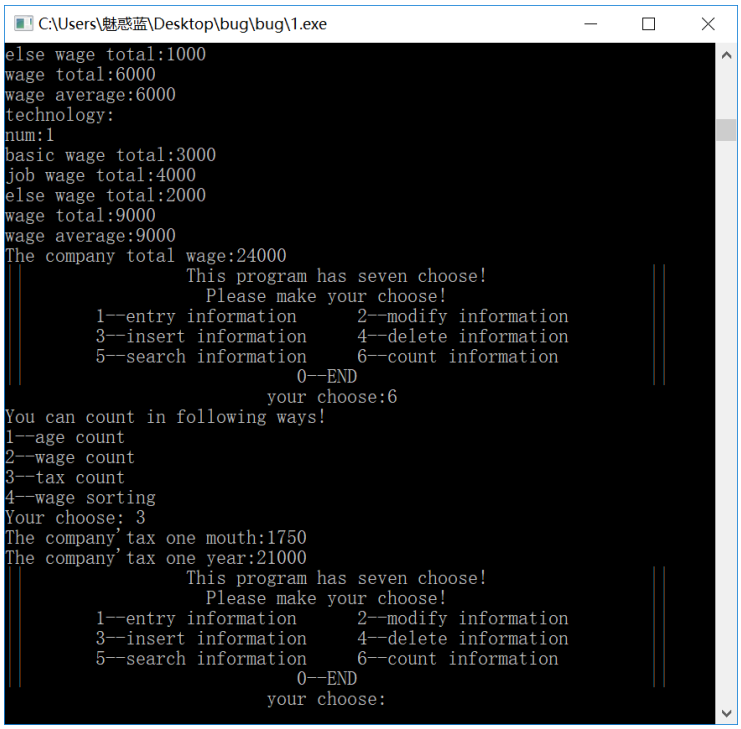

图 7-3　统计功能统计税金界面

按照应发工资排序

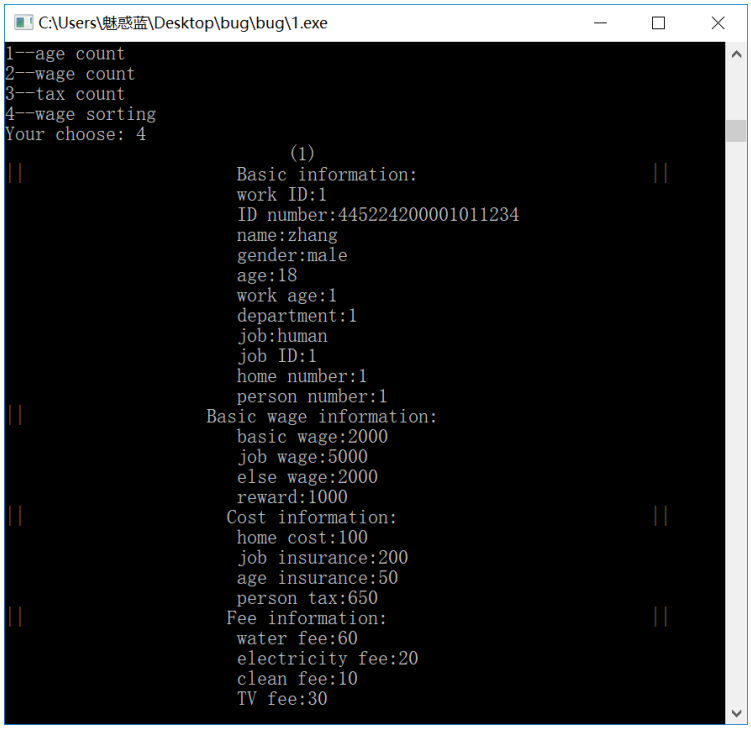

图 8-1　统计功能按照应发工资排序

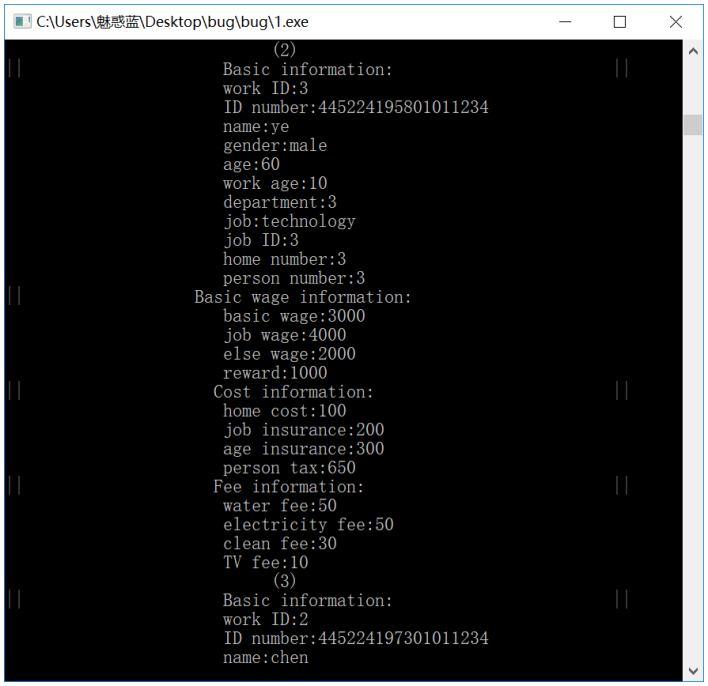

图 8-2　统计功能按照应发工资排序

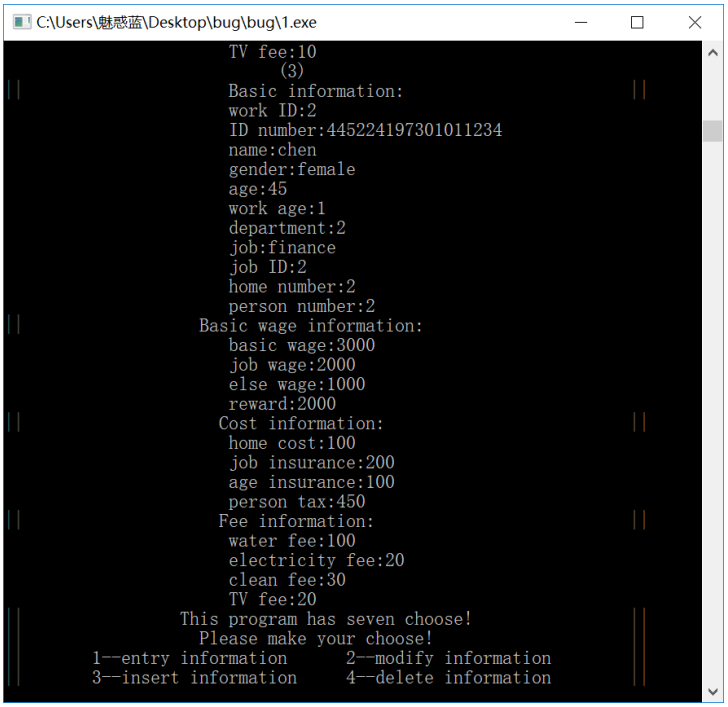

图 8-3　统计功能按照应发工资排序

（六）体会

通过了本次程序设计训练，对计算机原理，以及 C 语言的内存申请空间，以及指针的应用有了更加深刻一层的体会，同时也对双链表这个数据结构更加熟悉。

（七）参考文献

何钦铭，颜晖. 《C 语言程序设计（第 3 版）》.北京：高等教育出版社，2015

CSDN 博客：https://blog.csdn.net/lchunli/article/details/4786663

（八）附录

```
void new_list(info head)
{
    info p,q;
    p=head;
    int a;
    do
    {
        q=(info)malloc(sizeof(infor));
        printf("||                    Please enter info!                    ||\n");
        sc(q);
        p->next=q;
        q->prior=p;
        p=q;
        p->next=NULL;
        printf("                    Continue? 1/0    ");
        scanf("%d",&a);
    }while(a==1);
}//建立新的数据链表

int read_file(info head)
{
    info p=head,q;
    FILE *fp;
    if((fp=fopen("initial_list.dat","rb"))==NULL) return 0;//如果读取文件为空,则没有创建过数据文件,返回 0
    else
    {
        fp=fopen("initial_list.dat","rb");
        while(!feof(fp))
        {
            q=(info)malloc(sizeof(infor));
```

```
            fread(q,sizeof(infor),1,fp);
            p->next=q;
            q->prior=p;
            q->next=NULL;
            p=q;
        }
        p->prior->next=NULL;
        free(p);
        fclose(fp);
        return 1;//如果已经创建过数据文件,则读取到内存中,然后返回 1
    }
}//读取文件中的数据

void save_file(info head)
{
    info p=head;
    FILE *fp;
    fp=fopen("initial_list.dat","wb+");
    while(p->next!=NULL)
    {
        p=p->next;
        fwrite(p,sizeof(infor),1,fp);
    }
    fclose(fp);
}//把内存中的数据保存到文件

info search_key(info head)
{
    info p=head;
    int choose;
    printf("The keyword:\n1--work_ID\n2--name\nYour choose:");
    scanf("%d",&choose);//选择以哪种关键字来查询
    if(choose==1)
    {
        char a[80];
        printf("work_ID:");
        scanf("%s",a);
        while(p->next!=NULL)
        {
            p=p->next;
            if(!strcmp(a,p->basic.work_ID)) return p;//如果查询成功,则返回指针
        }
        if(p->next==NULL) return NULL;//如果查询失败,返回空
    }
```

```
    else if(choose==2)
    {
        char b[80];
        printf("name:");
        scanf("%s",b);
        while(p->next!=NULL)
        {
            p=p->next;
            if(!strcmp(b,p->basic.name)) return p;//如果查询成功,则返回指针
        }
        if(p->next==NULL) return NULL;//如果查询失败,返回空
    }
}//用关键词语查询信息

void search_list(info head)
{
    info p=head,q;
    int choose;
    printf("||              You have two ways to search the information!              ||\n");
    printf("||   1--search all information!   2--search keyword information!   ||\n");
    printf("                                   Your choose: ");
    scanf("%d",&choose);//选择那种方式查询
    if(choose==1)
    {
        printf("||                          The information:                          ||\n");
        int a=1;
