复旦大学计算机科学技术学院

《程序设计》期中考试试卷

共 5页

课程代码：COMP120006 考试形式：□开卷 □√闭卷 2022 年 11月

（本试卷答卷时间为120分钟，答案必须写在答题纸上，做在试卷上无效）

专业 学号 姓名

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 题号 | 一 | 二 | 三 | 四 | 五 | 六 | 总分 |
| 分值 |  |  |  |  |  |  |  |

1. 单选题（每题2.5分，共20分）

1、请选出可以作为c语言用户标识符的一组标识符（ ）

A．void、define、WORD B．A3\_B3、\_123、abc

C．FOR、-abc、Case D．2a、Do、Sizeof

2、设有说明：char w; int x; float y; double z;

则表达式w\*x+z-y值的数据类型为（ ）

A.float B. char C. int D. double

3、 若定义a[][2]={1,2,3,4,5,6,7}; 则a数组中行的大小是（　）。

A．2 　　　B．3 　　C．4 　　　　D．无确定值

4、 如果a=3,b=2,c=1,则下列表达式的值为0的是（ ）

A．a>b B．(a>b)==c C．b+c<a D．c=a>b

5、执行语句char str[]=”ab\tde\\fg\n”;printf(“%d”,strlen(str));后输出（ ）

A、12 B、9 C、14 D、10

6、有如下程序，输入数据：12345M678＜cR＞后（<CR>表示回车），x的值是( ) #include<stdio.h>

int main(){

int x;

float y;

scanf("%6d%f",&x,&y);

return 0;

}

A、12345 B、 12345M C、 123456 D、 345

7、有如下程序

#include<stdio.h>

int main()

{

int x=1,y=0,a=1,b=1;

switch(x)

{

case 1: switch(y)

{

case 0:b++; break;

case 1:a++;

}

case 2: b++;

case 3: a++;

}

printf("a=%d,b=%d",a,b);

return 0;

}

该程序的输出结果是( )

A、a=3,b=3 B、a=1,b=3 C、a=2,b=3 D、a=1,b=2

8、以下程序执行后sum的值是：( )

#include<stdio.h>

int main( )

{

int i,sum;

for (i=1;i<6;i++) sum+=i;

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

return 0;

}

A、15 B、14 C)、0 D、不确定

1. 程序阅读题（每题5分，共20分）

1、以下程序运行所产生的输出是\_\_\_\_\_\_\_\_\_\_。

#include<stdio.h>

int main( )

{

int a,s,n,count;

a=2;

s=0;

n=1;

count=1;

while(count<=7){

n=n\*a;

s=s+n;

++count;

}

printf("s=%d",s);

return 0;

}

2、以下程序运行所产生的输出是\_\_\_\_\_\_\_\_\_\_。

#include<stdio.h>

int main()

{

int a=1,b=5,c=4;

while(c-->0&&++a<6)

b=b-1;

printf("%d,%d,%d\n",a,b,c);

return 0;

}

1. 写出下面程序的功能和执行后的显示结果。

#include <stdio.h>

int main(void)

{ int n=10, i, j, t, a[10]={7,3,66,-5,22,90,34,11,58,19};

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

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

if (a[j] < a[j+1])

{

t=a[j];

a[j]=a[j+1];

a[j+1]=t;

}

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

printf("%3d",a[i]);

return 0;

}

4、写出下面程序的功能和执行后的显示结果

#include<stdio.h>

int main()

{

int arr[5][5] = { 0 };

int a = 1;

int b = 1;

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

{

arr[a][0] = 1;

for (b = 0; b <=a; b++)

{

if(a==b)

arr[a][b] = 1;

if(a>1 && b>=1)

arr[a][b] = arr[a - 1][b - 1] + arr[a - 1][b];

printf(“%d ”, arr[a][b]);

}

printf("\n");

}

return 0;

}

1. 改错题（15分。注意：不得增行或删行，也不得更改程序结构。）
2. 现有一个大小为10的数组，编写将数组的前3个元素和随后的7个元素互换的程序。

/\* 1 \*/ #include<stdio.h>

/\* 2 \*/ int main(){

/\* 3 \*/ int m=3,n=10;

/\* 4 \*/ int a[n];

/\* 5 \*/ for(int i=0;i<n;i++){

/\* 6 \*/ scanf("%d",a[i]);

/\* 7 \*/ }

/\* 8 \*/ int k,temp,j;

/\* 9 \*/ for(int i=0;i<m;i++)

/\* 10 \*/ {

/\* 11 \*/ temp=a[0];

/\* 12 \*/ for(j=0;j<n;j++)

