

1)

Dllist* combinaOrdemados (Dllist* l1, Dllist* l2, int (*comp?) (void*, void*)) {

```

if (l1 != NULL && l2 != NULL) {
    if (l1->first != NULL && l2->first != NULL) {
        Dlnode* cur1 = l1->first;
        Dlnode* cur2 = l2->first;
        Dlnode* temp;
        Dllist* list;
        list = (Dllist*) malloc (size of (Dlnode));
        temp = list->first;
        while (cur1 != NULL) {
            int stat = comp (cur1->data, cur2->data);
            if (stat == 1) {
                temp->data = cur2->data;
                temp = temp->next;
            } else {
                temp->data = cur1->data;
                temp = temp->next;
            }
            cur1 = cur1->next;
        }
        return list;
    }
}

```

return NULL;

}

2.

```

int commons(SList* l1, SList* l2, int comp)(void*, void*) {
    SList* cur1, *cur2;
    int cont = 0;
    if(l1 != NULL & l2 != NULL) {
        if(l1->first != NULL & l2->first != NULL) {
            cur1 = l1->first;
            cur2 = l2->first;
            stat = comp(cur1->data, cur2->data);
            while(cur1->next != l1->first) {
                while(cur2->next != l2->first) {
                    if(stat == 0) {
                        cont++;
                    }
                    cur2 = cur2->next;
                    stat = comp(cur1->data, cur2->data);
                }
                cur1 = cur1->next;
                cur2 = l2->first;
            }
            return cont;
        }
    }
    return 0;
}

```

3

```
int abCalcNumNosGraus (TNode *t) {
```

```
    int n=0, nl=0, nr=0;
```

```
    if (t != NULL) {
```

```
        if (t->left == NULL && t->right != NULL || t->left != NULL  
            && t->right == NULL) {
```

```
            n++;
```

```
        }
```

```
        nl = abCalcNumNosGraus (t->left);
```

```
        nr = abCalcNumNosGraus (t->right);
```

```
        return n + nl + nr;
```

```
    }
```

```
    return 0;
```

```
}
```

4

•

Pre-ordem

40, 5, 31, 27, 38, 36, 85, 60, 47, 52, 71, 65, 75

Pos-ordem

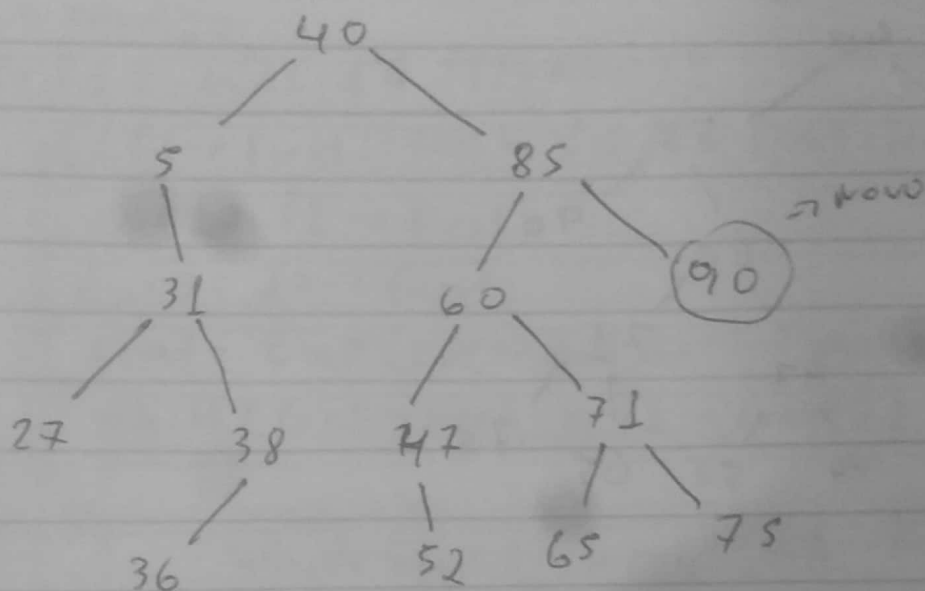
27, 36, 38, 31, 5, 52, 47, 65, 75, 71, 60, 85, 40

Simétrica

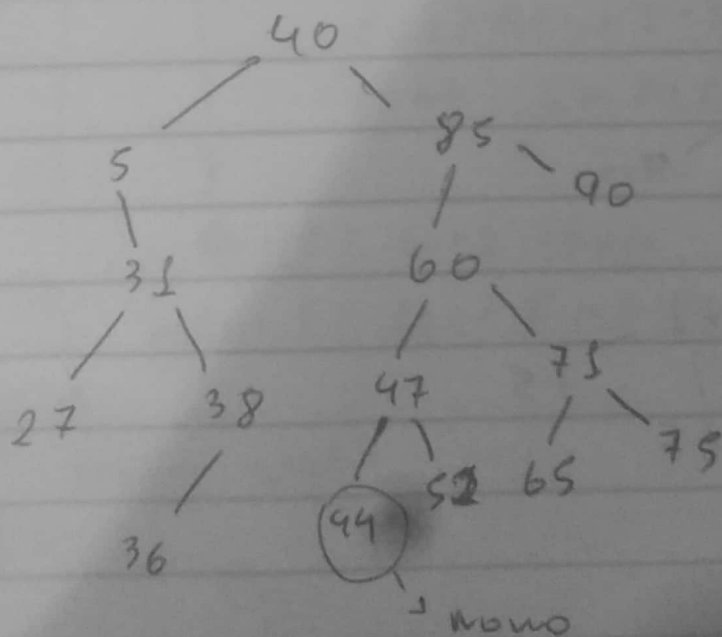
~~40, 5, 31, 27, 38, 36, 85, 60, 47, 52, 71, 65, 75~~

27, 31, 36, 38, 5, 40, 52, 47, 60, 65, 71, 75, 85

4. adição
a) 90



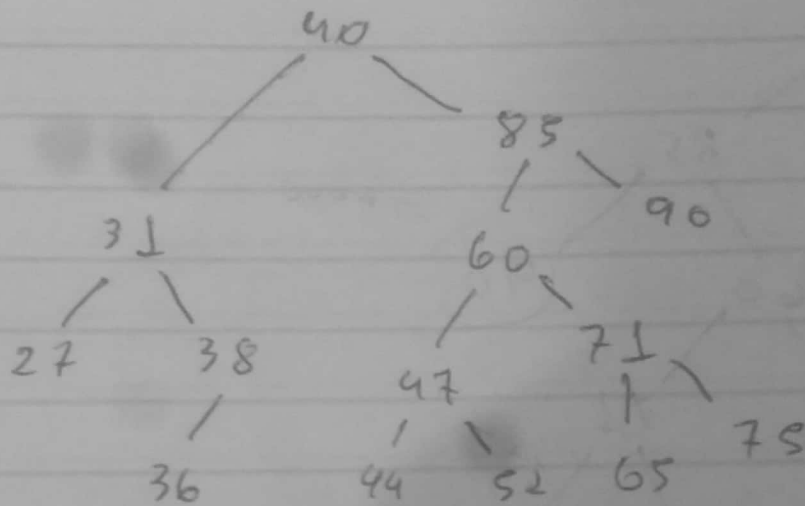
b) 44



QUA | SEX | TER | QUA | QUA | SEX | SÁB

4. remoção

d) 5



e) 60