        while(p->next!=NULL)
        {
            p=p->next;
            printf("                                (%d)                                \n",a++);
            pr(p);
        }//循环链表把链表显示出来
    }
    else if(choose==2)
    {
        q=search_key(head);//接收关键词语查询信息返回的指针
        if(q!=NULL)
        {
        printf("||              The information you want to search:              ||\n");
            pr(q);//若指针不为空,显示指针指向的结构体信息
        }
        else if(q==NULL) printf("||              The keyword can't find!                  ||\n");//
若指针为空,显示找不到
    }
```

```
}//查询信息

void modify_list(info head)
{
    info q=head,p;
    printf("which one you want to modify?\nYou can search the information you want to modify in keyword!\n");
    p=search_key(q);//用关键词语查询信息查询出要修改的信息
    printf("The information you want to modify:\n");
    pr(p);//显示出来查看修改哪一数据
    printf("Which one you want to modify?\n");
    printf("1--work ID\n2--ID number\n3--name\n4--gender\n5--age\n6--work age\n7--department\n8--job\n9--job ID\n10--home number\n11--person number\n12--basic wage\n13--job wage\n14--else wage\n15--reward\n16--home cost\n17--job insurance\n18--age insurance\n19--person tax\n20--water fee\n21--electricity fee\n22--clean fee\n23--TV fee\nYour choose: ");
    int a;
    scanf("%d",&a);
    printf("New date:");
    switch(a)
    {
        case 1 :scanf("%s",p->basic.work_ID);break;
        case 2 :scanf("%s",p->basic.ID_number);break;
        case 3 :scanf("%s",p->basic.name);break;
        case 4 :scanf("%s",p->basic.gender);break;
        case 5 :scanf("%d",&p->basic.age);break;
        case 6 :scanf("%d",&p->basic.work_age);break;
        case 7 :scanf("%s",p->basic.department);break;
        case 8 :scanf("%s",p->basic.job);break;
        case 9 :scanf("%d",&p->basic.job_ID);break;
        case 10:scanf("%s",p->basic.home_number);break;
        case 11:scanf("%s",p->basic.person_number);break;
        case 12:scanf("%d",&p->wage.basic_wage);break;
        case 13:scanf("%d",&p->wage.job_wage);break;
        case 14:scanf("%d",&p->wage.else_wage);break;
        case 15:scanf("%d",&p->wage.reward);break;
        case 16:scanf("%d",&p->cost.home_cost);break;
        case 17:scanf("%d",&p->cost.job_insurance);break;
        case 18:scanf("%d",&p->cost.age_insurance);break;
        case 19:scanf("%d",&p->cost.person_tax);break;
        case 20:scanf("%d",&p->fee.water_fee);break;
        case 21:scanf("%d",&p->fee.electricity_fee);break;
        case 22:scanf("%d",&p->fee.clean_fee);break;
        case 23:scanf("%d",&p->fee.TV_fee);break;
    }//修改数据
```

```
}//修改信息

void delete_list(info head)
{
    info p=head,q;
    printf("Search the information you want to delete in keyword!\n");
    q=search_key(p);//用关键词语查询信息查询出要删除的信息
    if(q==NULL) printf("The key can't find!\n");
    else if(q!=NULL)
    {
        printf("The information you want to delete:\n");
        pr(q);
