**TEST – 14 结构型、共用型和枚举型**

**测试 - 试题/答案**

【测试时间90分，自行批阅，不计分 】

**学习和解题要点**

1. 结构体的类型说明

在程序中如定义结构体类型：

struct st

{ int sno;

char sn[20];

float score [3];

}; **注意**：最后分号必不可少！

此时struct st是类型名（struct与 st一起用），它等效于int 这样的类型名，可以用来定义变量、数组和指针。如：struct st s1,s2,s[10],\*p；等等，si,s2为结构体变量，s[10]为结构体数组，p为可指向结构体的指针。

注意：struct st是类型名，s是变量名，程序中不要把st当成变量去用。

1. 结构体的使用

对结构体，不能整体的引用，只能对其中成员一个一个的引用，引用时要以“结构体变量名.成员名”的形式，当某个结构型变量的内层成员又是一个结构型时，要用“外层结构体变量名.外层成员名.内层成员名” 的形式。在输入输出或其他运算时，引用的规则完全依照该成员的数据类型而定。

1. 对结构体中的数组成员，要按其数组的规则使用，对数值型的要一个元素一个元素使用，如：s1.score[i] ，s1.sn[i] ，对字符型的可以整体当成字符串使用。如：s1.sn
2. 对结构体数组，要按一般数值数组的原则使用，不能整体使用数组名。

如s[i].sno表示第i个结构体的sno；s[i[].sn[j]表示第i个结构体的sn字符串的第j个字符，s[i].sn表示第i个结构体的字符串sn等等。

1. 对结构类型指针可以指向单个的结构体也可以指向结构数组。如：p=&si；或p=s；等。

这时对结构体成员的引用可用“(\*p).成员名”或“p→成员名”。

结构类型指针当指向数组时，注意下面的区别：

p→sno++ 等价于 (p→sno)++ 得到p指向的结构体中成员sno值，用完后该值加1。

++p→sno 等价于 ++(p→sno） 得到p指向的结构体中成员sno值，先加1再使用。

（++p）→sno 先使p加1，（指向下一个元素），再取该元素的sno

（p++）→sno 先得到p所指的元素son值，然后p加1（即指向下一个元素）

1. 当结构类型的成员中有自己类型的指针成员时，一般用作链表处理，即把一个一个的结构体用指针相链。

例如：struct st

{ int sono;

char sn[20];

sturct st \*next;

} \*head, \*p; 可做成如下链表：

─→

─→

─→

─→

...─→

─→

head

↑p

NULL

next

处理链表时，对于如p=p→next; 理解成p跟着p→next指。即指针移到下一个结构体。

**练习题**

1. 单项选择题
2. 放100个学生的数据，包括学号，姓名，成绩。在如下的定义中，不正确的是（ ）。
3. struct student { int sno; char name[20]; float score } stu[100];
4. struct student stu[100] { int sno; char name[20]; float score } ;
5. struct { int sno; char name[20]; float score } stu[100];
6. struct student { int sno; char name[20]; float score };struct student stu[100];
7. 设有定义语句“struct{ int x; int y; } d[2]={{1,3},{2,7}};”，

则printf(“%d\n”,d[0].y/d[0].x\*d[1].x); 的输出是（ ）。

A. 0 B. 1 C. 3 D. 6

1. 设有定义语句“enum team {my,your=4,his,her=his+10};”，

则printf(“%d,%d,%d,%d\n”,my,your,his,her); 的输出是（ ）。

A. 0，1，2，3 B. 0，4，0，10 C. 0，4，5，15 D. 1，4，5，15

1. 以下对枚举类型名的定义中正确的是（ ）。

A. enum a={one,two,three}; B. enum a {a1,a2,a3};

C. enum a={‘1’,’2’,’3’}; D. enum a {“one”,”two”,”three”};

1. 若有如下定义，则printf(“%d\n”,sizeof(them));的输出是（ ）。

typedef union {long x[2]; int y[4]; char z[8]; } MYTYPE;

MYTYPE them;

A. 32 B. 16 C. 8 D. 24

1. 设有以下说明和定义：

typedef union {long i; int k[5]; char c; } DATE;

struct date { int cat; DATE cow; double dog; } too;

DATE max;

则下列语句的执行结果是（ ）。

printf(“%d”,sizeof(struct date)+sizeof(max));

A. 26 B. 30 C. 18 D. 8

1. 根据下面的定义，能打印出字母M的语句是（ ）。

struct person { char name[9]; int age; };

struct person c[10]={“John”,17,”Paul”,19,”Mary”,18,”Adam”,16};

A. printf(“%c”,c[3].name); B. printf(“%c”,c[3].name[1]);

C. printf(“%c”,c[2].name[1]); D. printf(“%c”,c[2].name[0]);

1. 设有如下定义，则对data中的a成员的正确引用是（ ）。

struct sk {int a; float b; } data, \*p=&data;

A. (\*p).data.a B. (\*p).a C. p->data.a D. p.data.a

1. 设有如下定义，则对字符串li ming的不正确引用是（ ）。

struct person{ char name[20]; cher sex; } a={“li ming”,’m’},\*p=&a;

