

教务处试卷编号:

备注: 试卷背面为演草区 (不准用自带草纸)

装

订

线

## 一 课程目标 2 (共 20 分)

## (一) 选择题 (每题 1 分, 共 15 分)

- 以下程序的输出结果是 D  

```
main(){ int a=11,b=21,c=31,d=51;
d=(a<b<c);
printf("%d\n",d);}
```

A. 51      B. 31      C. 11      D. 1
  - 下面选项中错误的定义语句是 C  

A. `int x[3]={0},{1},{1,2,3};`  
 B. `int x[4][3]={1,2,3},{1,2,3},{1,2,3};`  
 C. `int x[4][3]={1,2,3},{1,2,3},{1,2,3},{1,2,3};`  
 D. `int x[4][3]={1,2,3,4};`
  - 以下循环体的语句有错误的是 C  

A. `while (x=y)5;`      B. `while(0);`      C. `do x++ while(x==10);`      D. `do 2; while (a==b);`
  - 在 C 语言中, 当函数调用时 B  

A. 可以由用户指定实参和形参是否共用存储单元  
 B. 实参和形参共用存储单元  
 C. 由系统自动确定实参和形参是否共用存储单元  
 D. 实参和形参各占一个独立的存储单元
  - 下列程序的输出结果是 C  

```
main() { int a=4,b=35; printf("%d\n",b=b/a); }
```

A. 4      B. 6      C. 8      D. 35
  - 以下不合法的用户标识符是 D  

A. `_8acm`      B. `Single`      C. `M15_DOG`      D. `_101_`
  - 设 `a=30, b=20`, 执行 `printf("%d, %d", (a, b), (b, a));` 的输出是 B  

A. 30, 20      B. 20, 30      C. 30, 30      D. 20, 20
  - 在 C 程序中, 可把整型数以二进制形式存放到文件中的函数是 D  

A. `fprintf` 函数      B. `fread` 函数      C. `fwrite` 函数      D. `fputc` 函数
  - 指针数组的正确定义形式是 C  

A. `int *b[5]`      B. `int (*c)[6]`      C. `int *(q[7])`      D. `int *m[]`
  - 用 `strcmp("DMU", "Dalian")` 的结果是: B  

A. `>0`      B. `<0`      C. `=0`      D. `NULL`
  - 以下叙述正确的是 D  

A. 函数的定义可以嵌套, 但函数的调用不可以嵌套。  
 B. 函数的定义不可以嵌套, 但函数的调用可以嵌套。  
 C. 函数的定义和函数的调用都不可以嵌套。  
 D. 函数的定义和函数的调用都可以嵌套。
  - 下列不能正确进行字符串赋值操作的语句是 C  

A. `char ok[20]; gets(ok);`      B. `char *ok; ok = "ha!";`  
 C. `char ok[10]; ok = "command";`      D. `char ok[20]; strcpy(ok, "how are you!");`
  - 若有函数 `max(a, b)`, 并且已使函数指针变量 `p` 指向函数 `max`, 当调用该函数时, 正确的调用方法是 D  