/\* 13 \*/ {

/\* 14 \*/ a[j]=a[j+1];

/\* 15 \*/ }

/\* 16 \*/ a[0]=temp;

/\* 17 \*/ }

/\* 18 \*/ for(int i=0;i<n;i++){

/\* 19 \*/ printf("%d ",a[i]);

/\* 20 \*/ }

/\* 21 \*/ return 0;

/\* 22 \*/ }

2、输入由小写字母构成的字符串s，其中有且仅有一个字母只出现了一次，输出它在s中的索引。

/\* 1 \*/ #include<stdio.h>

/\* 2 \*/ int main() {

/\* 3 \*/ char s[1000];

/\* 4 \*/ int cnt[200] = {0};

/\* 5 \*/ scanf("%s", &s);

/\* 6 \*/ for(int i = 0; s[i]; i++) {

/\* 7 \*/ cnt[s[i]]++;

/\* 8 \*/ }

/\* 9 \*/ for(int i = 0; s[i]; i++) {

/\* 10 \*/ if(cnt[i] == 1) {

/\* 11 \*/ printf("%d", i);

/\* 12 \*/ break;

/\* 13 \*/ }

/\* 14 \*/ }

/\* 15 \*/ return 0;

/\* 16 \*/ }

1. 完全程序题（每空3分，共30分）

1、读入一个整数x，用二分法求x的平方根，只需要输出整数部分。

#include<stdio.h>

int main() {

int x, mid;

scanf("%d", &x);

int left = 0, right = x, ans;

while( (1) ) {

mid = (left + right) / 2;

if( (2) ) {

ans = mid;

(3) ;

}

else {

right = mid - 1;

}

}

printf("%d", ans);

return 0;

}

2、读入两个只包含小写字母的字符串s和t，其中t是由s增加一个小写字母后乱序排列生成，也就是t恰好包含s中所有字符和一个额外的小写字母。输出这个额外的字符。

#include<stdio.h>

#define M 1000

int main() {

char s[M], t[M];

int cnt\_s[200] = {0}, cnt\_t[200] = {0};

scanf( (4) );

for(int i = 0; s[i]; i++) {

cnt\_s[s[i]]++;

}

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

cnt\_t[t[i]]++;

}

for(int i = 'a'; (5) ) {

if( (6) ) {

printf("%c", i);

break;

}

}

return 0;

}

3、“双指针”思想是指在遍历对象的过程中，同时使用两个指针进行快慢同向、相向等方式的扫描，这里的“指针”实际上指的就是索引。以下是一个运用相向双指针的例子。

读入整数n，S，以及一个长度为n的升序排列的整数数组a，判断a中是否存在一对不同位置的数相加为S。如果存在，输出其中任意一对在数组中的索引，否则输出NO。

思路：初始时两个指针分别指向第一个元素位置和最后一个元素的位置。每次计算两个指针指向的两个元素之和，并和目标值比较。如果两个元素之和等于目标值，则得到答案。如果两个元素之和小于目标值，则将左侧指针右移一位。如果两个元素之和大于目标值，则将右侧指针左移一位。移动指针之后，重复上述操作，直到找到答案。

#include<stdio.h>

#define N 2e5

int main() {

int n, S, a[N];

int flag = 0;

scanf("%d%d", &n, &S);

for(int i = 0; i < n; i++) scanf("%d", &a[i]);

int left = 0, (7) ;

while(left < right) {

int sum = (8) ;

if( (9) ) {

printf("%d %d", left, right);

flag = 1;

break;

}

else if(sum < S) left++;

else (10) ;

}

if(!flag) puts("No");

return 0;

}

1. 编程题（每题10分，共20分）

其中90％为程序正确性评分，10％为程序其他质量指标评分（可读性、书写风格等）。所有计算过程中的数据视为始终在int范围内。

1. 编写程序，输入正整数n，m，k，再输入一个n行m列的矩阵A和一个m行k列的矩阵B，用矩阵乘法计算AB并输出。

示例：当n=m=k=2，输入 A = [], B = [],

则输出为 AB=[] = []

1. 回文数猜想：对于一个正整数a，记其反序数为a’，令a=a+a’，得到新的a和a’，如此重复若干步后a一定成为回文数。其中，反序数指将原数的数位倒序排列的数，例如123的反序数为321，120的反序数为21。回文数指反序数仍等于本身的数，如8，99，232，1551等。

现在你想尝试验证这个猜想。编写程序，输入一个正整数a，重复以上步骤，输出a成为回文数所需的步数step。

示例：输入 a = 37， 则计算步骤如下：

37 + 73 = 110

110 + 011 = 121

得到回文数。所以输出 step 值为 2。