

北京航空航天大学

九三年招收
硕士研究生

题单号: 423 ✓

数据结构 试题 (共 7 页)

适用范围: 计算机软件、计算机应用等专业

考生注意:

写算法的语言可以用类 Pascal 语言, 也可以用某一种编程语言, 但不得采用框图形式或者生僻古怪的表示方法。

一、(本题 10 分)

简单而又准确地说明下列术语。

数据的逻辑结构: 数据元素之间的逻辑关系

数据的存储结构: 数据结构在计算机中的映象

算法: 用于完成某个特定课题的指令的有穷集合

算法的时间复杂度: 根据算法编制的程序在计算机上运行时间的度量

算法的空间复杂度: 根据算法编制的程序在计算机中运行时占用存储空间的度量。

二、(本题 10 分) 见 P56 (教材)

说明用一维数组存储一元 n 阶多项式

$$A(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 \quad (a_n \neq 0)$$

的两种方法, 并说明当给定一个由 m 个系数非零项组成的 n 阶多项式, 根据什么条件在这两种方法中选择其中最合适的一种。现给定一个多项式为 $A(x) = 4x^5 + 10x^3 - 9x + 5$, 请从这两种方法中选择最合适的一种方法, 并写出该多项式的这种存储数组。

n	a_n	a_{n-1}	a_1	a_0
-----	-------	-----------	-----	-----	-------	-------

三、(本题 10 分)

已知广义表 $L = (a, (b, (c, (d, e), f), g))$, 请画出该广义表的链式存储结构。该结构中链结点设计成

flag	info	link
------	------	------

, 若该结点

原子元素，则 $\text{flag}=0$ ，并且 info 域存放该元素；若该结点为一子表，则 $\text{flag}=1$ ，并且 info 域为指向子表第一个元素的指针； link 为指向同一层下一个结点的指针，当不存在下一个结点时，该指针为空（用 \wedge 表示）。

四、(本题 10 分)

已知某树林转换成二叉树以后所对应的顺序存储结构为 FT（见下图），请画出该树林。

FT:

A	B	H	C	E	I	J	D	F	G
---	---	---	---	---	---	---	---	---	---

五、(本题 10 分)

在存储管理系统中把存储块设计成图五.1 所示，其中， llink 域和 rlink 域分别为左、右指针域； TAG 为标志域， $\text{TAG}=0$ 表示该块为可用的自由块， $\text{TAG}=1$ 表示该块为已分配块； SIZE 域给出该块的大小； uplink 域指出本块首地址；可利用

