

Algoritmo de Inserção

```
void insere_no_avl(AVL *A, float x)
{
    int f=0;
    A->raiz = insere_avl(A->raiz,x,&f);
}

NO_AVL *insere_avl(NO_AVL *raiz, float x, int *flag)
{
    if (raiz)
    {
        if (raiz->elem > x)
        {
            raiz->fesq = insere_avl(raiz->fesq,x,flag);
            raiz->fesq->pai = raiz;
            if (*flag)
            {
                switch(raiz->bal){
                    case -1: raiz->bal = 0;
                        *flag = 0;
                        break;
                    case 0: raiz->bal = 1;
                        break;
                    case 1: if (raiz->fesq->bal == 1)
                        {
                            raiz = rotacao_LL(raiz);
                            raiz->bal = 0;
                            raiz->fdir->bal = 0;
                        }
                        else
                        {
                            raiz = rotacao_LR(raiz);
                            if (raiz->bal == 1)
                            {
                                raiz->fesq->bal = 0;
                                raiz->fdir->bal = -1;
                            }
                            else
                            {
                                raiz->fesq->bal = 1;
                                raiz->fdir->bal = 0;
                            }
                            raiz->bal = 0;
                        }
                        *flag = 0;
                        break;
                }
            }
        }
        else
        {
            raiz->fdir = insere_avl(raiz->fdir,x,flag);
            raiz->fdir->pai = raiz;
            if (*flag)
            {
                switch(raiz->bal){
                    case -1:
                        break;
                    case 0: raiz->bal = -1;
                        break;
                    case 1:
                        break;
                }
            }
        }
    }
    else
    {
        raiz = (NO *) malloc(sizeof(NO));
        raiz->fesq = raiz->fdir = raiz->pai = NULL;
        raiz->elem = x;
        raiz->bal = 0;
        *flag = 1;
    }
    return(raiz);
}
```

/* Exercício: As rotações RR e RL nao foram implementadas, atualize o código a fim de incorporar tais rotações */