课后作业8

字符'\0'的ASCII码值为0。

TF

字符串常量就是用一对双引号括起来的字符序列，它有一个结束标志 '\0'。

TF

1-3

以下代码实现从键盘输入一个字符串，并将小写字母转换为相应的大写字母后输出，其他字符原样输出。

while ((ch = getchar()) != '\0') {

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

ch = ch - 'a' + 'A';

putchar(ch);

}

TF

本题错误的原因是输入字符串时并不会输入’\0’，而是输入回车键结束，所以，这个题应该把while ((ch = getchar()) != '\0')这一行代码修改为while ((ch = getchar()) != '\n')

1-4

"a" 和 'a'是等价的。

TF

1-5

代码char s[5] = "Happy";将导致数组越界存储。

TF

1-6

执行语句int \*p = 1000;后，指针变量p指向地址为1000的变量。

TF

这种写法是不规范的，在指针变量初始化时，应该写为int \*p = &..的形式，直接这样写，会让编译器报错

1-7

语句int \*p, q, r; 定义了3个指针变量。

TF

一个指针变量，两个整数型常量

1-8

执行语句int \*p; 后，指针变量p只能指向int类型的变量。

TF

1-9

不同类型的指针变量是可以直接相互赋值的。

TF

1-10

语句int \*p; \*p = 50;执行时，不会有任何错误。

TF

理由同第6题

1-11

代码int \*p; scanf("%d", p);，由于指针变量未初始化，可能导致段错误。

TF

正确，此时指针变量未初始化，因此p指向随机的一个地址。如果这个地址的已经被占用了，那么修改这个地址储存的值就会导致段错误。

1-12

代码int \*p = NULL; scanf("%d", p);，由于指针变量已初始化，不会导致段错误。

TF

NULL是一个特殊的地址，无法修改，仍然会段错误

那么指针正确初始化的做法是什么呢？（已学过三种）

1. 通过取值运算符&
2. 动态内存分配
3. 数组名

1-13

代码char \*p = "hello"; \*p = 'H';将导致段错误。

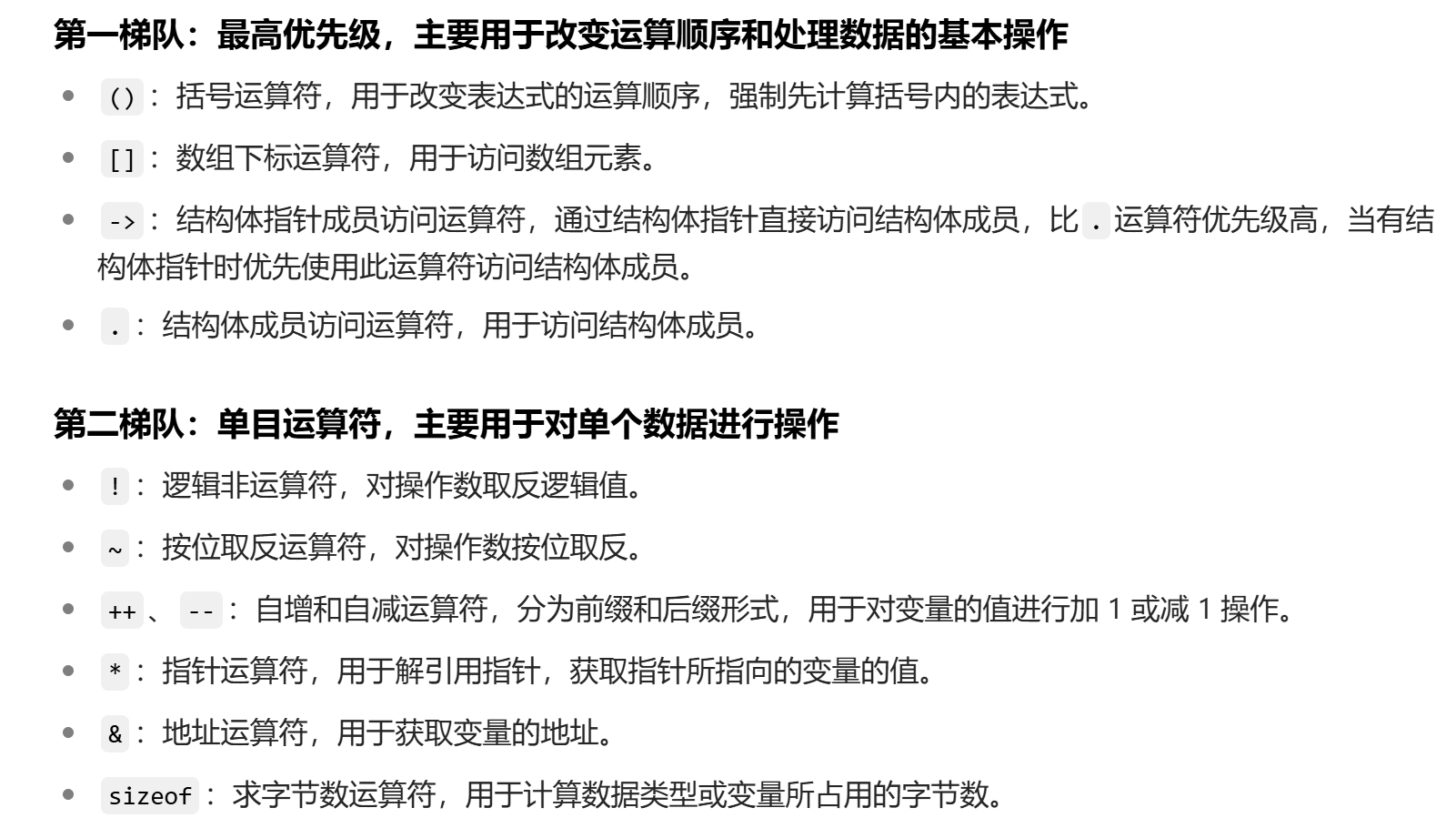
TF

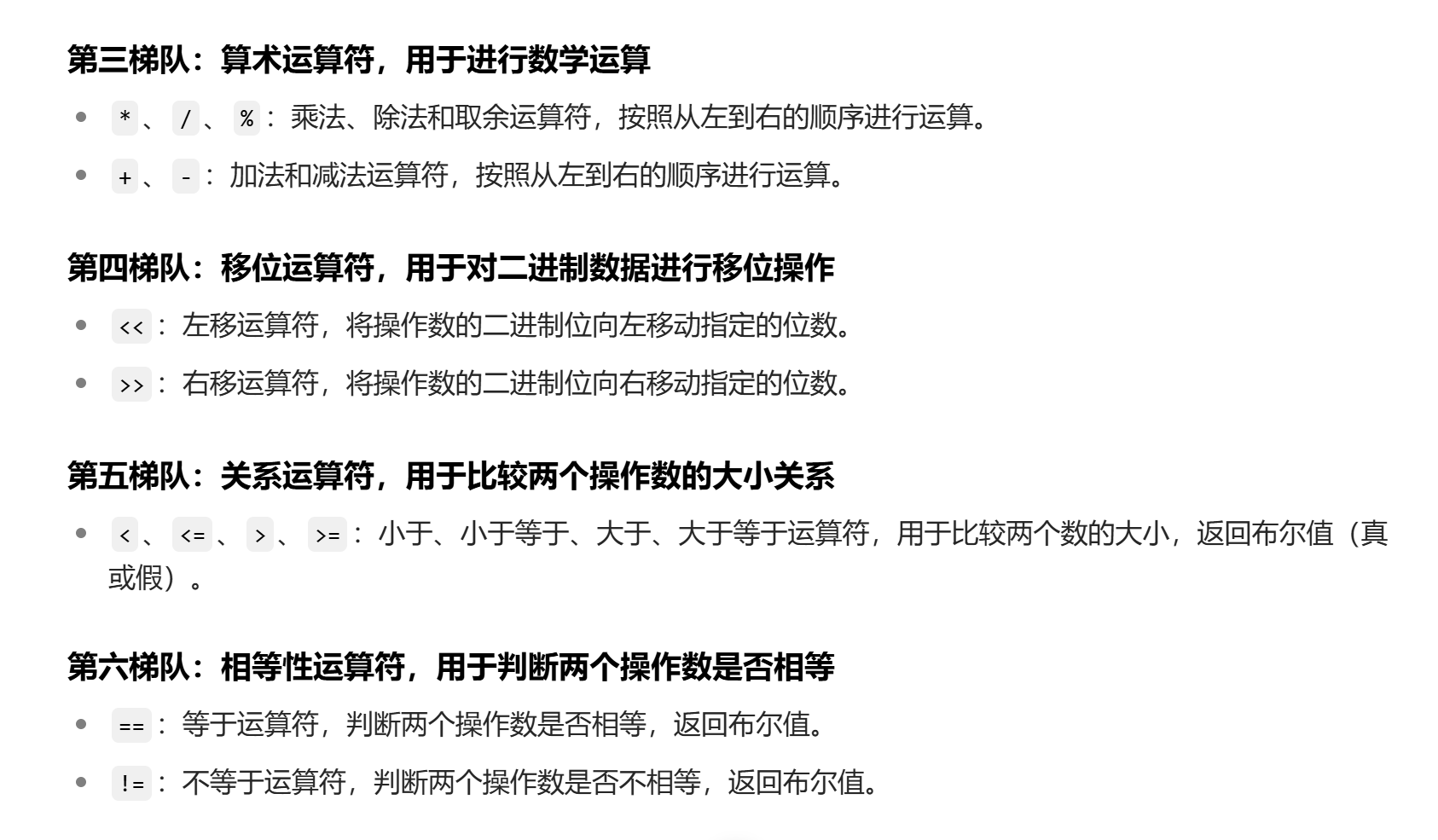
1-14

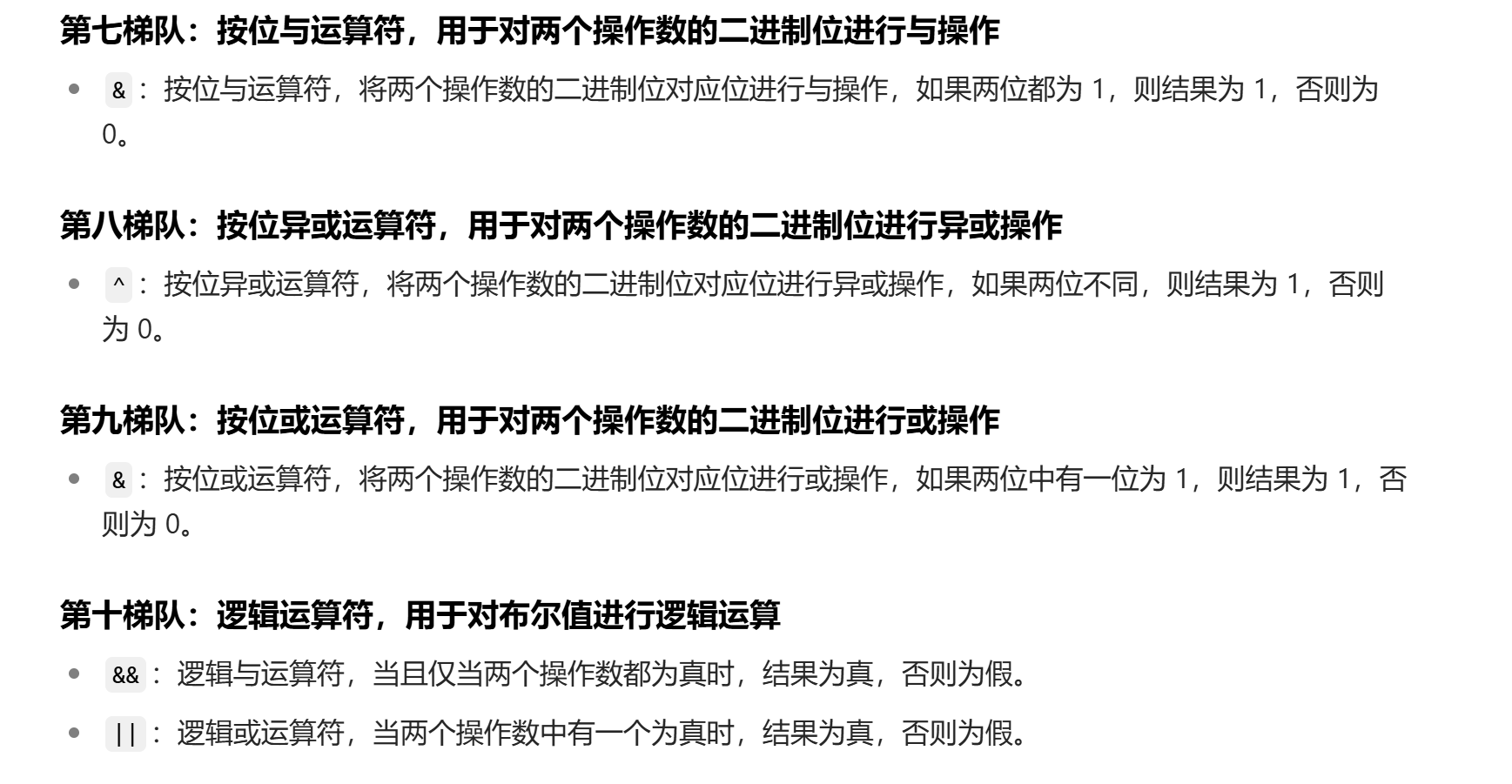
定义变量int a = 3, \*p = &a;，则\*p++相当于a++。

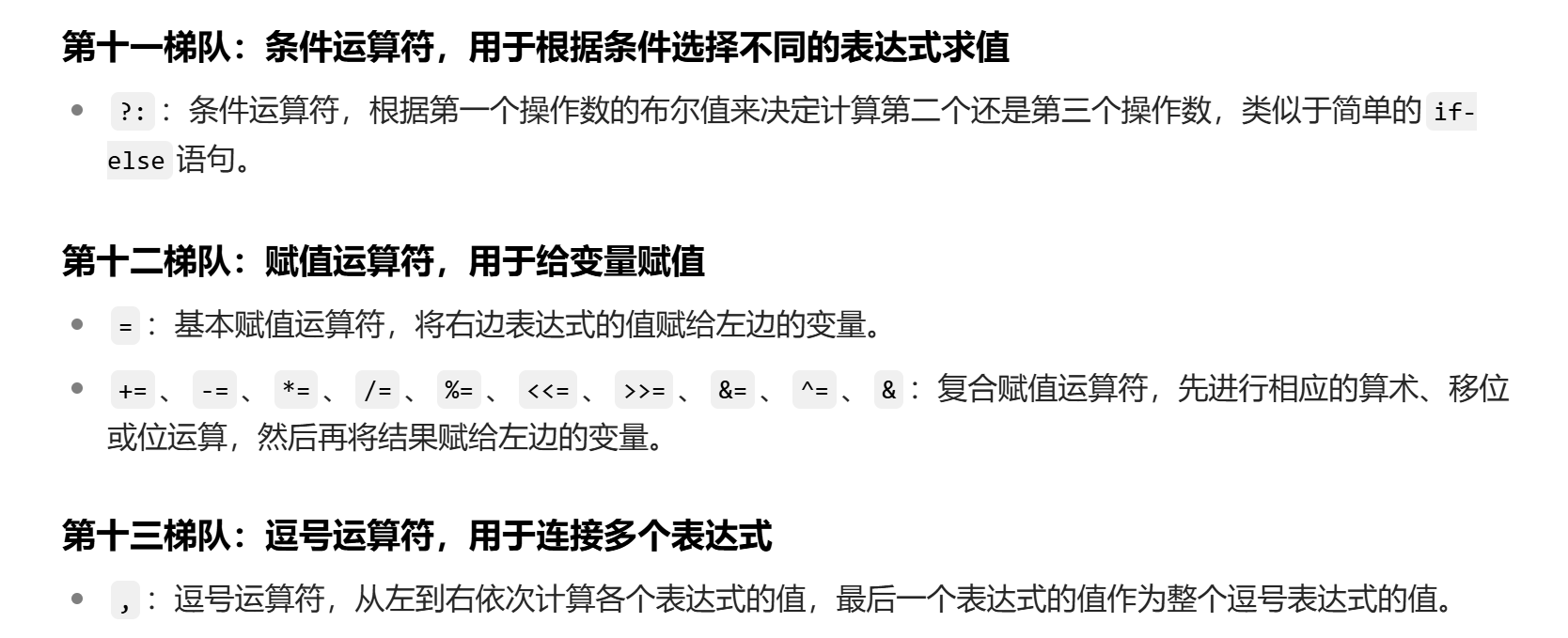
