

# Динамические структуры данных

## 1. Списки

### 1) описание:

```
type ptr = ^element;
element = record
    key:integer;
    next:ptr
end;

var p,q,l:ptr;
```

### 2) создание списка длины n: {в p ссылка на голову списка}

```
read(n); | p:=nil; read(n);
if n>0 then begin new(l);read(i); | while n>0 do
    l^.key:=i;p:=l;n:=n-1; | begin
    while n>0 do | new(q); read(q^.key);
    begin | q^.next:=p;
        new(q);read(i);q^.key:=i; | p:=q; n:=n-1
        l^.next:=q;l:=q;n:=n-1 | end
    end; |
    l^.next:=nil |
end else begin p:=nil;l:=nil end |
```

### 3) подсчет суммы элементов списка (пример обхода списка)

```
sum:=0;
q:=p;
while q<>nil do
begin
    sum:=sum+q^.key;
    q:=q^.next
end
```

### 4) включение в список элемента q

```
после элемента с содержимым k: | перед элементом с содержимым k:
p1:=p; | p1:=p;
while (p1<>nil) and (p1^.key<>k) do | while (p1<>nil) and (p1^.key<>k) do
    p1:=p1^.next; | p1:=p1^.next;
if p1<>nil then | if p1<>nil then
begin new(q); read(i);q^.key:=i; | begin new(q); read(i);q^:=p1^;
    q^.next:=p1^.next; | p1^.key:=i;
    p1^.next:=q; | p1^.next:=q;
end | end
```

### 5) Типичная ошибка при работе со списками:

```
while p1<>nil do
begin p1:=p1^.next; write(p1^.key) end;
```

### 6) Удаление элемента с ключем k из списка:

```
if p<>nil then
    if p^.key=k then begin q:=p; p:=p^.next end
    else
    begin p1:=p;
        while (p1^.next<>nil) and (p1^.next^.key<>k) do
            p1:=p1^.next;
        if p1^.next<>nil then
            begin
                q:=p1^.next;
                p1^.next:=p1^.next^.next
            end
    end;
dispose(q)
```

```

typedef struct element {
    int key;
    struct element *next;} *LIST;
LIST l,p,p1,q;
void write(LIST p) //печать списка
{LIST q;
    q=p;
    while (q) {
        printf("%d ",q->key);
        q=q->next;
    }
    printf("\n");
}
// создание списка длины n: {в p ссылка на голову списка}
void main(void)
{int n,i,k;
scanf("%d",&n);
if(n>0){l=new element;scanf("%d",&i);
    l->key=i;p=l;n--;
    while (n)
        { q=new element;scanf("%d",&i);
            q->key=i;
            l->next=q;l=q;n--;
        } l->next=NULL;
} else p=l=NULL;
write(p);
// создание списка длины n: {в p ссылка на голову списка}
scanf("%d",&n);
p=NULL;
while (n>0) {
    q=new element;scanf("%d",&i);
    q->key=i; q->next=p;
    p=q; n--;
}
write(p);
//включение в список элемента q после элемента с содержимым k:
scanf("%d",&k);
p1=p;
while (p1&&(p1->key!=k)) p1=p1->next;
if (p1!=NULL) {
    q=new element; scanf("%d",&i);q->key=i;
    q->next=p1->next;
    p1->next=q;
}
// удаление элемента, равного k, из списка
scanf("%d",&k);
if (p) if (p->key==k) p=p->next; else
{p1=p;
while ((p1->next)&&(p1->next->key!=k)) p1=p1->next;
if (p1->next) p1->next=p1->next->next;
}
}

```