******华中科技大学计算机科学与技术学院2021~2022第一学期**

解答内容不得超过装订线

**“C语言程序设计”考试试卷**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考试方式** | **闭卷** | **考试日期** | 单击或点击此处输入日期。 | **考试时长** | **150 分钟** |
| **专业班级** |  | **学 号** |  | **姓 名** |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **题号** | **一** | **二** | **三** | **四** | **五** | **总分** | **核对人** |
| **分值** | 10 | 14 | 24 | 24 | 28 | 100 |  |
| **得分** |  |  |  |  |  |  |  |

**注意：试卷后面附有运算符的优先级和结合性表，该页可撕下作草稿。**

|  |  |
| --- | --- |
| **分 数** |  |
| **评卷人** |  |

**一、计算表达式的值**（**每个小题相互独立，不会互相影响**。本大题共10小题，每小题1分，共10分。）

1. 请根据下面的声明，计算（1）~（5）题表达式的值并填入各题后面的下划线中。

short a = 100, b = 0101, c = 0x10;

char s[] = “HELLOWORLD”;

（1）表达式 strlen(s) 的值为：

（2）表达式 a = b = c 执行完成后，a的值为：

（3）表达式 a < b ? a-- : b > c ? ++b : c++ 的值为：

（4）表达式 a += b |= c 执行完成后，a的值为：

（5）c ^ b << 2 的值为：

2. 请根据下面的声明，计算（6）~（10）题表达式的值并填入各题后面的下划线中。

struct t {

float x;

struct t \*n;

} a[3] = {{3.1, a+1}, {1.2, a+2}, {3.2, 0}}, \*p = a;

int b[3] = {0, 1, 2}, \*q = b;

1. ++p->x 的值为：

（7）(\*++p).x 的值为：

（8）(p = p->n)->n->x 的值为：

（9）q[(int)(\*++p).x] + a[1].x 的值为：

（10）(p++)->x + a[\*q++].x 的值为：

|  |  |
| --- | --- |
| **分 数** |  |
| **评卷人** |  |

1. **改错题** （划线指出下面各题程序中的错误并改正，使之实现题目的功能。本大题共7小题，每小题2分，共14分。提示：每小题有2个错误。）

1. 利用pi/2=2/1 \*2/3 \*4/3 \*4/5 \*6/5\*6/7…的前100项之积，计算pi的值。

#include <stdio.h>

int main()

{

double term, result;

int n;

for (n=2; n<=100; n+=2) {

term = (n \* n) / ( n - 1.0) \* ( n + 1);

result = result \* term;

}

printf("result=%f\n", 2 \* result);

return 0;

}

2. 计算数列a[n] = 2n / (n + 1)的第100项与常数2的差：2 - a[100]。程序的输出为：0.019802

#include<stdio.h>

int main()

{

int k = 100;

printf("%f", 2 - func(k));

return 0;

}

double func(int k) {

double x;

x = (double)(2 \* k / (k + 1));

return x;

}

3. 打印字母表倒三角形。如输入n为4时，打印出：

ABCD ABC AB A

#include<stdio.h>

void func(int n){

char a[32];

int i;

for(i = 0; i < n; i++) a[i] = i + 'A';

printf("%s\n", a);

}

int main()

{

int n, k;

scanf("%d", n);

for(k = n; k > 0; k--) func(k);

return 0;

}

4. 利用指针p倒序打印出数组a的所有元素。

#include<stdio.h>

int main()

{

int a[ ] = {1,2,3,4,5};

int \*p = \*(a + 4);

while(p != a) {

printf("%d\n", p[0]);

p = p - 1;

}

return 0;

}

5. 将数组a中下标为奇数的元素除以2，下标为偶数的元素乘以2并打印出来。

#include<stdio.h>

double mult(double x){

return x \* 2;

}

double div(double x){

return x / 2;

}

int main(void)

{

double (\*p)(double)[2] = {mult, div};

double a[5] = {-1, 0, 1, 2, 3};

int i;

for(i = 0; i < 5; i++){

a[i] = (p + i % 2)(a[i]);

printf("%f ", a[i]);

}

return 0;

}

6. 利用指针函数func实现将数组y的元素设置为数组x的每个元素乘以2，并打印出数组y。

#include<stdio.h>

int \*func(int \*a)

{

int b[3];

int i;

for(i = 0; i < 3; i++) b[i] = a[i] \* 2;

return b;

}

int main()