        printf("yes? 1/0    ");//是否确定删除
        int a;
        scanf("%d",&a);
        if(a==1)
        {
            if(q->next!=NULL)
            {
                q->prior->next=q->next;
                q->next->prior=q->prior;
                free(q);
            }
            else
            {
                q->prior->next=q->next;
                q->prior=NULL;
                free(q);
            }
        }//删除数据
    }

}//删除信息

void insert_list(info head)
{
    info p=head,q,l;
    printf("Search the information your want to insert in keyword!\n");
    q=search_key(p);//用关键词语查询要添加的信息的节点位置
    if(q==NULL) printf("The key can't find!\n");
    else if(q!=NULL)
    {
        printf("The information you want to insert:\n");
        pr(q);
```

```
            printf("You want to insert in the next? 1/0    ");//询问要添加在节点前面还是后面
            int a;
            scanf("%d",&a);
            if(a==1)
            {
                if(q->next!=NULL)
                {
                    l=(info)malloc(sizeof(infor));
                    sc(l);
                    l->next=q->next;
                    l->prior=q;
                    q->next->prior=l;
                    q->next=l;
                }//非最后一个的添加
                else
                {
                    l=(info)malloc(sizeof(infor));
                    sc(l);
                    q->next=l;
                    l->prior=q;
                    l->next=NULL;
                }//最后一个的添加
            }//添加后面
            else if(a==0)
            {
                l=(info)malloc(sizeof(infor));
                sc(l);
                l->prior=q->prior;
                l->next=q;
                q->prior->next=l;
                q->prior=l;
            }//添加前面
        }
}//插入信息

int free_list(info head)
{
    info p=head;
    int choose;
    printf("||                You have entered the information already!                ||\n");
    printf("|| Do you enter new information by clearing the old information!||\n");
    printf("                                Your choose:(1/0)    ");
    scanf("%d",&choose);//询问是否销毁链表
    if(choose==1)
```

```
    {
        p->next->prior=NULL;
        p->next=NULL;
        printf("Your old information is cleared!\n");
        return 1;//销毁链表返回 1
    }
    else if(choose==0) return 0;//不销毁链表返回 0
}//销毁链表

void count_list(info head)
{
    info p=head;
    int choose;
    printf("You can count in following ways!\n1--age count\n2--wage count\n3--tax count\n4--wage sorting\nYour choose: ");
    scanf("%d",&choose);//选择哪种统计
    switch(choose)
    {
        case 1 :
            {
                int a=0,b=0,c=0;
                while(p->next!=NULL)
                {
                    p=p->next;
                    if((p->basic.age)<20) a++;
                    else if((p->basic.age)>=20&&(p->basic.age)<=55) b++;
                    else if((p->basic.age)>55) c++;
                }
