******华中科技大学计算机与科学技术学院2020~2021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 (A卷)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考试方式** | **闭卷** | **考试日期** | **2021-01-08** | **考试时长** | **150 分钟** |
| **专业班级** |  | **学 号** |  | **姓 名** |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **题号** | **一** | **二** | **三** | **四** | **五** | **六** | **七** | **总分** | **核对人** |
| **分值** | 10 | 14 | 24 | 24 | 28 |  |  | 100 |  |
| **得分** |  |  |  |  |  |  |  |  |  |

**注意：试卷最后一页附有运算符的优先级和结合性表、相关库函数声明。该页纸可撕下做草稿。**

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

**一、计算表达式的值**（写出每小题的计算结果，**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。

char stra[ ] = "helloword", strb[15] = { 'h','c','a','f','e','\0' };

int a = 7, b = 4, c = 3;

1. strlen(stra) 的值为：

(2) strlen(strb) 的值为：

(3) 表达式(a=7, b=3, a>b?a++:b++,a+b)的值为：

(4) 表达式 a += a \*= a执行完成后，a的值为：

(5) 表达式 !(a-b)+c-2 || b + c/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 st {

int n;

struct st \*next;

};

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

int b[6] = {2,3,5,6,7,9}, \*q = &b[1];

#define S(r) r/r

int x=2, y=4;

1. ++p->n 的值为： (7) (\*p++).n 的值为：

(8) p->next->n 的值为： (9) \*(q+1)的值为：

(10)表达式S(x+y)的值为：\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

**二、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。每题2处错，正确指出并改正1个错得1分，如果将正确代码改成错误表示则扣1分。本大题共7小题，每小题2分，共14分。）

1、使用getchar函数读取字符，并把字符输出到屏幕上。

|  |  |
| --- | --- |
| #include<stdio.h>  int main( )  { | while(c=getchar( )!=EOF) printf("%c",c);  return 0;  } |

2、求出以下分数序列的前n项之和：s=2/1 + 3/2 + 5/3 + 8/5 + 13/8 + 21/13 …。例如，若n=5，程序的输出应为：The value is: 8.391667。

|  |  |
| --- | --- |
| #include<stdio.h>  int func(int n)  {  int a=2, b=1, c, i;  double s = 0.0;  for(i=1; i<=n; i++) {  s = (double)a/b;  c=a; a=a+b; b=c;  } | return s;  }  int main()  {  int n=5;  printf("The value is: %f", func(n));  return 0;  } |

3、下面程序实现按照逐个字符的方式将数组s输出在屏幕上。

|  |  |
| --- | --- |
| #include<stdio.h>  int main( )  {  char s[5]="hello";  int i=0; | while(s[i]) {  printf("%s",s[i]); i++;  }  return 0;  } |

4、下面程序实现求一维数组a中值为偶数的元素之和，该程序的输出应为：The result is: 62。

|  |  |
| --- | --- |
| #include <stdio.h>  int sum ( int a[ ],int n )  { int i,s=0;  for ( i=0; i<n; i++)  if (a[i] % 2 == 0) s = s + i;  return s;  } | int main()  {  int a[10]={10,4,2,7,3,12,5,34,5,9}, s;  s = sum(a[10],10);  printf("The result is: %d\n", s);  return 0;  } |

5、下面程序用函数实现两个整数的交换。

|  |  |
| --- | --- |
| #include<stdio.h>  void swap(int \*p1, int \*p2)  {  int \*p;  p=\*p2;  \*p2=\*p1; \*p1=p;  } | int main()  {  int a=4,b=6;  swap(a,b);  printf("%d,%d", a, b);  return 0;  } |

6、下面程序利用函数指针p求三个整数中的最大值。

|  |  |
| --- | --- |
| #include <stdio.h>  int max(int x, int y)  {  return x > y ? x : y;  }  int main(void)  { | int \*p(int, int) = max;  int a=3, b=5, c=2, d;  d = \*p( p(a, b), c);  printf("最大的数是: %d\n", d);  return 0;  } |

