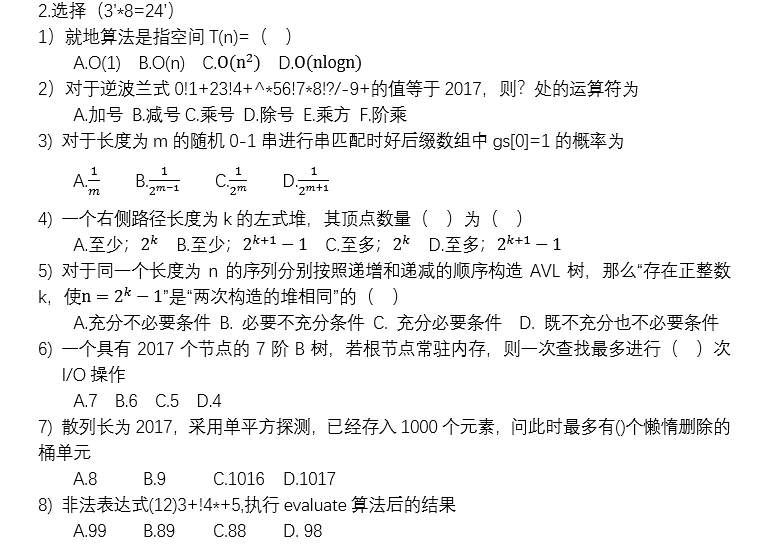
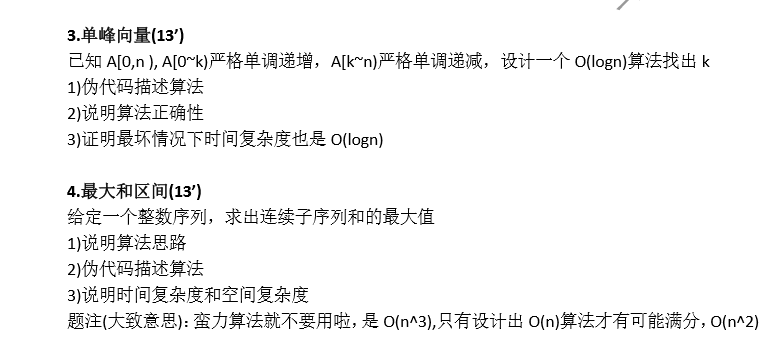


T F T T T T T F F F

A D B B C D B B

int search(int A[],int lo ,int hi){

int mi=(lo+hi)/2;

if(hi-lo<3) return mi=A[mi-1]<A[mi]?mi:mi-1;

if（A[mi]>A[mi-1]&&A[mi]>A[mi+1]）return mi;

if（A[mi]>A[mi-1]&&A[mi]<A[mi+1]）return search(A[],mi+1,hi);

if（A[mi]<A[mi-1]&&A[mi]>A[mi+1]）return search(A[],lo,mi);

}

Int sum(int A[],int n){

Int sum=0,i=0，max=A[0];

While(i<n){

Sum+=A[i];

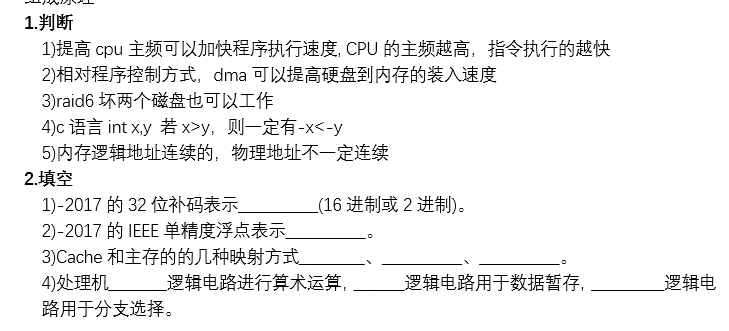
{Max=max<sum?sum:max ;}

If(sum<0)sum=0;

i++;}

Return max;

