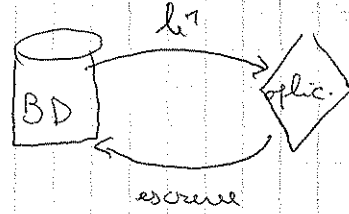
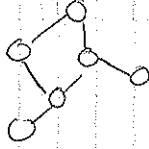
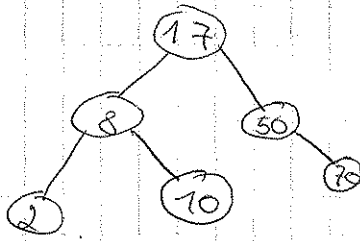


Representação intermédia



Maskell

data ArvBin a = cons ArvBin Null
 | cons ArvBin a
 (ArvBin o) (ArvBin e)



```

typedef struct sArvBin
{
    int valor;
    struct sArvBin * esq, * dir;
} * ArvBin, Nodo ArvBin;
    
```

```

void inorder (ArvBin a)
{
    if (a)
    {
        inorder (a -> esq);
        printf ("i. d", a -> valor);
        inorder (a -> dir);
    }
}
    
```

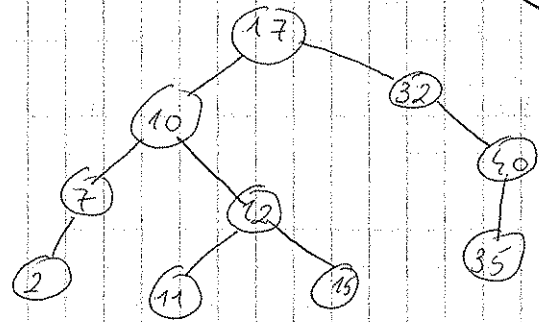
```

void preorder (ArvBin a)
{
    if (a)
    {
        printf ("...");
        preorder (a -> esq);
        preorder (a -> dir);
    }
}
    
```

```

void inorder (ArvBin a)
{
    if (a)
    {
        inorder (a -> esq);
        inorder (a -> dir);
        printf (...);
    }
}

```



inorder (a); 2, 7, 10, 11, 12, 15, 17, 32, 35, 40

gruover em
filho A preorder (a); 17, 10, 7, 2, 12, 11, 15, 32, 40, 35

postorder (a); 2, 7, 11, 15, 12, 10, 35, 40, 32, 17

```

int contaElem (ArvBin a)
{
    if (!a)
        return 0;
    else
        return 1 + contaElem (a -> esq) + contaElem (a -> dir);
}

```

```

ArvBin constrArvBin (ArvBin a, int m)
{

```

```

    ArvBin auxc;

```

```

    if (!a)
    {

```

```

        auxc = (ArvBin) malloc (sizeof (Nodo ArvBin));

```

```

        auxc -> valor = m;

```

```

        auxc -> esq = a;

```

```

        auxc -> dir = a;
    }
    return auxc;

```

```

    else
    {

```

```

        if (a -> valor > m)
        {

```

```

            a -> esq = constrArvBin (a -> esq, m);

```

```

            return a;
        }
    }

```

```

    else
    {

```

```

        a -> dir = constrArvBin (a -> dir, m);

```

```

        return a;
    }
}

```