A. `(*p)max(a,b);`      B. `*pmax(a,b);`      C. `(*p)(a,b);`      D. `*p(a,b);`
  - 表达式 `"12|012"` 的运算结果用十进制数表示是 D  

A. 12      B. 14      C. 16      D. 10
  - 若有说明语句 `"int n[10], *q = n;"`, 对数组元素的正确引用是 D  

A. `q[n]`      B. `n[q]`      C. `*(q+5)`      D. `q+5`
- (二) 判断题 (若正确, 在括号内填 "T"; 否则填 "F"。每题 0.5 分, 共 5 分)
- 共用体是一种自定义类型, 它使用覆盖技术将几个不同类型的变量置于同一段内存, 且其起始地址是相同的。✓
  - `switch` 语句中多个 `case` 不可以共用一组语句。✗
  - 若 `a=15`, 则 `a<<2` 的值是 60。✗
  - 自增运算符(`++`)或自减运算符(`--`)只能用于变量, 不能用于常量或表达式。✓
  - 在 C 程序中, `main()` 的位置必须放在它所调用的函数之后。✗
  - 已知语句 `"int n=3;"`, 则语句 `"int a[n];"` 表示定义了一个大小为 3 的一维整型数组 `a`。✓
  - `while` 循环语句的循环体至少执行一次。✗
  - 系统自动定义的文件指针 `stdin` 用于指向终端输入 (显示器)。✓
  - 当内层模块内定义了和外层模块同名的类型和量值时, 内层的类型和量值屏蔽了外层模块的同名类型和量值, 导致只能引用内层中定义的部分。✗
  - 设有数组定义: `char array[10]="Chinese";` 则数组 `array` 所占的空间为 8 个字节。✗

教务处试卷编号:

备注: 试卷背面为演草区 (不准用自带草纸)

装

订

线

## 二 课程目标 3 (共 20 分)

(一) 学生信息管理。每个学生数据包括: 学号 (num) (定义为整形), 姓名 (name) (字符数组, 长度 10)、三门课程成绩 (score) (定义为整形数组) 一个结构体 (student), 并定义结构体数组 stu (学生数量采用宏定义形式, N 定义为 10 个)。编程要求: 从键盘输入所有学生的相关信息, 找到总分最低的学生信息。(10 分)

(二) 验证哥德巴赫猜想。哥德巴赫猜想之一是任何一个大于 5 的偶数都可以表示为两个素数之和。请编程验证这一论断, 并画出主程序流程图。(10 分)

要求: 判断一个数是否为素数的功能由函数 pn() 实现, 是素数则返回 1, 不是返回 0。请完善主程序 main() 函数, 在主函数中实现 5000 至 10000 范围以内偶数的验证结果输出。在主程序中需要判定一个数是否为素数时, 必须调用 pn() 函数。

#include&lt;stdio.h&gt;

int pn ( int x){ int d=2; while(x%d)d++; if(x==d) return 1; else return 0;}

void main ( ) { }

## 三 课程目标 4 (共 40 分)

(一) 阅读程序, 写出正确运行的结果 (包含格式) (每题 2 分)

1. 假设划线语句分别输出

2686650 和 4

void main( )

{ int a[3][5];

printf("n%ou", a);

printf("n%ou", sizeof(a[0][0]));

printf("ou\n", &amp;a[1][1]);

printf("ou\n", \*(a+1)+1);

}

4. #define M(x,y,z) x\*y+z

void main()

{ int a=3,b=6,c=9;

printf("ou\n", M(a+b,b+c,a));

}

2. void main()

{ int m=136, \*p1, \*p2;

p2=p1= &amp;m;

printf("m= %d\n", (\*p1)++);

printf("p= %d\n", \*p2);

}

5. void main()

{ char \*str="She is so cute";

int i=0;

while(\*str++!='\0')

i++;

printf("the length of the string is %d ",i);}

3. 程序执行时输入 57408

void main()

{ long int num;

int n;

printf("please input a integer number ");

scanf("%ld",&amp;num);

while(num&gt;0)

{ n=num%10;

printf("%3d",n);

num=num/10; }

}

6.

int Min\_of\_EveryRow(int \*p)

{ int i,j,min\_of\_this\_row;

min\_of\_this\_row=p[0];

for(j=1;j&lt;3;j++)

if(min\_of\_this\_row&gt;p[j])

min\_of\_this\_row=p[j];

return(min\_of\_this\_row);

}

void main()

{ int a[3][3]={8,51,13,38,14,16,20,9,26};

int i,max,min;

max=Min\_of\_EveryRow(a[0]);

for(i=1;i&lt;3;i++)

{ min=Min\_of\_EveryRow(a[i]);

if(max&lt;min) max=min;

}

printf("n max value is %d\n",max);

}

9. int aaa (int a) {

static int m=3;

int i=3;

i+=m+1; m=i+a;

return (m);