llink	TAG	SIZE	rlink
TAG	uplink		

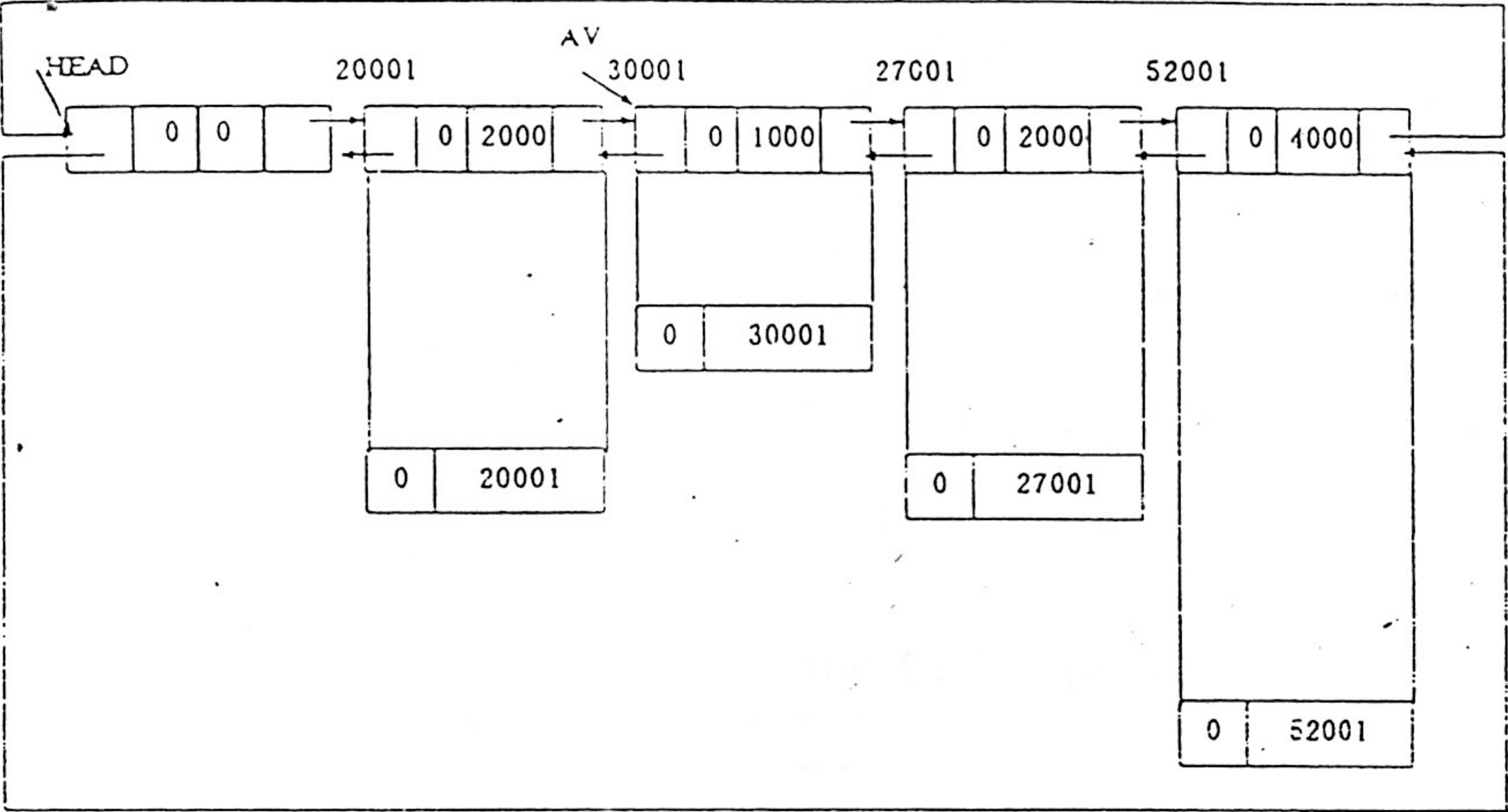
图五.1

用区所有自由块组成双向循环链表， AV 为分配算法中进行分配时搜索可用表的起始位置指针。假设当前可利用区的状态如图五.2 所示（本题图中所有数字均为十进制数字），现给出释放算法（即回收存储块的算法） $\text{FREE}(\text{AV})$ 如后。