                printf("The Young employees number:%d\nThe Middle-aged employees number:%d\nThe Retired employees number:%d\n",a,b,c);
                break;
            }//统计人数
        case 2 :
            {
                int basic_wage1=0,job_wage1=0,else_wage1=0,wage_total1=0;
                int basic_wage2=0,job_wage2=0,else_wage2=0,wage_total2=0;
                int basic_wage3=0,job_wage3=0,else_wage3=0,wage_total3=0;
                int d=0,e=0,f=0;
                while(p->next!=NULL)
                {
                    p=p->next;
                    if(!strcmp(p->basic.job,"human"))
                    {
                        basic_wage1=basic_wage1+p->wage.basic_wage;
```

```
                job_wage1=job_wage1+p->wage.job_wage;
                else_wage1=else_wage1+p->wage.else_wage;
                wage_total1=wage_total1+basic_wage1+job_wage1+else_wage1;
                d++;
            }//统计工资同时统计人数
            else if(!strcmp(p->basic.job,"finance"))
            {
                basic_wage2=basic_wage2+p->wage.basic_wage;
                job_wage2=job_wage2+p->wage.job_wage;
                else_wage2=else_wage2+p->wage.else_wage;
                wage_total2=wage_total2+basic_wage2+job_wage2+else_wage2;
                e++;
            }//统计工资同时统计人数
            else if(!strcmp(p->basic.job,"technology"))
            {
                basic_wage3=basic_wage3+p->wage.basic_wage;
                job_wage3=job_wage3+p->wage.job_wage;
                else_wage3=else_wage3+p->wage.else_wage;
                wage_total3=wage_total3+basic_wage3+job_wage3+else_wage3;
                f++;
            }//统计工资同时统计人数
        }
        printf("human:\nnum:%d\nbasic wage total:%d\njob wage total:%d\nelse wage total:%d\nwage total:%d\nwage average:%d\n",d,basic_wage1,job_wage1,else_wage1,wage_total1,wage_total1/d);
        printf("finance:\nnum:%d\nbasic wage total:%d\njob wage total:%d\nelse wage total:%d\nwage total:%d\nwage average:%d\n",e,basic_wage2,job_wage2,else_wage2,wage_total2,wage_total2/e);
        printf("technology:\nnum:%d\nbasic wage total:%d\njob wage total:%d\nelse wage total:%d\nwage total:%d\nwage average:%d\n",f,basic_wage3,job_wage3,else_wage3,wage_total3,wage_total3/f);
        printf("The company total wage:%d\n",wage_total1+wage_total2+wage_total3);
        break;
    }//统计工资
case 3 :
    {
        int total,tax,tax_total=0,tax_year;
        while(p->next!=NULL)
        {
            p=p->next;

total=(p->wage.basic_wage)+(p->wage.else_wage)+(p->wage.job_wage)+(p->wage.reward);
            if(total<3500) tax=0;
            else tax=(total-3500)*0.1;
```

```
                tax_total=tax_total+tax;
            }//统计一个月的税金
            tax_year=tax_total*12;//统计一年的税金
            printf("The company'tax one mouth:%d\nThe company'tax one year:%d\n",tax_total,tax_year);
            break;
        }//统计税金
    case 4 :