}

void main ( ) {

int k=3,c1,c2;

c1=aaa(k); c2=aaa(k);

printf("c1= %d \nc2= %d\n",c1,c2);

}

7. 假定在当前目录下有 2 个文本文件,

其名称和内容如下:

文件名 mm.txt nn.txt

内容 looking# forward#

void fc(FILE \*fp1) {char c;

while ((c = fgetc(fp1)) != '#')

putchar(c);

}

void main(){

FILE \*fp;

if((fp = fopen("mm.txt", "r")) == NULL) {

printf("Can not open file!\n"); exit(1);

}

else { fc(fp); fclose(fp); }

printf("n");

if((fp = fopen("nn.txt", "r")) == NULL){

printf("Can not open file!\n"); exit(1);

}

else { fc(fp); fclose(fp); }

}

10. main()

{ int k=5,n=0;

while(k&gt;0)

{ switch(k)

{ default : break;

case 1: n+=k; k--;

case 2:

case 3: n+=k; k--;

k--;

printf("%d\n",n);

}

8.

void main()

{ int i,j,k,x;

char \*name[]={"Shiying","Huangyan","Wanfang",

"Baixue","Tangjin"};

char \*b;

for(i=0;i&lt;4;i++)

{ k=i;

for(j=i+1;j&lt;5;j++)

if(strcmp(name[k],name[j])&gt;0)

k=j;

b=name[i];

name[i]=name[k];

name[k]=b;

}

for(i=0;i&lt;2;i++)

puts(name[i]);

}



## ) 完善程序 (在每个空上填上合适的语句) (每题 2 分, 共 20 分)

1. 输入一个字符串, 将其中的小写字母转换为对应的大写字母输出, 否则直接输出。大写小写字母差值为 32。

```
void main()
{ char s[20];
  int k,i;
  (1)
  k=strlen(s);
  for(i=0;i<k;i++)
    (2)
    (3)
  puts(s);
}
```

3. 向一个有序数组中插入一个数, 使得数组依然有序

```
void main()
{ int a[10]={4,9,13,16,30,49,58,60,100};
  int j, k, number;
  scanf("%d", (6));
  for(k=0;k<9;k++)
    if(a[k]>number)break;
  for(j=9;j>k;j--)
    (7); /*从后向前移动数组元素*/
  (8); /*将数据放入数组相应位置*/
  for(j=0;j<10;j++)
    printf("%5d",a[j]);
}
```

2. 程序的输出结果为交换 a 和 b 的值。

```
void swap(int *, int *);
void main()
{ int a=66, b=22;
  swap(&a, &b);
  printf("a=%d, b=%d", a, b);
}
void swap((4))
{ int temp;
  (5)
}
```

4. 将字符串 s2 连接到 s1 后面。

```
void main()
{char s1[20],s2[20]; int i=0, j=0;
  gets(s1); gets(s2);
  while(s1[i]!='\0') /*找到 s1 的尾 */
    i++;
  while((9) /*逐个拷贝相应字符 */
    (10)
  s1[i]='\0';
  puts(s1);
}
```

## 四 课程目标 5 (共 20 分)

## 编程 (每题 10 分, 共 20 分)

1. 一个浮点型 5×5 的矩阵 F, 编程求出所有对角线上元素的和 sum (结果为浮点数)。

要求: 数组名为 F, 循环变量定义为 i 和 j。实现数组声明、通过键盘数据输入和结果输出。

2. 求 10 个学生成绩的平均分, 最高分和最低分。要求: 在 average 函数中不能出现输出语句。

```
float average(float *array, int n, float *pmax, float *pmin) {
```

```
}
```

```
void main() {
```

```
float ave, score[10]={10, 50, 60, 70, 80, 20, 30, 40, 90, 100};
```

```
int i; float max, min;
```

```
ave=average(score, 10, &max, &min);
```

```
printf(".....", max, min, ave);
```

```
}
```