TF

错误。运算符的顺序：++高于\*









1-15

要通过函数调用来改变主调函数中某个变量的值，可以把指针作为函数的参数。

TF

1-16只要将多个指针作为函数的参数，函数就一定会返回多个值。

TF

2-1

设有数组定义：char array [ ]="China"; 则数组 array所占的空间为（）。

A.4个字节B.5个字节

C.6个字节D.7个字节

2-2

下述对C语言字符数组的描述中错误的是（）。

A.字符数组可以存放字符串

B.字符数组中的字符串可以整体输入、输出

C.可以在赋值语句中通过赋值运算符"="对字符数组整体赋值

D.不可以用关系运算符对字符数组中的字符串进行比较

2-3

有以下定义：char x[ ]="abcdefg"; char y[ ]={'a', 'b', 'c', 'd', 'e', 'f', 'g'}; 则正确的叙述为（）。

A.数组x和数组y等价 B.数组x和数组y的长度相同

C.数组x的长度大于数组y的长度 D.数组x的长度小于数组y的长度

2-4

对于如下变量定义及初始化，与m = n等价的表达式是（ ）。

int m, n = 5, \*p = &m;

A.m = \*p

B.\*p = n

C.m = &n

D.\*p = m

2-5

以下选项中，对基本类型相同的指针变量不能进行运算的运算符是 ( )。

A.+ B.- C.= D.==