A. (\*p).name B. p.name C. a.name D. p->name

1. 设有如下定义成的链表，则值为6的表达式是（ ）。

struct st { int n; struct st \*next ; } a[3]={5,&a[1],7,&a[2],9,NULL},\*p=&a;

A. p++->n B. p->n++ C. (\*p).n++ D. ++p->n

二、填充题

1. 以下三种定义结构类型的写法是 的。

struct abc { int a; char c; int b; };

struct abc { int a; int b; char c; };

struct abc { char c; int b,a; };

2．“.”称为 运算符，“->” 称为 运算符。

3．设有定义语句“struct {int a; float b; char c;} abc,\*p\_abc=&abc;”，则对结构体成员a的引用方法可以是abc a和p\_abc a。

4．若有以下说明和定义语句，则变量w在内存中所占的字节数是 。

union aa {float x; float y; char c[6]; } ;

struct st {union aa v; float w[5]; double ave; } w;

5. 若有以下定义和语句，则表达式 ++p->a 的值是 。

struct wc { int a; int \*b; };

int x[]={11,12}; y[]={31,32};

static struct wc z[]={100,x,300,y}, \*p=z;

1. 以下程序输入20个人的姓名和他们的电话号码（七个数字），然后输入姓名，查找该人的电话号码，请填空。

#include <stdio.h>

#include <string.h>

struct ph{ char name[10]; char tel[8]; };

main()

{ s[20];

void readin( struct ph \*p);

void search( struct ph \*p, char \*x);

char c[10];

readin( s );

printf(“请输入被查人的姓名：\n”); gets( );

search( s, c );

}

void readin( struct ph \*p)

{ int i;

for(i=0; i<20; i++,p++)

{ printf(“请输入姓名：”); gets( );

printf(“请输入他的电话号码：”); gets( );

}

}

void search( struct ph \*p, char \*x)

{ int i;

for(i=0; i<20; i++,p++)

if (strcmp( )==0)

{ printf(“%s 的电话号码是 %s\n”,x,p->tel); break; }

if (i==3) printf(“找不到 %s 的电话号码!\n”,x);

}

三、程序分析题

1. 阅读下列程序，写出运行结果。（字符0的ASCII码为16进制的30）

main()

{ union {char c; char i[4];} z;

z.i[0]=0x39; z.i[1]=0x36; printf(“%c\n”,z.c);

}

1. 阅读程序，写出程序的运行结果。

main()

{ struct student

{ char name[10];

float k1;

float k2;

} a[2]={{“zhang”,100,70},{“wang”,70,80}},\*p=a;

printf(“\nname:%s total=%f”, p->name,p->k1+p->k2);

printf(“\nname:%s total=%f”, a[1].name,a[1].k1+a[1].k2);

}

1. 阅读程序，写出程序的运行结果。

main()

{ enum em {em1=3, em2=1, em3};

char \*aa[]={“AA”,”BB”,”CC”,”DD”};

printf(“%s,%s,%s\n”,aa[em1],aa[em2],aa[em3]);

}

4. 阅读程序，写出程序的运行结果。

main()

{ struct std

{ int id;

char \*name;

float sf;

} a, \*p=&a;

int i=a.id=1998;

char \*s=a.name="Windos 98";

float f=a.sf=1800;

printf( "%d is %s sal %f\n",i,s,f);

printf( "%d is %s sal %f\n",p->id,p->name,p->sf);

}

1. 阅读程序，写出程序的运行结果。

void main()

{ struct st

{ int n;

struct st \*next ;

} a[3]={5,&a[1],7,&a[2],9,&a[0]},\*p=a;

int i;

for(i=0; i<5; i++) p=p->next;

printf("p->n=%d\n",p->n);

}

四、程序设计题

1. 用结构体存放下表中的数据，然后输出每人的姓名和实发数（基本工资+浮动工资-支出）。

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 基本工资 | 浮动工资 | 支出 |
| zhao | 240.00 | 400.00 | 75.00 |
| qian | 360.00 | 120.00 | 50.00 |
| sun | 560.00 | 0.00 | 80.00 |

1. 编一个程序，输入10个职工的编号、姓名、基本工资、职务工资，求出其中“基本工资+职务工资”最少的职工姓名并输出。
2. 编一个程序，输入10个学生的学号、姓名、3门课程的成绩，求出总分最高的学生姓名并输出。
3. 编一个程序，输入下列学生成绩表中的数据，并用结构体数组存放。然后统计并输出三门课程的名称和平均分数。

|  |  |  |  |
| --- | --- | --- | --- |
| name | foxbase | basic | c |
| zhao | 97.5 | 89.0 | 78.0 |
| qian | 90.0 | 93.0 | 87.5 |
| sun | 75.0 | 79.5 | 68.5 |
| li | 82.5 | 69.5 | 54.0 |