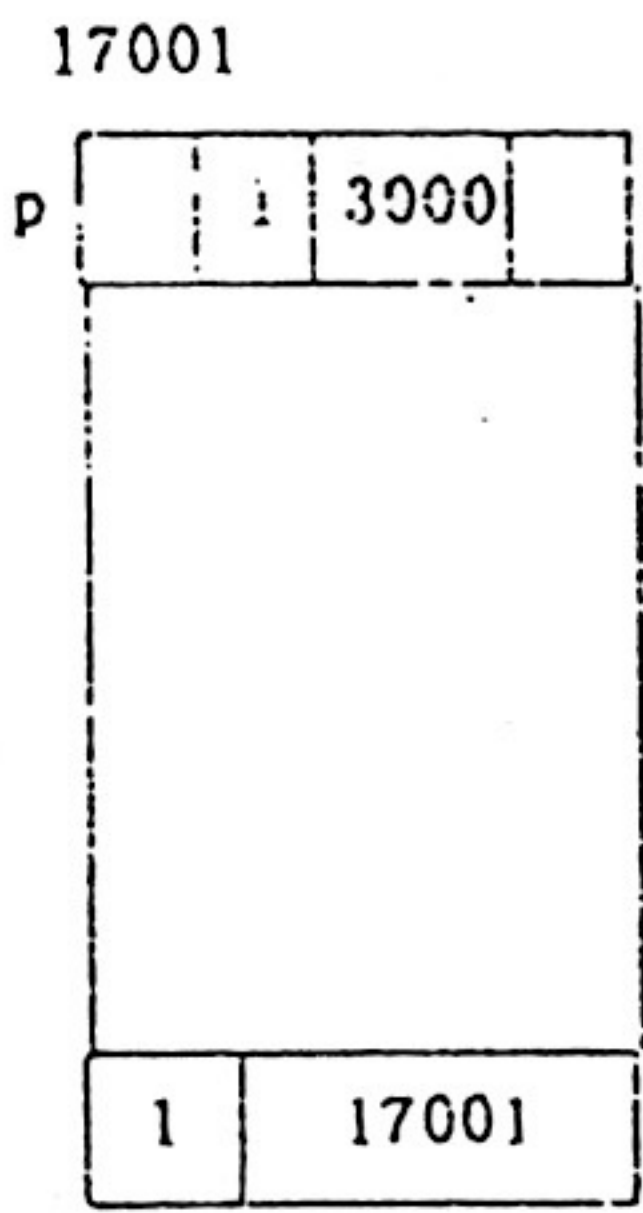
1. 请根据该算法画出回收图五.3 所示的存储块以后的可利用区的状态。

2. 接着根据该算法画出回收图五.4 所示的存储块以后的可利用

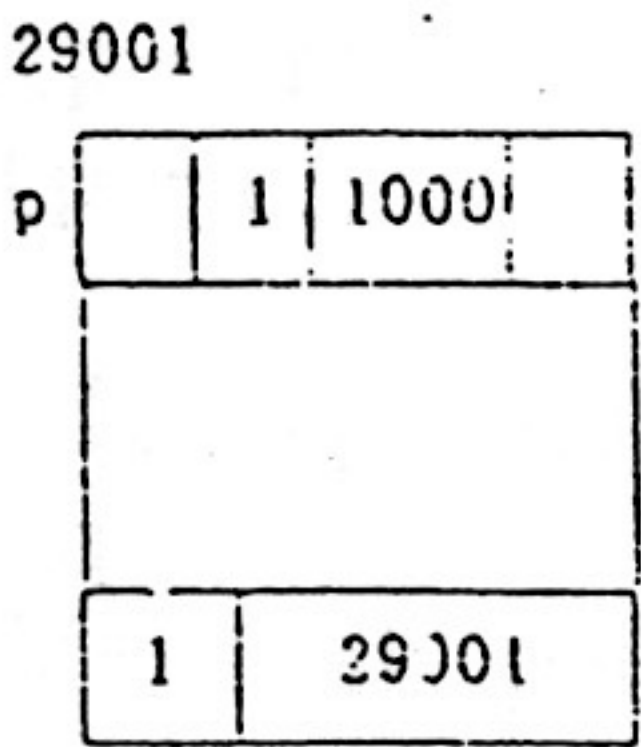
区的状态。



图五.2



图五.3



图五.4

procedure FREE(AV)

$n \leftarrow \text{SIZE}(p)$

case

: (TAG(p-1)=1) and (TAG(p+n)=1):

TAG(p) ← TAG(p+n-1) ← 0

uplink(p+n-1) ← p

llink(p) ← AV

rlink(p) ← rlink(AV)

llink(rlink(p)) ← p

rlink(AV) ← p

: (TAG(p+n)=1) and (TAG(p-1)=0):

q ← uplink(p-1)

SIZE(q) ← SIZE(q) + n

uplink(p+n-1) ← q

TAG(p+n-1) ← 0

: (TAG(p+n)=0) and (TAG(p-1)=1):

rlink(llink(p+n)) ← p

llink(rlink(p+n)) ← p

llink(p) ← llink(p+n)

rlink(p) ← rlink(p+n)

SIZE(p) ← n + SIZE(p+n)

uplink(p+SIZE(p)-1) ← p

TAG(p) ← 0

if AV = p+n then AV ← p

: else:

rlink(llink(p+n)) ← rlink(p+n)

llink(rlink(p+n)) ← llink(p+n)

q ← uplink(p-1)

