

```
#include <stdlib.h>
#include <stdio.h>
typedef struct arvgeral{
    int info;
    struct arvgeral *filho;
    struct arvgeral *irmao;
}ArvGeral;
ArvGeral* Ler_arvore_arq(FILE* arq){
    char c;
    int x;
    fscanf(arq,"%c",&c);
    fscanf(arq,"%d",&x);
    if(x==-1){
        fscanf(arq,"%c",&c);
        return NULL;
    }
    else{
        ArvGeral *no=(ArvGeral*)malloc(sizeof(ArvGeral));
        no->info=x;
        no->filho=Ler_arvore_arq(arq);
        no->irmao=Ler_arvore_arq(arq);
        fscanf(arq,"%c",&c);
        return no;
    }
}

void imprimir_preordem(ArvGeral *a){
    if(a==NULL)
        printf("(-1)");
    else{
        printf("(");
        printf("%d",a->info);
        imprimir_preordem(a->filho);
        imprimir_preordem(a->irmao);
        printf(")");
    }
}

int altura(ArvGeral* a){
    if(a==NULL)
        return 0;
    else{
        int af,ai;
        af=altura(a->filho)+1;
        ai=altura(a->irmao);
        if(af>ai)
            return af;
        else
            return ai;
    }
}
```

```

void imprimir_nivel(ArvGeral *a, int nivel, int cont){
    if(a!=NULL){
        if(cont==nivel){
            printf ("%d ",a->info);
            imprimir_nivel(a->irmao,nivel,cont);
        }
        else{
            imprimir_nivel(a->filho,nivel,cont+1);
            imprimir_nivel(a->irmao,nivel,cont);
        }
    }
}

void imprimir_largura(ArvGeral *a, int nivel, int cont, int altura){
    if (a!=NULL)
        while(nivel<altura){
            imprimir_nivel(a,nivel,cont);
            nivel++;
        }
}

int conta_nos(ArvGeral *a, int nivel, int cont){
    if(a==NULL)
        return 0;
    else{
        if(cont==nivel)
            return 1+conta_nos(a->irmao,nivel,cont);
        else
            return conta_nos(a->filho,nivel,cont+1)+conta_nos(a->irmao,nivel,cont);
    }
}

void imprimir_folhas(ArvGeral *a){
    if(a!=NULL){
        if (a->filho==NULL){
            printf ("%d ",a->info);
            imprimir_folhas(a->irmao);
        }
        else{
            imprimir_folhas(a->filho);
            imprimir_folhas(a->irmao);
        }
    }
}

ArvGeral* remover_arvore(ArvGeral *a){
    if(a!=NULL){
        remover_arvore(a->filho);
        remover_arvore(a->irmao);
        free(a);
    }
    return NULL;
}

int main (void){
    FILE* arq;
    arq=fopen("arvore.txt","rt");

```

```

if(arq==NULL){
    printf("Erro ao abrir o arquivo");
    exit(1);
}
ArvGeral *a;
int e=1,h=0,cont=0,i=0,nivel=0;
while (e!=7){
    system("cls");
    printf ("Digite a opcao desejada\n\n");
    printf ("1 - Ler a arvore de um arquivo\n");
    printf ("2 - Imprimir em pre-ordem\n");
    printf ("3 - Imprimir em largura\n");
    printf ("4 - Calcular a altura\n");
    printf ("5 - Contar o numero de nohs de um determinado nivel\n");
    printf ("6 - Imprimir folhas\n");
    printf ("7 - Sair\n");
    scanf ("%d",&e);
    system("cls");
    switch (e){
        case 1:{
            a=Ler_arvore_arq(arq);
            printf ("A arvore foi lida com sucesso\n");
            system("pause");
            break;
        }
        case 2:{
            printf ("Impressao em pre-ordem\n\n");
            imprimir_preordem(a);
            printf (" \n");
            system("pause");
            break;
        }
        case 3:{
            printf ("Impressao em Largura\n\n");
            h=altura(a);
            nivel=0;
            imprimir_largura(a,nivel,cont,h);
            printf (" \n");
            system("pause");
            break;
        }
        case 4:{
            h=altura(a);
            printf ("A altura da arvore eh %d\n",h);
            printf (" \n");
            system("pause");
            break;
        }
        case 5:{
            printf ("Impressao do numero de nohs de um determinado nivel\n\n");
            printf ("Digite o nivel desejado\n");
            scanf ("%d",&nivel);
            h=altura(a);

```

```

        if (nivel<h){
            i=conta_nos(a,nivel,cont);
            printf ("O numero de nos no nivel %d eh %d\n",nivel,i);
        }
        else
            printf("Nivel nao existente");
        printf (" \n");
        system("pause");
        break;
    }
    case 6:{
        printf ("A(s) folha(s) da arvore e(sao): \n");
        imprimir_folhas(a);
        printf (" \n");
        system("pause");
        break;
    }
    case 7: {
        a=remover_arvore(a);
        break;
    }
}
}
fclose(arq);
return 0;
}

```