

一、选择题

1. 有如下结构体说明，以下叙述中错误的是_____。
- ```
struct stu {
 int a; float b;
} stutype;
```
- A) struct 是结构体类型的关键字  
B) struct stu 是用户定义的结构体类型  
C) stutype 是用户定义的结构体类型名  
D) a 和 b 都是结构体成员名
2. 设有结构体说明 struct ex { int x; float y; char z;} example;以下叙述中错误的是\_\_\_\_\_。
- A) struct 是结构体类型的关键字      B) example 是结构体类型名  
C) x,y,z 都是结构体成员名      D) struct ex 是结构体类型
3. 以下\_\_\_\_\_定义不会分配实际的存储空间。
- A) struct {  
 char name[10] ; int age ;  
} student ;
- B) struct STUDENT {  
 char name[10] ;  
 int age ;  
} student ;
- C) struct STUDENT {  
 char name[10] ; int age ;  
} ;  
struct STUDENT student;
- D) struct STUDENT {  
 char name[10] ;  
 int age ;  
} ;
4. 以下对结构体类型变量 td1 的定义中，不正确的是\_\_\_\_\_。
- A) #define AA struct aa  
 AA { int n;  
 float m;  
 } td1;
- B) struct  
 { int n;  
 float m;  
 } td1;
- C) typedef struct aa  
 { int n;  
 float m;  
 } AA;  
 AA td1;
- D) struct  
 { int n;  
 float m;  
 } aa;  
 struct aa td1;

5. 在 Turbo C 2.0 中, 若有定义:

```
struct data { int i ; char ch; double f; } b ;
```

则结构变量 b 占用内存的字节数是\_\_\_\_\_。

- A) 1                      B) 2                      C) 8                      D) 11

6. 当定义一个结构体变量时, 系统分配给它的内存量是\_\_\_\_\_。

- A) 各成员所需内存量的总和                      B) 结构中第一个成员所需内存量  
C) 成员中占内存量最大的容量                      D) 结构中最后一个成员所需内存量

7. C 语言结构体类型变量在程序执行期间\_\_\_\_\_。

- A) 所有成员驻留在内存中                      B) 只有一个成员驻留在内存中  
C) 部分成员驻留在内存中                      D) 没有成员驻留在内存中

8. 已知学生记录描述为:

```
struct student {
 int no ; char name[20]; char sex;
 struct { int year; int month ; int day ; } birth;
} s;
```

设结构变量 s 中的“birth”应是“1985 年 10 月 1 日”, 则下面正确的赋值是\_\_\_\_\_。

- A) year=1985;                      B) birth.year=1985;  
month=10;                      birth.month=10;  
day=1;                      birth.day=1;  
C) s.year=1985;                      D) s.birth.year=1985;  
s.month=10;                      s.birth.month=10;  
s.day=1;                      s.birth.day=1;

9. 若有定义 struct num {int a; int b;} d[3]={ {1,4}, {2,5}, {6,7} }; 执行语句 printf ("%d", d[2].a\*d[2].b/d[1].b); 后的输出结果是\_\_\_\_\_。

- A) 2                      B) 2.5                      C) 8                      D) 8.4

10. 下面程序执行后的输出结果是\_\_\_\_\_。

```
1 #include <stdio.h>
2 void main()
3 {
4 struct complx { int x; int y ; } cnum[2]={1,3,2,7} ;
5 printf ("%d", cnum[0].y/cnum[0].x*cnum[1].x);
6 }
```

- A) 0                      B) 1                      C) 2                      D) 6

11. 根据下述定义, 能输出字母 M 的语句是\_\_\_\_\_。

```
struct person {char name[9]; int age;};
struct person class[10]={ "Johu",17, "Paul",19, "Mary",18,

 "Adam",16};
```

- A) printf ("%c", class[3].name);  
B) printf ("%c", class[2].name[0]);  
C) printf ("%c", class[3].name[1]);  
D) printf ("%c", class[2].name[1]);

12. 下面程序执行后的输出结果是\_\_\_\_\_。

```
1 #include <stdio.h>
2 struct st { int x; int *y; } *p;
3 void main()
4 { int dt[4]={10,20,30,40};
5 struct st aa[4]={ 50,&dt[0],60,&dt[0],60,&dt[0],60,
6 &dt[0]};
7 p=aa;
8 printf("%d",++(p->x));
9 }
```