1. 下面程序使指针p指向数组x中最后一个学生的信息，利用p输出该学生的姓名。

|  |  |
| --- | --- |
| include <stdio.h>  struct stu {  int score;  char name[20];  };  struct stu x[5]={{97,"Marry"},{80,"John"},  {78,"Jack"},{87,"Mark"},{93,"James"}}; | int main()  {  struct stu \*p ;  p = x[4];  printf("The 5th student name is %s", p.name);  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. 下列表示中哪些是关键字，哪些是合法标识符？

print\_3d db8 aBc a123 %x a+b 123 \_3ax case union getchar

2. 以下说明中，正确的有哪些？

（1）int a[5]=(0,1,2,3,4,);

（2）int a[ ‘0’ ];

（3）int a[3]={0,1,2} ;

（4）int a{5}={10\*1};

（5）char f[ ][4]={“Mon”, “Tue”, “Wed”, “Thu”, “Fri”, “Sat”, “Sun”};

（6）char  a[3]={‘a’,’b’,’c’};

（7）int a[n][4]={1, 2, 3}, n=10;

3. 请写一个C表达式，将int型数x向左循环移动n位，循环左移即把移出的高位放到该数的低位，n值小于int型数所占的位数。

4. 请定义一个标准宏MIN(x,y,z)，这个宏返回三个参数中最小的一个。

5. 请说明下面声明中p1和p2的含义与区别。

int x=2;

const int \*p1 =&x;

int \*const p2 =&x；

6. 请根据下面各题的解释，写出对应的声明语句。  
(1)a是一个指向有10个整型数数组的指针。

(2) b是一个有10个元素的指针数组，数组元素是指向函数的指针，该函数有一个整型参数并返回一个整型数。

|  |  |
| --- | --- |
| **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、请写出下面程序的运行结果。 | 2、请写出下面程序的运行结果。 |
| #include<stdio.h>  int main(void)  { int k=3, n=0x102;  printf("n=%d\n", n);  do {  k--;  n /= 10;  if(!n) {  printf("%d", -1); return 0;  }  } while(k>1);  printf("%d", n % 10);  return 0;  } | #include <stdio.h>  int a=10;  void fun(int y)  {  static int b=20;  a += y; b += y;  printf("a=%d, b=%d\n", a, b);  }  int main(void)  { int b=1;  while(a<13) fun( b);  printf("a=%d, b=%d\n", a, b);  return 0;  } |
| 运行结果： | 运行结果： |

3. 假设输入为(↙表示换行）：cpp2005program↙，请写出下面程序的运行结果。

#include<stdio.h>

enum { LET, DIG} ;

int main( void ) 运行结果：

{

char ch;

int state=LET;

while((ch=getchar( )) != '\n') {

switch(state) {

case LET:

if('0'<=ch&&ch<='9') {

state=DIG; putchar('\_');

}

break;

case DIG:

if('a'<=ch&&ch<='z') {

state=LET; putchar('\_');

}

break;

}

putchar(ch);

}

return 0;

}

1. 请写出下面程序的运行结果。

|  |  |
| --- | --- |
| #include <stdio.h>  void func1(int x[],int n);  int func2(int x[],int n);  #define N 7  int main()  {  int n,i,a[N]={16,9,9,4,20,16,16};  func1(a,N);  for( i=0; i<N; i++)  printf("%d ",a[i]);  printf("\n");  n=func2(a,N);  for( i=0; i<n; i++)  printf("%d ",a[i]);  return 0;  } | void func1(int x[],int n)  {  int t,i,m;  for(m=0,i=1; i<n; i++) {  if(x[i]<x[m]) m=i;  }  if(m) { t=x[0]; x[0]=x[m]; x[m]=t; }  }  int func2(int x[], int n)  {  int w, r;  for(w=0,r=1; r<n; r++) {  if( x[r]!=x[w] ) { ++w; x[w]=x[r]; }  }  return(w+1);  } |