{

int i, x[3] = {1,2,3};

int y[3];

y = func(x);

for(i = 0; i < 3; i++) printf("%d ", y[i]);

return 0;

}

7. 一个长整型数占4个字节，其中每个字节又分为高4位和低4位。试从长整型数的低字节开始，依次取出每个字节的高4位和低4位并以ASCII字符的形式进行显示。

#include <stdio.h>

int main(void)

{ long x=0xABCD1234, k;

long \*p=&x;

char up\_half,low\_half;

for(k=0;k<4;k++,p++) {

low\_half=(\*p)&0x0f;

if(low\_half<10) low\_half+='0';

else low\_half=(low\_half-10)+'A';

up\_half=(\*p)&0x0f;

if(up\_half<10) up\_half+='0';

else up\_half=(up\_half-10)+'A';

printf("%c %c\n",up\_half,low\_half);

}

return 0;

}

|  |  |
| --- | --- |
| **分 数** |  |
| **评卷人** |  |

**三、简答题**（本大题共6小题，每小题4分，共24分。）

1. 下列表示中哪些是正确的C语言字符常量表达式？(备注：第一个是三个单引号，第二个是单引号双引号单引号。）

''' '"' '\18' '\0xa' '\45' '\0' '\a'

2. 用5个位（第15~11位）表示日，用4个位（第10~7位）表示月，用7个位（第6~0位）表示年（0~99），可以将一个日期的日、月、年压缩存储在16位的short类型整数中。假设有如下声明： short date, month; 变量date中存放的是按上述方法压缩后的日、月、年，变量month中存放的是一个合理的月份值。请写一个C表达式，更改date的值，将date中的月替换为month。

3. 设变量声明为：float a; double b; char c; int x; 。将下列表达式中隐含的类型转换用强制类型转换运算符显示地表示出来。

x = a / b + c

4. 设变量声明为：int a[4][5], (\*p1)[5] = a, \*p2 = a[0]; 。请写两个表达式，分别用p1和p2来表示a[i][j]（0≤i≤3，0≤j≤4），要求不允许使用下标运算且表达式尽可能最简。

5. C源程序sort.c经编译链接后得到执行程序sort.exe，按以下格式在命令行上执行：

sort <filename1> <filename2> [-d]

格式中，< >中是必需的参数项，[ ]中是可选的参数项。该命令将文件filename1中用空格分隔的整型数据排序后，输出到文件filename2中，排序后的整型数据仍按空格分隔，排序方式由是否带参数 –d 来确定，如果命令行中带有 -d参数，那么按降序排序，否则按升序排序。请完成下列任务：

1. 写出源程序sort.c中main函数定义的头部;
2. 打开文件filename1，如果打开成功，则读取文件中的整数（假设数据个数不超过100）到数组a中，写出相关语句;
3. 检查命令行中是否有 -d参数，如果有，则将flag设置为0，写出相关语句。
4. 以下是有关类型声明的小题，请按要求答题。

(1) 设有声明：const int X = 10; ，请定义一个指针常量p，并将其初始化为指向X。

(2) 指针数组pa长度为5，元素类型为指向带一个字符指针形参的指针函数的指针，该指针函数返回值类型为整型指针。请写出指针数组pa的声明语句。

|  |  |
| --- | --- |
| **分 数** |  |
| **评卷人** |  |

**四、程序分析题**（本大题共6小题，每小题4分，共24分。）

1. 假设输入如下（↙表示换行），写出下面程序的运行结果。

95 67 25 54 77 34 45 65 32 51↙

#include<stdio.h>

#define N 10

int main(void)

{

int num, m1, m2, i=0;

for(;i<N;i++){

scanf("%d", &num);

if(i == 0){

m1 = num;

m2 = num;

continue;

}

if(num < m1) m1 = num;

if(num > m2) m2 = num;

}

printf("%d %d\n", m1, m2);

return 0;

}

2. 写出下面程序的运行结果。

#include <stdio.h>

int main()

{

int n, start, num,i,j;

for(n=4,i=0;i<=n;i++) {

start = 1;

printf("%d ", start);

for(j=1;j<=i;j++) {

num = start\*(i-j+1)/j;

printf("%d ", num);

start = num;

}

printf("\n");

}

return 0;

}

3. 写出程序的运行结果。

#include <stdio.h>

void func(int nums[ ], int cnt)

{

int i, tmp;

for(i = 0; i < cnt / 2; ++i) {

tmp = nums[i];

nums[i] = nums[cnt - 1 - i];

nums[cnt - 1 - i] = tmp;

