**南京师范大学中北学院**

**《数据结构》**

**实**

**验**

**报**

**告**

**姓 名: 钟玮**

**学 号: 85213241**

**班 级: 85213241**

**日 期: 5.6**

**哈夫曼树及其应用**

1. **实验目的**

**1、掌握哈弗曼树的定义和哈弗曼编码的作用；**

**2、掌握哈弗曼树的构造方法以及如何根据哈弗曼树得到哈弗曼编码。**

1. **实验内容**

**1、给定n个字符的权，设计哈弗曼编码。**

1. **实验主要代码**

**可以截图并做简要解释，如：**

1. **构造哈夫曼树的过程中调用的Select函数**

**void Select(HuffmanTree &HT, int i, int &s1, int &s2)**

**{**

**int t;**

**s1 = min(HT, i);**

**s2 = min(HT, i);**

**if (s1 > s2)**

**{**

**t = s1;**

**s1 = s2;**

**s2 = t;**

**}**

**}**

1. **构造赫夫曼树HT，求出n个字符的哈夫曼编码HC**

**void HuffmanCoding(HuffmanTree &HT, HuffmanCode &HC, int \*w, int n)**

**{**

**int m, i, j;**

**int s1, s2;**

**int start;**

**char \*cd;**

**HuffmanTree p;**

**if (n <= 1)**

**return;**

**m = 2 \* n - 1;**

**HT = (HuffmanTree)malloc(sizeof(HTNode) \* (m + 1));**

**if (!HT)**

**exit(OVERFLOW);**

**for (p = HT + 1, i = 1; i <= n; i++, w++, p++)**

**{**

**p->weight = \*w;**

**p->parent = 0;**

**p->lchild = 0;**

**p->rchild = 0;**

**}**

**for (; i <= m; ++i, ++p)**

**{**

**p->weight = 0;**

**p->parent = 0;**

**p->lchild = 0;**

**p->rchild = 0;**

**}**

**for (i = n + 1; i <= m; i++)**

**{**

**Select(HT, i - 1, s1, s2);**

**HT[s1].parent = HT[s2].parent = i;**

**HT[i].lchild = s1;**

**HT[i].rchild = s2;**

**HT[i].weight = HT[s1].weight + HT[s2].weight;**

**}**

**HC = (HuffmanCode)malloc((n + 1) \* sizeof(char \*));**

**cd = (char \*)malloc(n \* sizeof(char));**

**cd[n - 1] = '\0';**

**for (i = 1; i <= n; i++)**

**{**

**start = n - 1;**

**for (j = i; j < m; j = HT[j].parent)**

**if (HT[HT[j].parent].lchild == j)**

**cd[--start] = '0';**

**else**

**cd[--start] = '1';**

**HC[i] = (char \*)malloc((n - start) \* sizeof(char));**

**strcpy(HC[i], &cd[start]);**

**}**

**free(cd);**

**}**

1. **main函数**

**int main()**

**{**

**HuffmanTree HT;**

**HuffmanCode HC;**

**int w[8] = {14, 32, 7, 9, 14, 29, 3, 11};**

**HuffmanCoding(HT, HC, w, 8);**

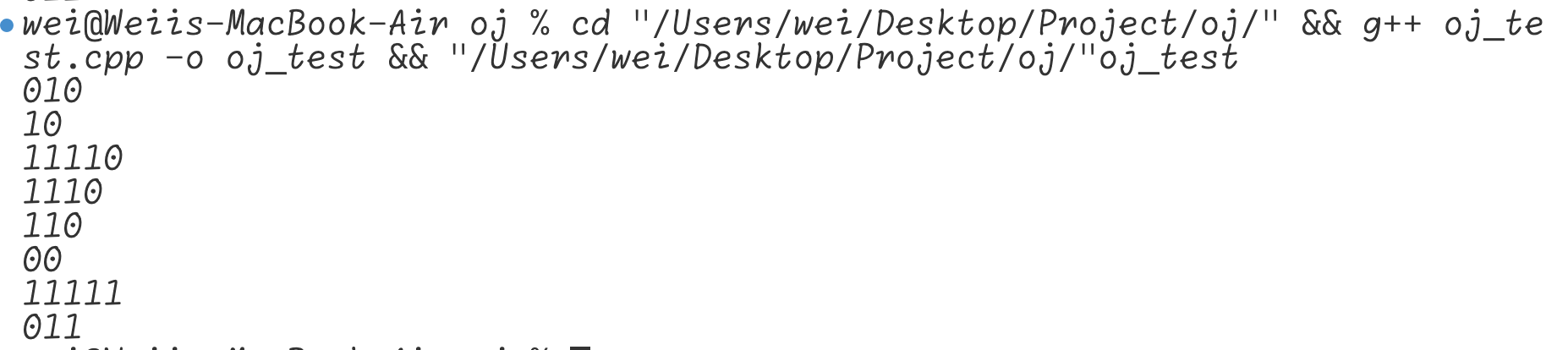
**for (int i = 1; i <= 8; i++)**

**puts(HC[i]);**

**}**

1. **实验结果**

**实验运行结果的截图，如main函数运行结果截图：**

****