减法可以，算不同指针地址的地址差，加法感觉可能会发生段错误，越界访问

2-6

下面程序的输出结果是( )。

# include <stdio.h>

void fun (int \*x, int \*y)

{

printf("%d%d", \*x, \*y);

\*x = 3;

\*y = 4;

}

int main(void)

{

int x=1, y=2;

fun(&x, &y);

printf("%d%d", x, y);

return 0;

}



A.

2134



B.

1212



C.

1234



D.

2112

Judge Result

Accepted

Score

1 Point(s)

2-7

#include <stdio.h>

void fun (int \*x, int y)

{

printf("%d%d", \*x, y);

\*x=3;

y=4;

}

int main(void)

{

int x = 1, y = 2;

fun(&y, x);

printf("%d%d", x, y);

return 0;

}

A.1234

B.1221

C.2131

D.2113

选项（ ）与以下字符数组定义等价。

static char s[6] = {'H', 'a', 'p', 'p', 'y', '\0'};

A.static char s[6] = {'H', 'a', 'p', 'p', 'y'};

B.static char s[6] ="Happy";

C.static char s[6] ={"Happy"};

D.static char s[6] = {'H', 'a', 'p', 'p', 'y', 0};

全选，因为是static，’\0’的ASCⅡ码是0，因此全都可以选

课后作业 9

1-1

数组的基地址是在内存中存储数组的起始位置，数组名本身就是一个地址即指针值。

TF

1-2

对于定义int a[10],\*p=a; 语句p=a+1;和a=a+1;都是合法的。

TF

a=a+1包不合法的

1-3

两个任意类型的指针可以使用关系运算符比较大小。

TF

1-4

冒泡排序效率较高，因为它只需要约  
IMG_256  
次比较。

TF



大O计算时间复杂度时会忽略常数项和低次项，最坏情况（n-1）\*n/2

但是通过以上操作以后会变成O（n^2）

1-5

执行以下代码后，变量x的值为042，变量y的值为87，变量z的值为0x4e。

int a[10] = { '\012', 34, '\x56', 78, 0x910 }, \*p = a + 1;

int x = \*p++;后置的++运算符在式子结束以后再运算

int y = ++\*p;

int z = \*++p;

