**一、选择题：**

**1.以下代码的输出结果是多少( )**

int value[16];

int test(char value[]) {

return sizeof(value);

} **A.64 B.16 C.8 D.4**

**2.以下叙述中正确的是( )**

A.语言中的关键字不能作变量名，但可以作为函数名

B.标识符的长度不能任意长，最多只能包含16个字符

C.用户自定义的标识符必须“见名知义”，如果随便定义，则会编译错误

D.标识符可以由字母、数字和下划线组成，且第一个字符不得为数字

**3.以下关于内存泄漏和内存溢出描述错误的是（ ）**

A.内存溢出是指程序在申请内存时，没有足够的内存空间供其使用，出现out of memory

B.内存泄漏是指分配出去的内存不再使用，但无法回收

C.在内核中，内存溢出会立即出发死机

D.在内核中，内存泄漏不会造成访问异常地址

**4.设栈最大长度为3，入栈序列为1,2,3,4,5,6,则不可能得出栈序列是( )**

**A**.1, 2, 3, 4, 5, 6 **B**.2, 1, 3, 4, 5, 6 **C**.3, 4, 2, 1, 5, 6 **D**.4, 3, 2, 1, 5, 6

**5.以下代码打印结果分别为多少( )**

main()

{

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

int \*ptr = (int \*)(&a+1);

printf(“%d, %d”, \*(a+1), \*(ptr-1));

}  **A.1, 5 B.2, 4 C.2, 5 D.5, 5**

**6.以下程序输出结果( )**

int arr[] = {6, 7, 8, 9, 10};

int \*ptr = arr;

\*(ptr++) += 123;

printf(“%d, %d”, \*(ptr), \*(++ptr)); **A.130, 8 B.7, 7 C.7, 8 D.8, 8**

**7. 请写出如下程序运行结果( )**

#include <stdio.h>

#include <stdlib.h>

void getmemory(char \*p)

{

p=(char \*) malloc(100);

}

int main( )

{

char \*str=NULL;

getmemory(str);

strcpy(str,"hello world");

printf("%s/n",str);

free(str);

return 0;

**}** A.helloworld B.编译错误 C.null D.段错误

**8.以下程序完成的功能是( )**

Int fun(int c)

{

While (c)

{

If (c&1)

Cnt++;

c>>1;

}

Return cnt;

}

1. 获得二进制中1的个数 B.获得二进制中0的个数

C.奇偶检验 D.奇校验

**9.已知二叉树的后序遍历为: C E D B H G J I F A 中序遍历为: C B E D A G H F J I**

请写出前序遍历:

**10.**class B {public:

virtual ~B() { cout << "~B"; } };

class T :public B{public:

virtual ~T() { cout << "~T"; } };

B \* p = new T; Delete p;

上述程序片段的输出是：（ ）

A． ~T B．~B C．~B~T D．~T~B

**二、简答题：**

1. 请按照自己的理解解释如下概念:1)文件描述符 2)临界资源 3)时间片 4)原子操作 5)信号量 6)互斥锁 7)套接字 8)同步 9)阻塞 10)异步

2.进程和线程的区别？多进程和多线程的优缺点分别是什么？

3.僵尸进程是如何产生的？如何避免产生僵尸进程

4.解释死锁的现象？如何避免产生死锁？

5.请简述UDP和TCP服务器搭建函数接口及调用顺序

6.重载、隐藏和覆盖有什么区别？

**三、编程题:**

1. 请编写一个程序测试当前平台是大端存储还是小端存储(8分)
2. 请写出几种你常用的排序方式，并写出该种排序的时间复杂度以及算法稳定性(12分)
3. 现有链表定义形式如下：

typedef struct node

{

int data;

struct node \*next;

} linknode;

请编写程序实现:

1）int insertlinklist(linknode \*phead, int data); /\* 插入链表 \*/

2）int deletelinklist(linknode \*phead, int data); /\* 删除链表 \*/

3）int destroylinklist(linknode \*\*phead); /\* 销毁链表 \*/

4）O(n)时间复杂度完成链表倒置

1. 给定两个字符串，求出它们之间最长的相同子字符串的长度

5.创建三个任务,分别用来在终端打印A、B、C

但要确保打印出来的顺序总是A、B、C

6. 不用if，else、条件表达式、（？：）、switch 等判断语句，找  
出两个数 a,b 中较大的那个