}

}

int main()

{

int m=4, n=9, j, line[20]={1,4,7,2,5,8,3,6,9};

func(line, n - m % n);

for(j = 0; j < n; ++j) printf("%d ",line[j]);

func(&line[n - m % n], m % n);

func(line, n);

printf("\n");

for(j = 0; j < n; ++j) printf("%d ",line[j]);

return 0;

}

4. 假设输入如下（↙表示换行），请写出下面程序的运行结果。

awCve↙

Taiwan is an inalienable part of Chinese territory↙

#tay↙

#include <stdio.h>

int main()

{

char s1[7], s2[80],c;

int i, j, n;

while(1) {

i=0;

while((c=getchar())!='\n') s1[i++] = c;

s1[i] = '\0';

if(s1[0] == '#') break;

i = 0;

while((c=getchar())!='\n') s2[i++] = c;

s2[i] = '\0';

for(i=0;s1[i]!='\0';i++) {

n = 0;

for(j=0;s2[j]!='\0';j++) {

if(s1[i] == s2[j]) n++;

}

printf("%c %d\n", s1[i], n);

}

}

printf("%s\n", s1);

return 0;

}

5.写出下面程序的运行结果。

#include <stdio.h>

void func1(char \*s,char \*t)

{

char \*ps,\*pt;

for (pt=t,ps=s; \*ps!='\0'; ps++,pt++) \*pt= \*ps;

do {

--ps；

\*pt++ = \*ps;

} while(ps>s);

\*pt = '\0';

}

int main()

{

char s[ ] = "Hust",t[100];

func1(s,t);

printf("%s",t);

return 0;

}

6. 写出下面程序的运行结果。

#include <stdio.h>

void fun2(char \*a[ ])

{

char \*\*p1,\*\*p2;

char \*t;

p2=p1=a;

p1++;

while(\*p1 != NULL) {

t=\*p1, \*p1=\*p2,\*p2=t;

p2++,p1++;

}

}

int main( )

{

char \*s[]={"one","four","two","five","three",NULL};

int i;

for(i=0; s[i]!=NULL; i++) printf("%c ", \*(s[i])-'a'+'A');

printf("\n");

fun2(s);

for(i=0; s[i]!=NULL; i++) printf("%s\n", s[i]);

return 0;

}

|  |  |
| --- | --- |
| **分 数** |  |
| **评卷人** |  |

**五、程序完善题**（本大题有4小题，给出的都是部分程序，在下划线处填入合适代码来完善程序,使之能够正确实现题目的功能。本大题共14空，每空2分，共28分。）

1. 根据if语句所给条件，填写 switch 语句，使它完成相同的功能。(假设score的类型为int，取值在0~100之间。)

|  |  |
| --- | --- |
| if(score<50) k=1;  else if(score<60) k=2;  else if(score<70) k=3;  else if(score<80) k=4;  else if(score<90) k=5;  else if(score<=100) k=6; | switch（ ① ）{  ② : k=1; break;  case 5: k=2; break;  case 6: k=3; break;  case 7: k=4; break;  case 8: k=5; break;  ③ : k=6; break;  } |

1. ② ③

2. 函数delete的功能是删除字符串t中的所有数字字符。

void delete (char \*t)

{

int k=0, i;

for(i=0; t[i]; i++)

if( ① ) t[k++]=t[i];

t[k]= ② ;

}

① ②

3. 函数find的功能：在有序二维数组x的前n行中查找值为key的数组元素，若找到则将该数组元素在二维数组中的行、列下标值分别保存到row和col指向的变量中且函数返回1，若未找到则函数返回0。已知二维数组x的每一行中元素的值均从小到大顺序存储，第i行中所有元素的值均小于第i+1行中所有元素的值（i=0,1,2,3,...,n-1)。

算法：用折半查找法定位值为key的数组元素所在行，用线性查找法在已经定位的行中查找值为key的数组元素。

#include<stdio.h>

int find(int x[ ][5],int n,int key,int \*row,int \*col);

int main()

{

int a[5][5]={{2,5,6,8,11},{14,16,17,19,21},{24,25,26,27,29},{33,35,36,38,40},{44,46,48,50,52}};

int m, n, key;

puts("input key:");

scanf("%d",&key);

if( find( ① ) ) printf("%d is stored at a[%d][%d]",key,m,n);

else printf("\n %d not found!",key);

return 0;

}

int find(int x[ ][5],int n,int key,int \*row, int \*col)

