

2021-2022 第二学期《数据结构与算法》大作业练习

一、简答题（每小题 12 分，共 48 分）

1. 简述数据的逻辑结构有哪几种类型，各自的特点是什么？以线性表为例，简述其存储结构有哪几种，各自的特点并举例说明不同存储结构适用的场景？

2. 已知 7 个字符{A, B, C, D, E, F, G}在电文中出现的频率依次是{17, 12, 3, 11, 20, 7, 30}，请回答下列问题。

(1) 画出相应的哈夫曼树，构造过程中，假设约定左子树权值小于右子树权值。

(2) 写出每个字符对应的哈夫曼编码（左侧分支编码为‘0’，右侧分支编码为‘1’）。

(3) 电文编码总长度 WPL 值？

3. 对于关键字序列（18, 12, 22, 10, 17, 29, 21, 28, 33, 49），设定散列地址空间为[0..10]，散列函数为 $H(key)=key\%11$ ，请回答下列问题。

(1) 采用线性探查解决冲突，填写构造的散列表。

0	1	2	3	4	5	6	7	8	9	10

(2) 计算等概率下查找成功的平均查找长度 ASL。

4. 对于关键字序列（29, 35, 23, 37, 18, 19, 27, 47, 39, 56），试按照下列格式，写出利用堆排序方法将其重新排列为非递减顺序的前 3 个堆的状态。

初始状态

0	1	2	3	4	5	6	7	8	9
29	35	23	37	18	19	27	47	39	56

第 1 个堆（初始堆）

0	1	2	3	4	5	6	7	8	9

第 2 个堆

0	1	2	3	4	5	6	7	8	9

第 3 个堆

0	1	2	3	4	5	6	7	8	9

二、算法阅读题（每小题 6 分，共 18 分）

阅读如下算法，回答问题。

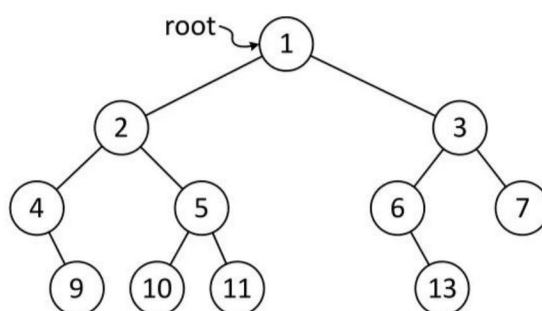
1	struct BinTreeNode { //二叉树节点的定义
2	int value;
3	BinTreeNode* leftchild;
4	BinTreeNode* rightchild;
5	};
6	void Algorithm(BinTreeNode* root) {
7	//定义 2 个栈，栈中数据元素为 BinTreeNode 指针类型
8	std::stack<BinTreeNode*> stk1, stk2;
9	bool flag = true;
10	if (root == NULL) { return; }
11	stk1.push(root);
12	BinTreeNode* temp = 0;
13	while (!(stk1.empty() && stk2.empty())) {
14	if (flag) {
15	temp = stk1.top(); stk1.pop();
16	std::cout << "[" << temp->value << "]";
17	if (temp->leftchild != NULL)
18	stk2.push(temp->leftchild);
19	if (temp->rightchild != NULL)
20	stk2.push(temp->rightchild);

资料由公会员【王士旺】收集整理并免费分享

21	if (stk1.empty())
22	flag = !flag;
23	}
24	else {
25	temp = stk2.top(); stk2.pop();
26	std::cout << "[" << temp->value << "]";
27	if (temp->rightchild != NULL)
28	stk1.push(temp->rightchild);
29	if (temp->leftchild != NULL)
30	stk1.push(temp->leftchild);
31	if (stk2.empty())
32	flag = !flag;
33	}
34	}
35	}

1、请简述该算法 Algorithm 的功能。

2、对于如下给定的二叉树 T，root 指向根节点，请写出运行该算法后的输出结果。



3、该算法中 flag 变量的作用是什么？

三、算法设计题（共 34 分）

1.（算法设计，14 分）给定一个图 $G=(V, E)$ ，以及指定图中的一个顶点 V_i （顶点编号为 i ），设计算法，编写算法，判断是否该图中顶点 V_i 到图中任意一顶点 V_j 均存在简单路径，且路径长度小于 n ，如果是返回 true，否则返回 false。该算法的设计与存储结构无关，既适用于相邻矩阵又适用于邻接表。算法的设计中可以使用自定义的辅助数据结构。

图 G 的基类声明如下，其中的方法可以在算法中直接使用：

```
class Graph{
```

```
public:
```

```
int* Mark; //指向标记图的顶点是否被访问过的数组的指针
```

```
Graph(int vertex_num); //构造函数，参数为图中顶点数
```

资料中 并免费分享

```

virtual ~Graph();          //析构函数
int GetVerticesNum();      //获得图中顶点的数量
int GetEdgeNum();          //获得图中边的数量
void ClearVisitState();    //清除图中所有顶点的访问标记
void Visit(int vertex);    //访问编号为 vertex 的顶点信息
Edge GetFirstEdge(int vertex); //获得依附于编号为 vertex 的顶点的第一条边
Edge GetNextEdge(Edge preedge); //获得与 preedge 具有相同顶点的下一条边
bool IsEdge(Edge edge);    //判断给定 edge 是否为图中的一条边
};

Edge 的声明如下:
class Edge{
public:
    int from, to, weight;
    Edge(); //构造函数
    Edge(int f, int t, int w); //带参的构造函数
};

```

算法原型: `bool HasSimplePath (Graph& G, int i, int n);`
// G 为给定的图, i 为指定的顶点编号, n 为路径长度限制

2. (数据结构设计, 20 分) 域名(Domain Name)是因特网上一个服务器或一个网络系统的名字, 如: `www.sina.com.cn`, 它是互网络上识别和定位计算机的层次结构式的字符标识。域名由若干英文单词组成, 中间由“.”分隔, 从右到左依次为顶级域名段、次高域名段等。如:
cn: 顶级域名,代表某个国家、地区或大型机构; **com:** 网络名,也叫二级域名, 代表部门系统或隶属一级区域的下级机构; **sina:** 三级域名,即组织机构名,是本系统、单位或院所的软硬件平台的名称。每级域名由其上一级的域名管理机构管理, 顶级域名由因特网的有关机构管理, 只要每个管理机构保证其下一层节点名不冲突, 即可保证域名全局唯一性。设计域名管理系统, 进行问题分析, 回答下列问题:

- (1) 解决此问题需要已知哪些数据? (文字描述即可)
- (2) 这些数据采用什么数据结构组织? (文字描述即可)
- (3) 写出主要数据结构的抽象数据类型定义。