运行结果：

5. 请写出下面程序的运行结果。

|  |  |
| --- | --- |
| #include<stdio.h>  void fun1(char \*s1,char \*s2)  {  char \*pw,\*pr,\*pt,\*p;  pw = pr = s1;  while(\*pr != '\0') {  for(pt=pr,p=s2; \*p!= '\0'; pt++,p++ ){  if(\*pt != \*p )  break;  }  if(\*p== '\0') {  pr = pt;  continue;  }  \*pw++=\*pr++;  }  \*pw = '\0';  } | void fun2(char \*s1,char \*s2)  {  while(\*s2 != '\0') {  \*s1++ = \*s2;  if(\*(s2+1) != '\0') s2++;  s2++;  }  \*s1 ='\0';  }  int main(void)  {  char str1[80],str2[ ]="wrs";  fun2(str1,"upwards\_downwards")；  printf("%s\n",str1);  fun1(str1,str2);  printf("%s\n",str1);  return 0;  } |

运行结果：

6. 请写出下面程序的运行结果。

|  |  |
| --- | --- |
| #include<stdio.h>  void fun1(char \*\*p,char \*\*q)  {  char \*t;  t=\*p,\*p=\*q,\*q=t;  }  void fun2(char \*a[ ])  {  char \*\*p1,\*\*p2;  for(p2=p1=a; \*p1!=NULL; p1++)  **;**  p1--;  while(p1>p2) {  fun1(p1,p2);  p2++,p1--;  }  } | int main( )  {  char \*s[]={"red",  "blue",  "green",  "black",  NULL  };  int i;  for(i=0; s[i]!=NULL; i++)  printf("%c ", \*(s[i]+i));  printf("\n");  fun2(s);  for(i=0; s[i]!=NULL; i++)  printf("%s\n", s[i]);  return 0;  } |

运行结果：

|  |  |
| --- | --- |
| **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，通过填空来完善程序。本大题共14空，每空2分，共28分。）

1. [程序说明] 下面的程序将输入正文的多行文本存放在一个足够长的一维字符数组中，存放时将相连的多个空格用一个空格代替。然后在输出该一维数组时，能够像输入时一样以多行方式显示。

请在①、②、③处填入适当的代码，使程序能够输出正确结果。

|  |  |
| --- | --- |
| #include <stdio.h>  #define N 1000  int main(void)  {  char c,a[N];  int flag=0,i=0;  while ( ① ) {  if (c != ' ') {  a[i++] = c;  flag = 0;  } | else {  if ( ② )  a[i++] = c;  flag = 1;  }  }  ③ ;  printf("%s", a);  return 0;  } |

① ② ③

1. [程序说明] 函数原型 char \*strnins(char \*s, char \*t, int n); 所声明的函数strnins，功能是将字符串t从字符串s的第n个字符位置插入；若字符串s的长度小于n，则将字符串t插在s的串尾。函数返回值为插入操作完成后字符串s的地址。

下面是函数strnins的定义，请在①、②、③处填入适当的代码，使函数能够正确实现功能。

|  |  |
| --- | --- |
| char \*strnins(char \*s, char \*t, int n)  {  int i, j, k;  for (i=0; i<n; i++) { /\* 找到插入点 \*/  if (s[i]== '\0') ① ;  }  for (k=i; s[k]!='\0'; k++) /\* 找到串尾 \*/  **;**  for (j=0; t[j]!='\0'; j++) /\* 计算串t的长度 \*/  **;** | while (k >= i) { /\* 将插入点到串尾的字符向后搬移j个字符位置 \*/  s[k+j] = s[k];  ② ;  }  for (k=0; k<j; k++) { /\* 插入操作 \*/  s[i+k] = ③ ;  }  return s;  } |

① ② ③

1. [程序说明] 下面程序输入一行长度无限制超长字符串，用一个先进先出，且每个结点接受一个输入字符的单向链表存放这个字符串；程序遍历输出链表中的所有字符，接下来，将该超长字符串无冗余地存放到一个通过动态存储分配创建的字符数组中，最后输出该超长字符串。