A) 10                      B) 11                      C) 51                      D) 60

13. 下面程序执行后的输出结果是\_\_\_\_\_。

```
1 #include <stdio.h>
2 struct STU {
3 char name[10]; int num; float TotalScore;
4 };
5 void f(struct STU *p)
6 { struct STU s[2]={{ "SunDan",20044,550},{ "Penghua",
7 20045,537}}, *q=s;
8 ++p ; ++q; *p=*q;
9 }
10 void main()
11 { struct STU s[3]={{ "YangSan",20041,703},{ "LiSiGuo",
12 20042,580}};
13 f(s);
14 printf("%s %d %3.0f", s[1].name, s[1].num,
15 s[1].TotalScore);
16 }
```

A) SunDan 20044 550                      B) Penghua 20045 537  
C) LiSiGuo 20042 580                      D) SunDan 20041 703

14. 以下对结构体变量成员不正确的引用是\_\_\_\_\_。

```
struct pupil {
 char name[20]; int age; int sex;
```

```
 } pup[5], *p=pup;
```

A) scanf("%s",pup[0].name);              B) scanf("%d",&pup[0].age);  
C) scanf("%d",&(p->sex));                  D) scanf("%d",p->age);

15. 有以下程序段:

```
int a=1,b=2,c=3;
struct dent {
 int n ; int *m;
} s[3] = {{101,&a}, {102,&b},{103,&c}};
struct dent *p=s;
```

则以下表达式中值为2的是\_\_\_\_\_。

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

16. 以下引用形式不正确的是\_\_\_\_\_。

```
struct s {
 int i1; struct s *i2,*i0;
};
static struct s a[3]={2,&a[1],0,4,&a[2],&a[0],6,0,&a[1]},
*ptr=a;
```

- A) ptr->i1++      B) \*ptr->i2      C) ++ptr->i0      D) \*ptr->i1

17. 设有如下定义:

```
struct sk {
 int a;
 float b;
} data;
int *p;
```

若要使p指向data中的a, 正确的赋值语句是\_\_\_\_\_。

- A) p=&a;      B) p=data.a;      C) p=&data.a;      D) \*p=data.a;

18. 若有定义 struct{int a; char b;} Q,\*p=&Q;则错误的表达式是\_\_\_\_\_。

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

19. 设有定义 struct ru{long x;float y;} time,\*timep=&time;则对time中成员x的正确引用是\_\_\_\_\_。

- A) ru.time.x      B) timep.x      C) (\*timep).x      D) time->x

20. 下面程序执行后的输出结果是\_\_\_\_\_。

```
1 #include <stdio.h>
2 struct s {
3 int x,y;
4 } data[2]={10,100,20,200};
```

```
5 void main()
6 { struct s *p=data;
7 printf("%d",++(p->x));
8 }
```

- A) 10      B) 11      C) 20      D) 21

21. 有以下说明和定义:

```
struct student { int age; char num[8];} ;
struct student stu[3]={ {20, "200401"}, {21, "200402"}, {19,
 "200403"} };
struct student *p=stu;
```

以下选项中引用结构体变量成员的表达式错误的是\_\_\_\_\_。

- A) (p++)->num      B) p->num      C) (\*p).num      D) stu[3].age

22. 有以下程序段：

```
struct st {
 int x; int *y;
} *pt;
int a[]={1,2},b[]={3,4};
struct st c[2]={10,a,20,b};
pt=c;
```

以下选项中表达式的值为 11 的是\_\_\_\_\_。

- A) \*pt->y      B) pt->x      C) ++pt->x      D) (pt++)->x

23. 有以下结构体说明和变量的定义，且指针 p 指向变量 a，指针 q 指向变量 b，则不能把节点 b 连接到节点 a 之后的语句是\_\_\_\_\_。

```
struct node {
 char data;
 struct node *next;
} a,b,*p=&a,*q=&b;
```

- A) a.next=q;    B) p.next=&b;    C) p->next=&b;    D) (\*p).next=q;

24. 若要利用下面的程序段使指针变量 p 指向一个存储整型变量的存储单元，则在【\_\_\_\_\_】中应有的内容是\_\_\_\_\_。

```
int *p;
p = 【_____】malloc(sizeof(int));
```

- A) int      B) int \*      C) (\*int)      D) (int\*)

25. 下面程序执行后的输出结果是\_\_\_\_\_。

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 struct NODE {
4 int num; struct NODE *next;
5 };
6 void main()
7 { struct NODE *p,*q,*r;
8 p=(struct NODE*)malloc(sizeof(struct NODE));
9 q=(struct NODE*)malloc(sizeof(struct NODE));
10 r=(struct NODE*)malloc(sizeof(struct NODE));
11 p->num=10; q->num=20; r->num=30;
12 p->next=q;q->next=r;
13 printf("%d",p->num+q->next->num);
14 }
```

- A) 10      B) 20      C) 30      D) 40

编程题：

13 个人转成一圈，从第一个人开始报数，报到 3 的退出，问最后退出的人原来的序号是多少？