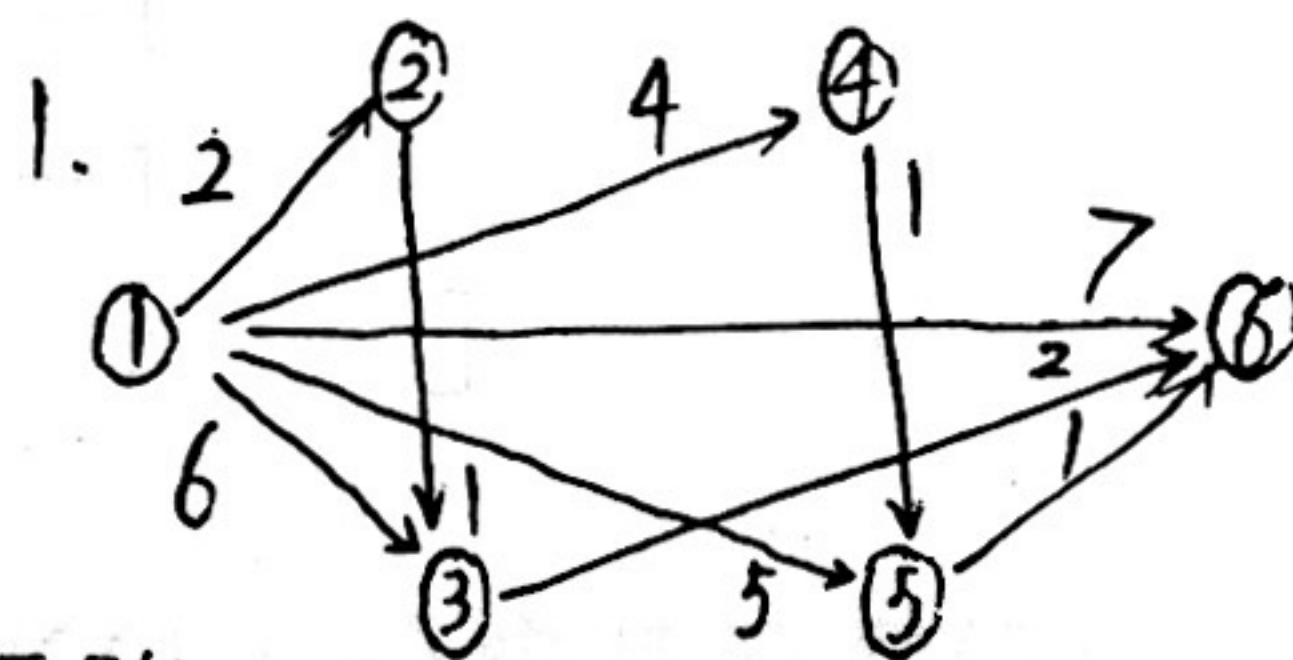
SIZE(q) ← SIZE(q) + n + SIZE(p+n)

uplink(q+SIZE(q)-1) ← q

if AV = p+n then AV ← llink(p+n)

end

end



六 (本题 10 分)

