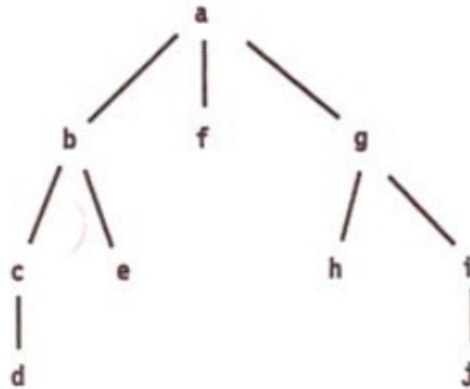


Q1 (3,5 PONTOS):

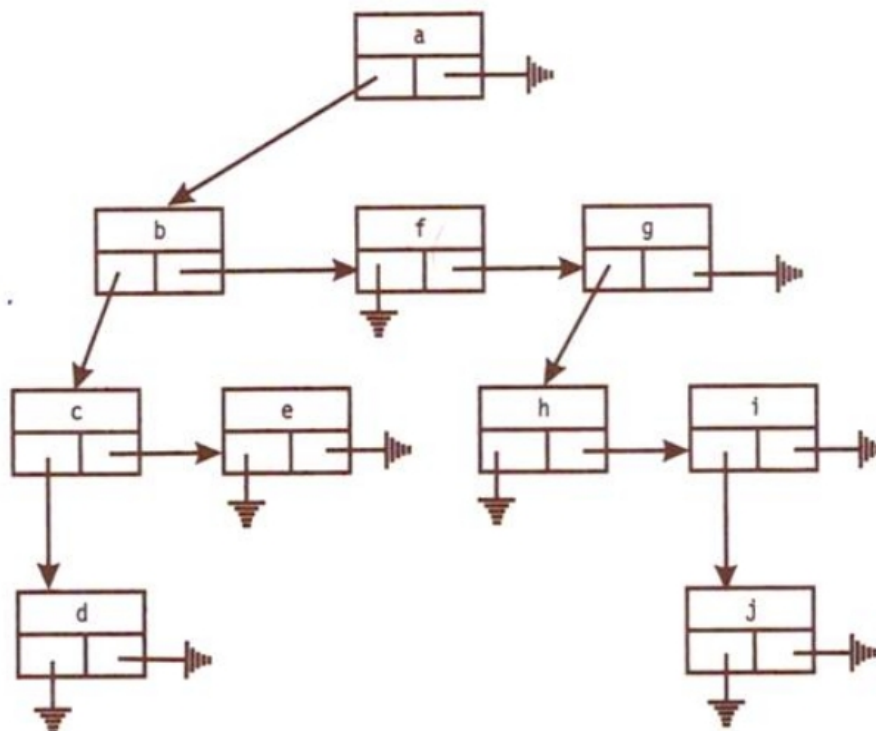
Sabendo-se que uma árvore com o número variável de filhos pode ser representada como a seguinte árvore binária:

```
typedef struct arv_var{
    int info;
    struct arv_var *prim_filho, *prox_irmao;
}TAV;
```

Para exemplificar, a árvore ternária (maior número de filhos de um nó) abaixo:



Pode ser representada por:



Escreva uma função que retorne o maior número de filhos que uma árvore como essa tem. Se ela for binária, isto é, $n = 2$, sua função deve retornar a conversão dessa árvore numa binária tradicional, com os ponteiros **esq** e **dir**. Se o número de filhos for maior que 2, a árvore retornada é **NULL**. O protótipo da função é o que segue: **TAB* av2ab(TAV *a, int *n)**. Alguns exemplos de entradas e saídas são mostradas abaixo:

ENTRADAS	SAÍDAS
Digite a raiz da arvore... 2 Quer continuar inserindo nos na árvore (digite N ou n para parar)? s Digite o pai (que deve existir) e o filho a ser inserido na arvore... 2 3 Quer continuar inserindo nos na árvore (digite N ou n para parar)? s Digite o pai (que deve existir) e o filho a ser inserido na arvore... 2 1 Quer continuar inserindo nos na árvore (digite N ou n para parar)? n 2 1 3	2 1 2 3
Digite a raiz da arvore... 2 Quer continuar inserindo nos na árvore (digite N ou n para parar)? s Digite o pai (que deve existir) e o filho a ser inserido na arvore... 2 4 Quer continuar inserindo nos na árvore (digite N ou n para parar)? s Digite o pai (que deve existir) e o filho a ser inserido na arvore... 2 3 Quer continuar inserindo nos na árvore (digite N ou n para parar)? s Digite o pai (que deve existir) e o filho a ser inserido na arvore... 2 1 Quer continuar inserindo nos na árvore (digite N ou n para parar)? n 2 1 3 4	3