第 9 章 用户自己建立数据类型

一、选择题

1. A　　2. B　　3. C　　4. C　　5. C　　6. A　　7. C　　8. A　　9. B　　10. A

二、编程题

1.

```
#include <stdio.h>
#include <string.h>
#define NUMBER 10
struct book
{char name[10];
   int price;
};
main()
{ int i,max1,min1,min,max;
   struct book test[NUMBER];
   printf("\nInput 10 books' name and price:\n");
   for(i=0;i<NUMBER;i++)
   {printf("name:");
      scanf("%s",test[i].name); //使用 gets(test[i].name)更好
      printf("price:");
      scanf("%d",&(test[i].price));
   }
   max=min=test[0].price;
   max1=min1=0;
   for(i=0;i<NUMBER;i++)
   {if(max<test[i].price){max=test[i].price;max1=i;}
      if(min>test[i].price){min=test[i].price;min1=i;}
   }
   printf("\nMax Price\n");
   printf("%-11d%-10s",test[max1].price,test[max1].name);
   printf("\nMin Price\n");
   printf("%-11d%-10s",test[min1].price,test[min1].name);
}
```

2.

```
#include "stdio.h"
#include <malloc.h>
#define LEN sizeof(struct student)
struct student
{
      int num; char name[10];
```

```
    struct student *next;
};
struct student *creat()
{
    struct student *head;
    struct student *p1, *p2;
    int n=0;
    p1 = p2 = (struct student *)malloc(LEN);
    head = NULL;
    while (p1->num != 0)
    {
        n = n + 1;
        if (n == 1) head = p1;
        else p2->next = p1;
        p2 = p1;
        p1 = (struct student *) malloc(LEN);
        scanf("%d%s", &p1->num, &p1->name);
    }
    p2->next = NULL;
    return(head);
}
int main()
{
    int n, num;
    char name[10];
    struct student *p,*head;
    head =creat();
    p = head->next;
    while (p)
    {
       printf("%d, %s\n", p->num, p->name);
       p = p->next;
    }
    return 0;
}
```

3.

```
#include "stdio.h"
#include <malloc.h>
#include <string.h>
#define LEN sizeof(struct student)
struct student
{
```

```
    int num; char name[10];
    struct student *next;
};
struct student *creat()
{
    struct student *head;
    struct student *p1, *p2;
    int n=0;
    p1 = p2 = (struct student *)malloc(LEN);
    head = NULL;
    while (p1->num != 0)
    {
        n = n + 1;
        if (n == 1) head = p1;
        else p2->next = p1;
        p2 = p1;
        p1 = (struct student *) malloc(LEN);
        scanf("%d%s", &p1->num, &p1->name);
    }
    p2->next = NULL;
    return(head);
}
struct student *insert(struct student *head,int n,int num,char name[])
{
    struct student *p,*q;
    int i;
    p = head;
    for (i = 1; i <n && p->next!=NULL; i++) //若输入的 n 大于结点个数，则待插入结点作为最后一结点
        p = p->next;
    q = (struct student *)malloc(LEN);
    q->num = num;
    strcpy(q->name, name);
    q->next = p->next;
    p->next = q;
    return head;
}
int main()
{
    int n, num;
    char name[10];
    struct student *p,*head;
    head =creat();
    printf("输入待插入结点位置 n: ");
    scanf("%d",&n);
```

```
    printf("\n 输入学号和姓名: ");
    scanf("%d%s",&num,name);
    head = insert(head,n, num, name);
    printf("\n 输出结果:\n");
    p = head->next; //头结点为空
    while (p)
    {
       printf("%d, %s\n", p->num, p->name);
       p = p->next;
    }
    return 0;
}
```

4.（附加题）

（方法一）

```
#define N 13
struct person
{int number;
   int nextp;
}link[N+1];
main()
{int i,count,h;
   for(i=1;i<=N;i++)
   {if(i==N)link[i].nextp=1;
    else link[i].nextp=i+1;
      link[i].number=i;
   }
   count=0;
   h=N;
   printf("sequence that person leave the circle:\n");
   while(count<N-1)
   {i=0;
      while(i!=3)
      {h=link[h].nextp;
         if(link[h].number)i++;
      }
    printf("%4d",link[h].number);
    link[h].number=0;
    count++;
   }
   printf("\nThe last one is:");
   for(i=1;i<=N;i++)
        if(link[i].number)printf("%3d",link[i].number);
```

```
}
```
（方法二）
```
#include <stdio.h>
#define N 13
main()
{int i,j,k,a[N+1],*p;
  for(i=0,p=a;p<=a+N;i++,p++)
    *p=i;
  p=a+1;k=N;
  for(i=0,j=1;k!=1;j++)
  {if(p>(a+N))
    p=a+1;
    if(*p!=0) i++;
    if((i-3)==0)
    {*p=0;i=0;k--;}
    p++;
  }
  for(i=1;i<=N;i++)
    if(a[i]!=0)printf("The last number is %d\n",a[i]);
}
```