第三周周测

选择题:每题3分,共45分

1. 下列关于IP地址说法正确的是？(BCD) 多选

A. 每台计算机的IP地址一定各不相同

B. 所有公网IP地址各不相同

C. IP地址用于在网络上查找一台主机

D. IPv6地址的数量比IPv4多很多

解析: 私有IP地址是路由器分配的，所以可能出现相同的情况。

2. 关于OSI模型的说法正确的是?（AC）多选

A. OSI模型一共7层

B. OSI模型就是我们工作中常用的TCP/IP模型

C. OSI模型降低了网络通信工程中各部分的耦合度，将工作流程标准化

D. OSI模型顺序可以打乱

解析：OSI模型并不是TCP/IP模型，OSI模型是一个工作流程，所以顺序是固定的。

3. TCP/IP模型的说法正确的是（ACD）

A. TCP/IP模型中应用层功能包含，规定应用功能，压缩优化加密，选择传输服务

B. TCP/IP模型中一共对应两个协议，TCP协议和IP协议

C. TCP/IP模型就是四层模型

D. TCP/IP模型是实际工作中使用的工作模型

解析：TCP/IP模型对应很多协议，不只是TCP IP 两个协议。

4. 下列关于协议的说法不正确的是（B）单选

A. HTTP协议是应用层协议

B.TCP协议是应用层协议

C.TCP协议和UDP协议都是传输层协议

D. IP协议是网络层协议

解析：TCP协议是传输层协议。

5.下面关于端口的说法正确的是（A）单选

A. 端口取值范围是0-65535

B. 我们常用10000以内的数字作为服务程序端口

C. 客户端程序和服务端程序都必须绑定一个端口

D. 一台主机中的不同程序使用端口可以重复

解析：为避免冲突实际经常使用大于10000的数作为自用端口，客户端没有必要绑定端口，端口使用不能重复。

6. 关于网络的基本概念说法正确的是?（ABD）多选

A. 网络通信过程是一个综合过程，有多方面工程师配合

B. ping命令可以测试网络的连通性

C. 网络信息是安全的

D. 网络数据传输过程中间往往需要经过很多节点

解析：网络信息并不安全，我们当前正处于一个维护网络安全为中心的网络发展阶段。

7. 下面的函数接口是非阻塞函数的是？（D）单选

A. accept()

B. recv()

C. recvfrom()

D. listen()

解析：上面函数中只有listen为非阻塞函数。

8. 以下不是HTTP协议的特点的是(C）单选

A. 请求/响应模式

B. 目前使用HTTP/1.1版本

C. 只能传输文本数据

D. 简单高效

解析：HTTP协议终止众多类型数据的传输不只是文本。其他几项均为http协议的特点。Http1.1版本中支持持久连接。

9. 关于HTTP请求响应说法正确的是?（B）单选

A. 请求类型有GET，POST两种

B. 响应码为404表示无法找到请求数据

C. 响应码为100 则一定能响应成功

D. 请求和响应格式中空行可以省略

解析：现在http请求类型与很多种，100表示响应被接受但是不一定保证成功，请求响应格式是严格规定的，不能改变。

10. 关于进程的说法正确的是？（B）单选

A.进程就是一个程序

B.进程占用系统运行资源

C.一个计算机中同时运行的进程数量是无限的

D.计算机同时只能运行一个进程

解析：进程和程序是两个概念，计算机中同时运行进程数量由计算机性能决定。

编程题

1. 给你一个长度为n的数组，其中只有一个数字出现大于n/2次，问如何快速找到这个数字（20分）

时间复杂度

def m\_element(num\_list):

return sorted(num\_list)[len(num\_list)//2]#排序后找索引为n/2的

print(m\_element([1,1,2,2,2,2]))

1. 假设给定一个由 4 位数字组成的数组，返回可以设置的符合 24 小时制的最大时间。最小的 24 小时制时间是 00:00，而最大的是 23:59。从 00:00 （午夜）开始算起，过得越久，时间越大。以长度为 5 的字符串返回。(30分)

**示例**   
输入：[1,2,3,4]  
输出："23:41"

from itertools import permutations#排列组合模块  
  
def largest\_time(A):  
 *"""* ***:param*** *A: List[int]* ***:return****: str  
 """* max\_time = 0 # 初始化最大时刻  
 res = '' # 结果字符串  
 for ht, hb, mt, mb in permutations(A): # 遍历  
 hour, minute = ht \* 10 + hb, mt \* 10 + mb # 求取时分  
 t = hour \* 60 + minute # 当前时刻（秒）  
  
 if hour < 24 and minute < 60 and t >= max\_time: # 时间合法的条件  
 res = "{}{}:{}{}".format(ht, hb, mt, mb) # 更新结果  
 max\_time = t # 更新最大时刻  
 return res  
  
print(largest\_time([1, 2, 3, 4]))

1. 输入一个正数n，输出所有和为n 连续正数序列。（20分）

例如输入15，由于1+2+3+4+5=4+5+6=7+8=15，所以输出3 个连续序列1-5、4-6 和7-8。

def find\_continuous\_seq(n):  
 if n<3:  
 return  
 small = 1#初始化序列最小值  
 big = 2#初始化序列最大值  
 sum = small+big  
 while small < (n+1)//2:  
 if sum == n:  
 print\_continuous\_seq(small,big)  
 while sum>n:  
 #如果序列累加的和比n大 将small向前移动  
 sum -= small  
 small += 1  
 if sum == n:  
 print\_continuous\_seq(small,big)  
 big += 1  
 sum += big  
  
def print\_continuous\_seq(small,big):  
 for i in range(small,big+1):  
 print(i,end=' ')  
 print()  
  
find\_continuous\_seq(15)