5.10:

int sum = 0;

template <typename E>

int inorder(BinNode<E>\* root)

{

if (root == NULL) return 0;

sum = inorder(root->left()) + sum;

sum = root->value + sum;

sum = inorder(root->right()) + sum;

return sum;

}

5.19

template <typename E>

int smallcount(BinNode<E>\* root, Key K)

{

if (root == NULL) return 0;

if (KEComp.gt(root->value(), K))

return smallcount(root->leftchild(), K);

else

return smallcount(root->leftchild(), K)+smallcount(root->rightchild(), K) + 1;

}

5.25

12 9 10 5 4 1 8 7 3 2

5.26

<a> 6 5 3 4 2 1

<b> 7 4 6 3 2 1

5.28

l,0

k,010

j,0110

i,01110

h,011110

g,0111110

f,01111110

e,011111110

d,0111111110

c,01111111111

a,011111111100

b,011111111101

预期存储1012/209约等于4.84位

5.29

深度为四,叶子数为16的树,平均代码长度为4,此时与固定长度编码的没有差别

5.31

<a>在串里只有z或q的最差情况下需要5n位.

<b>在串里只有o或e的最佳情况下需要2n位.

<c>平均需要270/100=2.7位

5.32

<a>有序顺序表,因为相比有序链表来说,查询较为节省时间

<b>无序链表

<C>BST

<d>无序顺序表,查询节省时间