        {
            info head1=creat_head();
            info head_max=creat_head();
            info n=head_max;
            info l=head_max;
            info m=head1;
            info q,max;
            info t=head;
            int a=1;
            while(t->next!=NULL)
            {
                t=t->next;
                q=(info)malloc(sizeof(infor));
                *q=*t;
                m->next=q;
                q->prior=m;
                q->next=NULL;
                m=q;
            }//复制一个链表
            while(head1->next!=NULL)
            {
                max=head1->next;
                q=head1->next;
                while(q!=NULL)
                {

if(((max->wage.basic_wage)+(max->wage.else_wage)+(max->wage.job_wage)+(max->wage.reward))<((q->wage.basic_wage)+(q->wage.else_wage)+(q->wage.job_wage)+(q->wage.reward)))
                        max=q;
                    q=q->next;
                }
                m=max;//找出链表中应发工资的最大值
                if(max->next!=NULL)
                {
                    max->prior->next=max->next;
                    max->next->prior=max->prior;
```

```
                        }
                        else
                        {
                            max->prior->next=max->next;
                            max->prior=NULL;
                        }
                        n->next=m;
                        m->prior=n;
                        m->next=NULL;
                        n=m;//将最大值的数据依次接入链表头节点
                    }
                    free(head1);
                    while(l->next!=NULL)
                    {
                        l=l->next;
                        printf("                                                                    (%d)
\n",a++);
                        pr(l);
                    }//把排序后的链表显示出来
                    head_max->next->prior=NULL;
                    head_max->next=NULL;//销毁排序后链表
                    break;
                }//按应发工资排序
        }
}//统计

int main()
{
    info head;
    head=creat_head();//定义头结点

printf("=================================================================
==\n");
    printf("||                              Hello!                              ||\n");
    printf("||                  It's my honor to make this program!             ||\n");
    printf("||        This program is collecting the info of the works!         ||\n");
    printf("||                      Please use it in English!                   ||\n");
    printf("||                              Thank you!                          ||\n");
    int choose,i=0;
    i=read_file(head);//检测是否存在文件,如果存在则读取进内存
    while(1)//循环让用户选择功能
    {
        choose=user_choose();
        switch(choose)
```

```
{
    case 1 :if(i==0)
            {
                new_list(head);
                save_file(head);
                i=1;
                break;
            } //如果没有存在文件,则新建数据并保存