请在①、②、③、④处填入适当的代码，使程序能够正确实现功能。

#include <stdio.h>

#include <stdlib.h>

typedef struct str\_node {

char ch;

struct str\_node \*next;

} TLONG\_STR, \*PLONG\_STR;

void creat(PLONG\_STR head);

void out(PLONG\_STR head);

int long\_str\_len(PLONG\_STR first);

int main()

{

PLONG\_STR p, head = (PLONG\_STR)malloc(sizeof(TLONG\_STR));

char \*pch;

long len;

int i;

creat(head);

out(head);

putchar('\n');

len = long\_str\_len( ① );

pch = (char \*)malloc(len+1);

for (i=0,p=head->next; p!=NULL; p=p->next)

pch[i++] = p->ch;

pch[i] = '\0';

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

return 0;

}

void creat(PLONG\_STR head)

{

PLONG\_STR tail = head;

char ch;

while ((ch=getchar()) != EOF) {

tail->next = (PLONG\_STR)malloc(sizeof(TLONG\_STR));

tail = ② ;

tail->ch = ch;

}

tail->next = NULL;

}

void out(PLONG\_STR head)

{

PLONG\_STR p;

for (p=head->next; ③ ; p=p->next)

putchar(p->ch);

putchar('\n');

}

int long\_str\_len(PLONG\_STR first)

{

if (first == NULL)

return 0;

return 1 + ④ ;

}

① ②

③ ④

1. [程序说明] 下面程序用来实现一个文件分割工具FileSplit，将超大文件分割成大小为1G字节的若干个二进制文件，最后一个文件可能不满1G字节。超大文件的文件名通过命令行输入，分割后的文件名：out0000，out0001，out0002,......

请在①、②、③、④处填入适当的代码，使程序能够正确实现功能。

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAXBUF (1<<16) /\* 为了提高读写效率，每次读写64k字节 \*/

#define MAX 16384 /\* 16k \* 64k = 1G \*/

#define OUTFILE "out"

int main (int argc, char \*argv[])

{

char filename[50], buf[MAXBUF], ext[10];

FILE \*fin, \*fout;

int filenum = 0, recnum = 0, rdnum;

if (argc < 2) {

printf("error command format!\n"); exit(-1);

}

if ((fin = fopen( ① , "rb")) == NULL) {

printf("failed to open the file!\n"); exit(-2);

}

strcpy(filename, OUTFILE);

itoa(filenum+10000, ext, 10);

strcat(filename, ext + 1);

fout = fopen(filename, ② );

while ((rdnum = fread(buf, 1, MAXBUF, fin)) >= MAXBUF) {

fwrite(buf, 1, rdnum, fout);

recnum++;

if (recnum >= MAX) {

③ ;

recnum = 0;

④ ;

strcpy(filename, OUTFILE);

itoa(filenum+10000, ext, 10);

strcat(filename, ext + 1);

fout = fopen(filename, "wb");

}

}

fwrite(buf, 1, rdnum, fout);

fclose(fout); fclose(fin);

return 0;

}

① ②

③ ④

**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(char \*s); | 计算字符串s中字符的个数(不包括 '\0' ) |
| char \*strcpy(char \*s, char \*t); | 把t指向的字符串复制到s中 |
| char \*strcat(char \*s, char \*t); | 把串t连接到s的尾部，返回s |
| void malloc(unsigned size); | 分配size字节的存储区，该存储区未初始化 |
| void free(void \*p); | 释放p所指的内存 |
| int fread(void \*ptr, unsigned,size, unsigned n, FILE \*fp); | 从文件fp中读取长度为size字节的n个数据项，存到ptr所指向的内存区,返回所读的数据项个数 |
| int fwrite(void \*ptr, unsigned size, unsigned n, FILE \*fp); | 把ptr所指向的n\*size个字节输出到文件fp中 |
| char \*itoa (int x, char \*s, int base ); | 把整数x转换成base进制的字符串s |
| int fclose(FILE \*fp); | 关闭文件fp |
| FILE \*fopen(char \*filename, char \*mode); | 以mode方式打开文件filename |