{

int i,low=0,high=n-1,mid;

while(low<=high) {

mid= ② ;

if(key>=x[mid][0]&& key<=x[mid][4]) break;

if(key<x[mid][0]) ③ ;

else if(key>x[mid][4]) low=mid+1;

}

for(i=0;i<5;i++)

if( ④ ) {

\*row=mid;

\*col=i;

return 1;

}

return 0;

}

① ② ③ ④

4. 下面的程序先调用函数sort对指针数组x所指向的字符串用选择法进行升序排序，然后调用函数creatList建立一个链表，链表各结点的name成员数据来源于指针数组a所指向的字符串，最后调用函数counter统计链表中name成员相同的结点个数，并保存在count成员中，保留一个name成员相同的结点，删除其余结点。程序的输出应为：

book 2

chair 3

eraser 1

pen 1

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

typedef struct p {

char name[20];

int count;

struct p \*next;

} PS;

PS \*creatList(char\*a[],int n);

void counter(PS \*h);

void sort(char \*x,int n);

int main( )

{

PS \*head,\*p;

char \*a[7]={"book", "pen", "chair", "chair","chair","eraser", "book"};

sort(a,7);

head=creatList(a,7);

counter (head);

for(p=head;p!=NULL; p=p->next)

printf( "%s\t%d\n",p->name,p->count);

return 0;

}

void sort(char \*x[],int n) /\*对指针数组x所指向的n个字符串按字典序排序 \*/

{

int j,min;

char \*t;

if(n>1){

min=0;

for(j=1;j<n;j++)

if(x[j]<x[min]) min=j;

if(min) { t=x[0]; x[0]=x[min]; x[min]=t; }

① ;

}

}

/\*将指针数组a所指向的n个字符串建立一个链表，返回链表首结点的地址 \*/

PS \*creatList(char \*a[],int n)

{

int i,c=1;

char name[20];

PS \*head=NULL,\*p,\*tp;

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

{

tp=(PS \*)malloc( ② );

if( ③ ) {

puts("fail!");

exit(-1);

}

strcpy(tp->name,a[i]);

tp->count=1;

tp->next=NULL;

if(head==NULL) head=p=tp;

else {

④ =tp;

p=tp;

}

}

return head;

}

/\* 对于h指向的链表中连续出现的多个name成员相同的那些结点，统计并保存这些相同结点的数量，保留一个结点，删除其余结点 \*/

void counter(PS \*h)

{

while(h->next!=NULL)

if(strcmp(h->name,h->next->name)==0) {

h->next= ⑤ ;

h->count++;

}

else h=h->next;

}

① ②

③ ④ ⑤

**附表1 运算符的优先级和结合性**

|  |  |
| --- | --- |
| 运 算 符 | 结合性 |
| （） [ ] -> .  ! ~ ++ -- + - \* & (类型) sizeof  \* / %  + -  << >>  < <= > >=  == !=  &  ^  |  &&  ||  ?:  = += -= \*= /= %= &= ^= |= <<= >>=  , | 左结合  右结合  左结合  左结合  左结合  左结合  左结合  左结合  左结合  左结合  左结合  左结合  右结合  右结合  左结合 |

注：同一行上各运算符具有相同的优先级，从上往下优先级递降。

**附表2 相关库函数声明**

|  |  |
| --- | --- |
| 函数原型 | 功 能 |
| int strlen(const char \*s); | 求字符串s长度（不计‘\0’） |
| void malloc(unsigned size); | 分配size字节的存储区，返回指向该区的指针，分配不成功，返回NULL |
| void free(void \*p); | 释放p所指的内存 |
| FILE \*fopen(const char \* fname,  const char \*mode); | 以mode方式打开文件fname,成功打开，返回文件指针，否则返回NULL |
| int fclose(FILE \*fp); | 关闭文件，使fp不再与具体文件相关联，成功，返回0，否则返回非0 |
| int strcmp(const char \*s1，const char \*s2); | 按字典序比较字符串s1和s2,s1<s2返回负数，s1>s2返回正数，两串一样返回0 |
| char \*strcpy(char \*dest, const char \*src); | 把 src 所指向的字符串复制到 dest |
| size\_t fwrite(const void \*ptr, size\_t size, size\_t n, FILE \*fp); | 将内存中从ptr地址开始的数据往fp所指向的文件里写，写入到文件里的字节数为size\*n |
| exit(int status); | 关闭所有打开的文件并终止程序 |