            else if(i==1) //如果存在文件
            {
                int a;
                a=free_list(head);//询问是否要销毁,然后重新创建
                if(a==0) break;//不重新创建
                else if(a==1)
                {
                    new_list(head);
                    save_file(head);
                    break;
                }//重新创建并保存到文件
            }
    case 2 :if(i==0)
            {
                printf("||                You haven't entered the information!                ||\n");
                printf("||                Please enter the information firstly!               ||\n");
                break;
            }//如果没有存在文件,引导至新建数据
            else
            {
                modify_list(head);
                save_file(head);
                break;
            }//若存在,修改数据并保存文件
    case 3 :if(i==0)
            {
                printf("||                You haven't entered the information!                ||\n");
                printf("||                Please enter the information firstly!               ||\n");
                break;
            }//如果没有存在文件,引导至新建数据
            else
            {
                insert_list(head);
                save_file(head);
                break;
            }//若存在数据,则进行插入操作
```

```
            case 4 :if(i==0)
                    {
                        printf("||                You haven't entered the information!                ||\n");
                        printf("||                Please enter the information firstly!               ||\n");
                        break;
                    }//如果没有存在文件,引导至新建数据
                    else
                    {
                        delete_list(head);
                        save_file(head);
                        break;
                    }//若存在数据,则进行删除操作
            case 5 :if(i==0)
                    {
                        printf("||                You haven't entered the information!                ||\n");
                        printf("||                Please enter the information firstly!               ||\n");
                        break;
                    }//如果没有存在文件,引导至新建数据
                    else
                    {
                        search_list(head);
                        save_file(head);
                        break;
                    }//若存在数据,则进行查找操作
            case 6 :if(i==0)
                    {
                        printf("||                You haven't entered the information!                ||\n");
                        printf("||                Please enter the information firstly!               ||\n");
                        break;
                    }//如果没有存在文件,引导至新建数据
                    else
                    {
                        count_list(head);
                        break;
                    }//若存在数据,则进行统计操作
            case 0 :exit(0);//退出程序
        }
    }
    return 0 ;
}
```