已知 AOE 网采用邻接矩阵

WE(1:6, 1:6) 表示。

2. 以顶点表示事件，以有向边表示活动，边上的权

1. 请画出该 AOE 网；值表示活动的开销的带权有向图。

2. 说明什么是 AOE 网；3. 从源点到终点的路径中具有最大路径长度

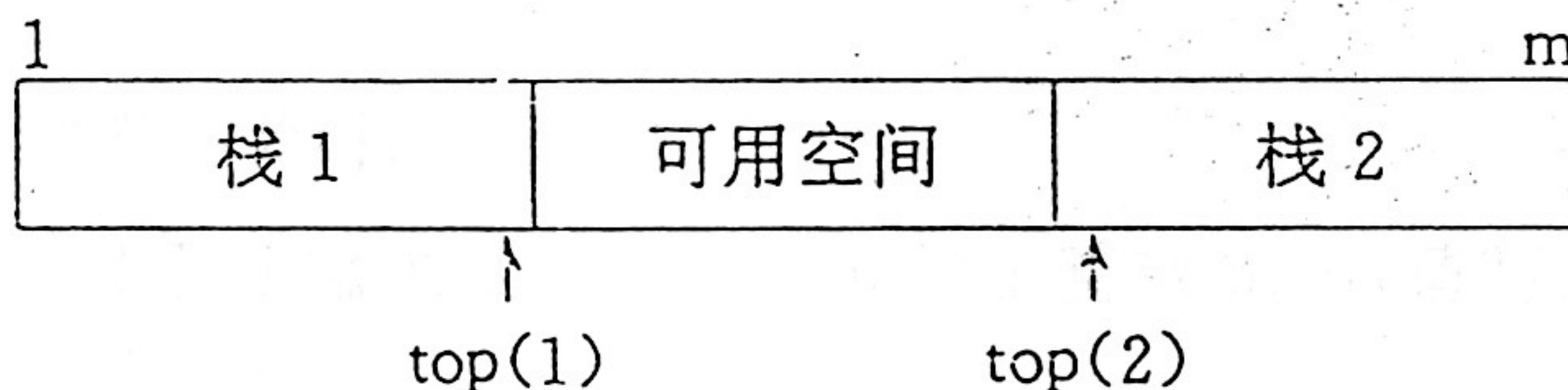
3. 说明什么是关键路径；的路径。

4. 指出本题给出的 AOE 网中的关键路径。①③⑥

$$WE = \begin{bmatrix} 0 & 2 & 6 & 4 & 5 & 7 \\ \infty & 0 & 1 & \infty & \infty & \infty \\ \infty & \infty & 0 & \infty & \infty & 2 \\ \infty & \infty & \infty & 0 & 1 & \infty \\ \infty & \infty & \infty & \infty & 0 & 1 \\ \infty & \infty & \infty & \infty & \infty & 0 \end{bmatrix}$$

七、(本题共 10 分, 每小题 5 分)

1. 已知两个堆栈共享连续空间 $STACK(1:m)$, 各栈的栈顶元素的位置分别由 $top(1)$ 与 $top(2)$ 指出, (见图七)。下面给出对其中第 i 栈 ($i=1, 2$) 插入一个数据信息为 $item$ 的新元素的算法 $PUSH$, 该算法只有当整个空间 $STACK(1:m)$ 全部占用时作插入操作才产生溢出。请在算法的空白处(矩形框内)填上必要的内容。P62(教材)



图七

```

procedure PUSH(STACK, m, top, i, item)
    if top(i) = m then call STACK_FULL
    ("OVERFLOW")
    else
        [ if i=1 then top(1) ← top(1) + 1
          else top(2) ← top(2) - 1
          STACK(top(i)) ← item ]
    end
    
```

2. 已知对具有 n 个元素组成的序列 $K(1:n)$ 按其元素值从小到大进行插入排序的算法如下:

```

procedure INSERT_SORT(K, n)
    for i ← 2 to n do
        temp ← K(i)
    
```



```

j ← i - 1
while (temp < K(j) and j ≥ 1) do
    K(j + 1) ← K(j)
    j ← j - 1
end
K(j + 1) ← temp
end

```

end

(1) 当序列在排序前已经有序, 则执行该算法过程中进行的比较次数是多少?

(2) 当序列在排序前为逆序, 则执行该算法过程中进行的比较次数是多少?

(3) 该算法的时间复杂度为多少?

八、(本题 10 分) P68 (教材)

对于任意大于或者等于 0 的正整数 m, n , 请写出计算下列函数的非递归算法。

$$f(m, n) = \begin{cases} m + n + 1 & \text{当 } m * n = 0 \text{ 时} \\ f(m - 1, f(m, n - 1)) & \text{当 } m * n \neq 0 \text{ 时} \end{cases}$$

九、(本题 10 分)

已知二叉树采用二叉链表存储结构, 根结点的存储地址为 T , 结点的构造为 $[lchild | data | rchild]$, 其中, $data$ 域存放某结点的数据信息, $lchild$ 与 $rchild$ 域分别为指向该结点左、右孩子结点的指针 (当孩子结点不存在时, 相应指针域值为空, 用 nil 表示)。请写一非递归算法, 把该二叉树所有结点的数据信息按层次从上到下, 每一层从左到右的顺序打印成一个线性序列。

十、(本题 10 分)

已知具有 n 个顶点的图已采用邻接矩阵存储方式, 邻接矩阵为 A

n 个顶点，顶点的数据信息分别存储在数组 $V(1:n)$ 中。请写一算法，生成该图的邻接表存储方式。邻接表中顶点数组中结点的构造为 vertex|link，其中，vertex 域存放某顶点的数据信息，link 域为指向依附于该顶点的第一条边所对应的边结点的指针；边结点的构造为 adjvex|next，其中，adjvex 域存放依附于该顶点的一条边的另一端顶点（用该顶点在顶点数组中的序号（下标）表示），next 域为指向下一条邻接边所对应的边结点的指针。