1. 定义一个结构体变量（包括年、月、日）。编一函数，以结构类型为形参，返回该日在本年中是第几天。

6． 编写一个函数，对结构类型（包括学号，姓名，三门课的成绩）开辟存储空间，此函数返回一个指针（地址），指向该空间。

**【答案】**

1. 单项选择题

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| B | D | D | B | C | B | D | B | B | D |

二、填充题

1. 等价

2. 成员 指向成员

3. . ->

4. 18

1. 101

解答该题，要能画出如下的存储结构示意图：

++p->a即 ++(p->a)，由于p->z数组则p即z，p->a即z[0].a为100，再将此值自增1，后取值，为101

100

300

12

11

32

31

━→

━→

━→

y

x

x[1]

x[0]

y[1]

y[0]

z[0].a

z[0].b

z[1].a

z[1].b

z

p

6. struct ph c p->name p->tel p->name, x

三、程序分析题

1. 运行结果：由于对union 类型变量 z ，z.i[0]和z.c为同一内存位置，所以输出字符9。

2. 显然p指向结构数组第一个元素，故运行结果，第一个printf输出name=zhang total=170.0

而a[1]为结构数组第二个元素，故第二个printf输出name=wang total=150.0

3. 枚举值em3的值为2，因此相当于输出aa[3],aa[1],aa[2]，故运行结果为DD,BB,CC

4. 程序的运行结果：由于p指向a，a和变量i,f,s等同时赋初值，因此二个printf 同

输出：1998 is Windows sal 1800.00

5. 程序的运行结果：p->9

5

a[0]

p↑

next

7

a[1]

n

next

9

a[2]

n

next

n

a

四、程序设计题

1. main()

{ struct emp

{ char name[10];

float jbg; float fdg; float zc;

} e[3]={ {“zhao”,240,400,75},{“qian”,360,120,50},{“sun”,560,0,80}};

int i;

for(i=0;i<3;i++)

printf(“姓名：%5s, 实发数：%7.2f\n”, e[i].name,e[i].jbg+e[i].fdg-e[i].zc );

}

2. main()

{ struct emp

{ int eno;

char en[20];

float jbg; float zwg;

} a[10];

int i, k; float min;

printf(“输入10个职工：编号，姓名，基本工资，职务工资：\n”);

for(i=0; i<10; i++)

scanf(“%d,%s,%f,%f”,&a[i].eno, a[i].en, &a[i].jbg, &a[i].zwg);

min=a[0].jbg+a[0].zwg;

for( k=0,i=1;i<10;i++)

if(min> a[i].jbg+a[i].zwg) { min=a[i].jbg+a[i].zwg; k=i;}

printf(“基本工资+职务工资 最少的职工姓名是：%s\n”,a[k].en);

}

3. main()

{ struct student

{ int sno;

char sn[20];

float score[3];

float sum;

} s[10], \*p=s,\*q;

int i;

float max=0;

printf("输入10个学生：学号，姓名:\n 成绩1 成绩2 成绩3: \n");

for(; p<s+10; p++)

{ scanf(“%d,%s”,&p->sno, p->sn);

p->sum=0;

for(i=0;i<3;i++) { scanf(“%f”, &p->score[i]); p->sum+=p->score[i]; }

}

p=s; q=p; max=p->sum;

for(; p<s+10; p++)

if(max < p->sum) { max=p->sum; q=p; }

printf(“总分最高的学生姓名是：%s 总分 %f\n”,q->sn,q->sum);

}

4. main()

{ struct student

{ char sn[20];

float score[3];

float avg;

} s[4]={{“zhan”,97.5,89.0,78.0,0},{“qian”,90.0,93.0,87.5,0},

{“sun”,75.0,79.5,68.5,0},{“li”,82.5,69.5,54.0,0},}, \*p=s;

int i;

for(; p<s+4; p++)

{ for(i=0;i<3;i++) p->avg+=p->score[i];

p->avg/=3;

}

p=s;

printf(“name foxbase basic c average\n”);

for(; p<s+4; p++)

printf(“%-10s %4.1f %8.1f %8.1f %8.2f\n”,p->sn,

p->score[0], p->score[1], p->score[2], p->avg);

}

5. int days(struct date \*p)

{ int n;

switch(1)

{ case 1: n=0; if(p->m==1) break;

case 2: n=n+31; if(p->m==2) break;

case 3: n=n+28; if(p->m==3) break;

case 4: n=n+31; if(p->m==4) break;

case 5: n=n+30; if(p->m==5) break;

case 6: n=n+31; if(p->m==6) break;

case 7: n=n+30; if(p->m==7) break;

case 8: n=n+31; if(p->m==8) break;

case 9: n=n+31; if(p->m==9) break;

case 10: n=n+30; if(p->m==10) break;

case 11: n=n+31; if(p->m==11) break;

case 12: n=n+30;

}

n=n+p->d;

if(p->m>2)

if((p->y%4==0&&p->y%100!=0)||(p->y%400==0)) n=n+1;

return n;

}

6. struct student

{ int sno;

char name[10];

float score[3];

};

struct student \*nnew()

{ struct student \*p;

p=(struct student \*)malloc(sizeof(struct student));

return p;

}