}

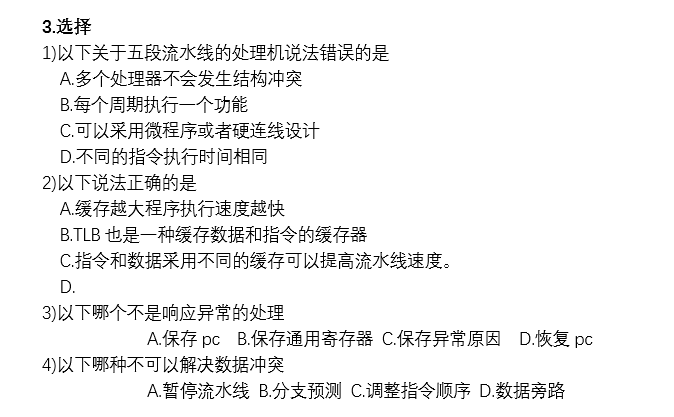


T T T F T

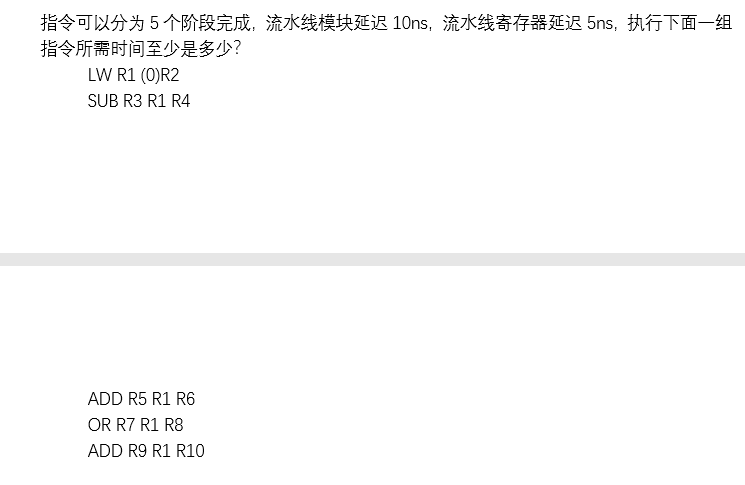
FFFFF81F C4 FC 20 00

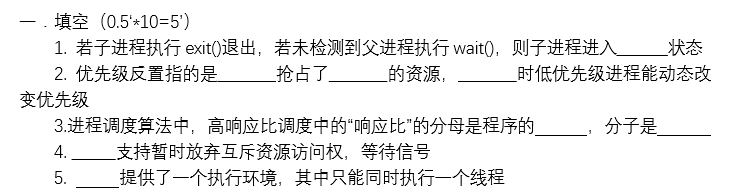
直接映射，组相联映射，全相联映射，

组合 时序 控制

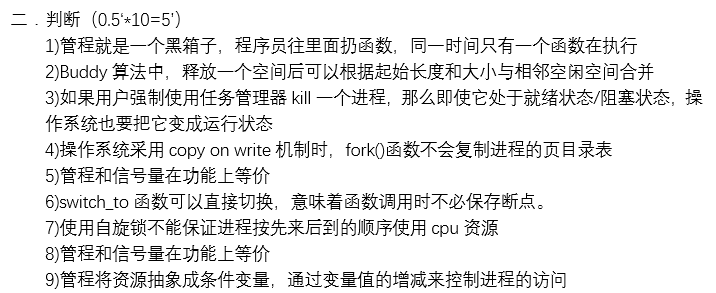


A C C B

至少：算上数据旁路且先写后读：5+2+1+1+1=10\*15=150ns

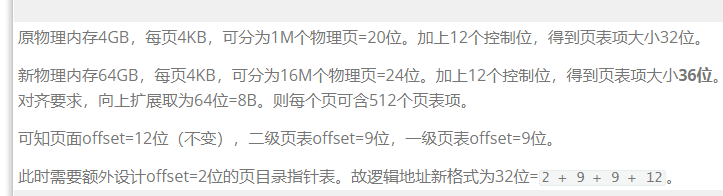


僵尸 低优先级 高优先级 动态优先级继承 执行时间 等待时间+执行时间 …管程 管程



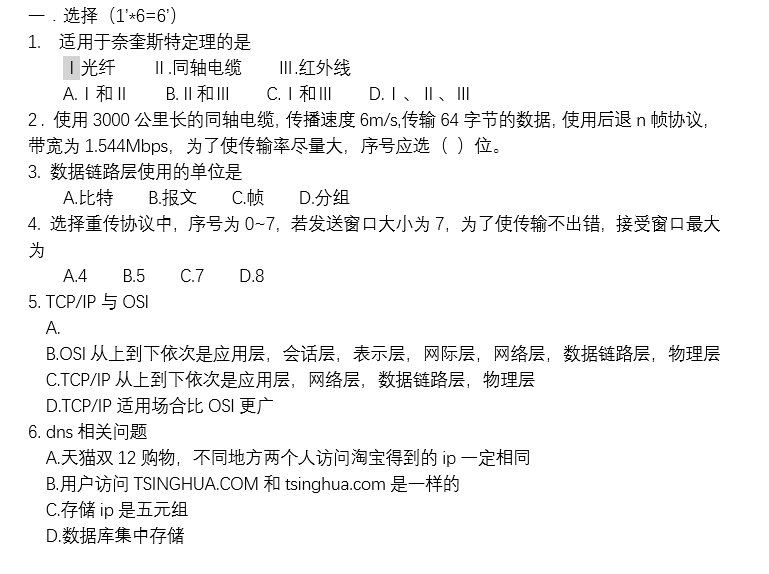
T T T F T F T T T



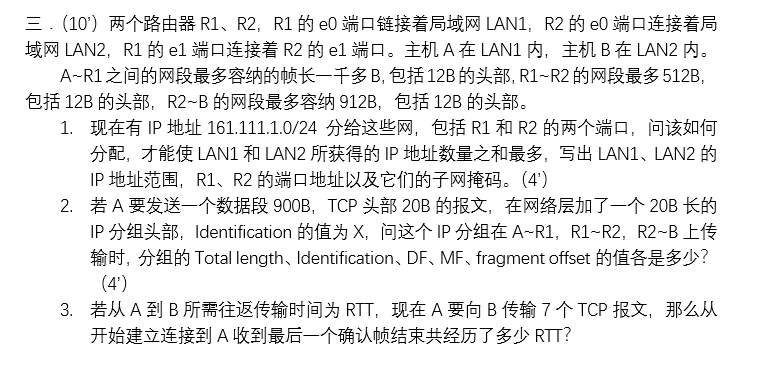




会，在第一个人拿左叉子后，第二位哲学家也拿左叉子，同时第二位哲学家为第一位右手边，之后第三位再右侧的哲学家拿起左叉子，以此类推，最终则死锁

2允许：第一位和第三位互不冲突

D log2（500ms\*0.193/64 ） C A B



A192.166.1.0/25 B192.166.1.192/25 A：1-254 B193-254

R1 e0 192.166.1.1 e1 192.166.1.129 要给中间的子网分

R2 e0 192.166.1.193 e1 192.166.1. 130

子网掩码懒得写了

2.A-R1 940 x 0 0 0

R1-R2 500（480） x 0 1 0 460（440） x 0 0 60

R2 B 500 x 0 1 0 460 x 0 0 60

3.1.5个rtt建立连接同时第1.5个rtt开始请求，共7次，到发送结束共8个rtt

1+7+1.5=9.5个rtt连接断开