TF

1-6

变量定义如下，指针变量q的值比指针变量p的值大sizeof(char)。

int a[5];

int \*p = a;

char \*q = (char \*)p + 1;

TF

1-7

定义数组int a[10];，则表达式(int)(&a + 1) - (int)&a的值为4。

TF

这个式子的值为1

类比拓展：表达式(int)(a + 1) - (int)a的值为\_\_\_4\_\_\_\_\_

1-8

函数申明和使用代码如下，当调用该函数时，实参数组b的10个元素的值单向传递给形参数组a的10个元素。

int fun(int a[10]);

int b[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

fun(b);

TF

指针传递的是地址！！！

1-9

Score 1

Author 陶煜波

Organization 浙江大学

形参为数组的函数在编译后都转化为形参为指针的函数，如int fun(int a[10])转化为int fun(int \*a)。

TF

Judge Result

Accepted

Score

1 Point(s)

1-10

Score 1

Author 陶煜波

Organization 浙江大学

字符串压缩代码如下，char \*q = p;是将指针变量p的值（字符串首字符的地址）赋值给指针变量\*q。

void zip(char \*p)

{

char \*q = p;

......

}

TF

Judge Result

Accepted

Score

1 Point(s)

1-11

Score 1

Author 张泳

Organization 浙大城市学院

字符串常量在内存中的存放位置由系统自动安排。

TF

Judge Result

Accepted

Score

1 Point(s)

1-12

Score 1

Author 张泳

Organization 浙大城市学院

字符串常量实质上是一个指向该字符串首字符的指针常量。

TF

Judge Result

Accepted

Score

1 Point(s)

1-13

Score 1

Author 张泳

Organization 浙大城市学院

调用printf函数，%s的格式输出字符串时，字符数组名、字符指针和字符串常量都可以作为输出参数。

TF

Judge Result

Accepted

Score

1 Point(s)

1-14

Score 1

Author 陶煜波

Organization 浙江大学

char s[128];使用scanf("%127s", s);输入时，将读取127个字符，并加上\0后返回，这样可以避免字符数组越界存储。

TF

Judge Result

Wrong Answer

Score

0 Point(s)

1-15

Score 1

Author 陶煜波

Organization 浙江大学

输出字符串puts(s);等价于printf("%s\n", s);。

TF

Judge Result

Accepted

Score

1 Point(s)

1-16

Score 1

Author 陶煜波

Organization 浙江大学

int sprintf(char \*string, char \*format [,argument,...])是字符串格式化函数，即将格式化的数据写入某个字符串中，例如

char str[128];

int pos = sprintf(str, "%d-%.2f-%s", 123, 3.14159, "abc");

str字符串的内容为"123-3.14-abc"，pos的值为输出的字符数12。

TF

Judge Result

Accepted

Score

1 Point(s)

1-17

Score 1

Author 陶煜波

Organization 浙江大学

代码char p[] = "Hello"; strcpy(p, "World");将导致段错误。

TF

Judge Result

Accepted

Score

1 Point(s)

1-18

Score 1

Author 陶煜波

Organization 浙江大学

代码char \*p = "Hello"; strcpy(p, "World");将导致段错误。

TF

Judge Result

Accepted

Score

1 Point(s)

1-19

Score 1

Author 张泳

Organization 浙大城市学院

调用strcmp函数比较字符串大小时，通常较长的字符串会较大。

TF

Judge Result

Accepted

Score

1 Point(s)

1-20

Score 1

Author 陶煜波

Organization 浙江大学

mystrlen函数实现了字符串有效长度计算。

int mystrlen(char \*s)

{

char \*p = s;

while (\*++p) ;

return p - s;

}

TF