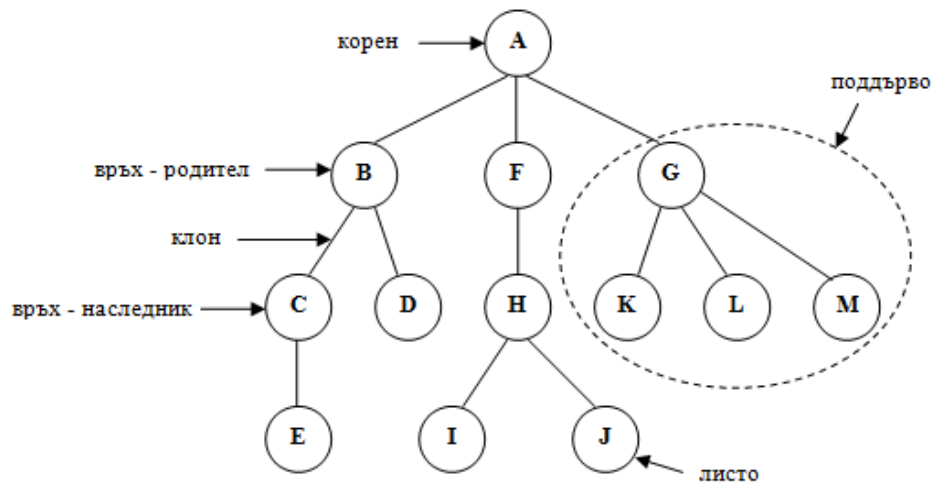


САА – Упражнение 11

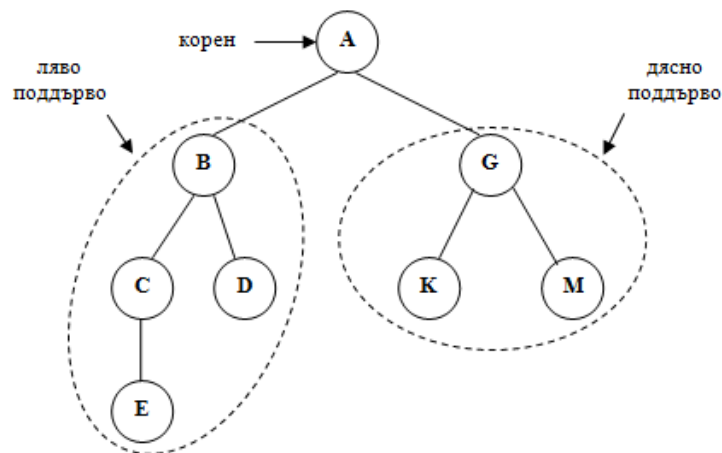
Дърво

- Всеки елемент на дървото ще наричаме **върх**.
- Връзката между родител и негов пряк наследник ще наричаме **клон**.
- Броят на преките наследници на даден върх определя неговата **степен**.
- Когато степенята на един върх е 0, той се нарича **листо**.
- **Път** е последователност от върхове без повторение, като всеки два последователни върха са свързани с клон в дървото.
- **Дължината** на пътя между два върха е равна на броя на клоните, които участват в този път.
- **Ниво** на върх се нарича дължината на пътя от корена до съответния върх.
- Максималното ниво на върх в дървото определя неговата **височина**.



фигура 1

Двоично дърво е това дърво, в което всеки върх може да има максимум два наследника.



фигура 2

Обхождане на двоично дърво

- 1) в прав ред (префиксно обхождане);
- 1) в междинен ред (инфиксно обхождане);
- 1) в обратен ред (постфиксно обхождане).

При **обхождане в прав ред** (префиксно обхождане) на двоично дърво се изпълняват рекурсивно следните три стъпки като се започне от корена:

- 1) Посещава се корена;
- 2) Обхожда се в прав ред лявото поддърво;
- 3) Обхожда се в прав ред дясното поддърво.

Пример: за двоичното дърво на фигура 2, резултатът от обхождането ще бъде следния: A B C E D G K M

При **обхождане в междинен ред** (инфиксно обхождане) на двоично дърво се изпълняват рекурсивно следните три стъпки:

- 1) Обхожда се в междинен ред лявото поддърво;
- 2) Посещава се корена;
- 3) Обхожда се в междинен ред дясното поддърво.

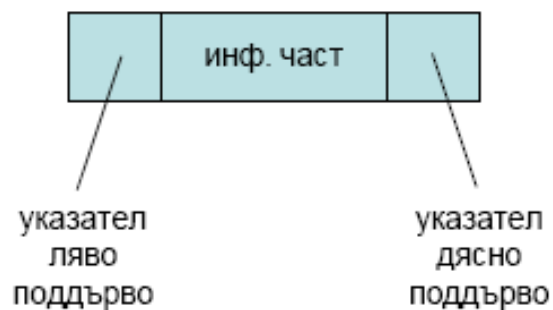
Пример: за двоичното дърво на фигура 2, резултатът от обхождането ще бъде следния: E C B D A K G M

При **обхождане в обратен ред** (постфиксно обхождане) на двоично дърво се изпълняват рекурсивно следните три стъпки:

- 1) Обхожда се в обратен ред лявото поддърво;
- 2) Обхожда се в обратен ред дясното поддърво;
- 3) Посещава се корена.

Пример: за двоичното дърво на фигура 2, резултатът от обхождането ще бъде следния: E C D B K M G A

Динамично представяне на двоично дърво- всеки връх в двоичното дърво има следната структура:



```
struct node
{
    char key;
    struct node * left;
    struct node * right;
};
```

Задачи:

1. Съставете програма, която създава двоично дърво и отпечатва дървото. Използвайте функциите `readtree` за въвеждане на данни в двоичното дърво и `printtree` за разпечатване на дървото.

```
void readtree(struct node ** p, char &x )
{
    char y, z;
    struct node * q1, *q2;

    // създава корена, ако няма такъв до момента

    if( *p == NULL )
    {
        printf( "Vyvedi korena" );
        scanf( "%c", &x );
        (*p)=(struct node *)malloc(sizeof(struct node));
        (*p)->key=x;
    }
    fflush(stdin);

    // въвежда ляв наследник на текущия връх

    printf("\nVyvedi liav naslednik na %c", x);
    z = getchar();
    fflush(stdin);

    // въвежда десен наследник на текущия връх

    printf("\nVyvedi desen naslednik na %c", x);
    y = getchar();

    // създава ляв наследник на текущия връх, ако стойността, въведена за него е
    // различна от '0'

    if (z!='0')
    {
        q1=(struct node *)malloc(sizeof(struct node));
        q1->key=z;
        (*p)->left=q1;
        readtree(&q1,z);
    }
    else
        (*p)->left=NULL;

    // създава десен наследник на текущия връх, ако стойността, въведена за него е
    // различна от '0'

    if (y!='0')
    {
        q2=(struct node *)malloc(sizeof(struct node));
        q2->key=y;
        (*p)->right=q2;
        readtree(&q2,y);
    }
}
```

```

    }
    else
        (*p)->right=NULL;
}

```

```

void printtree(struct node * p, int delta)
{
    int i;
    if (p!=NULL)
    {
        printtree(p->left,delta+1);
        for (i=1;i<=delta; i++)
            printf("\t");
        printf("%c",p->key);
        printf("\n");
        printtree(p->right, delta+1);
    }
}

```

2. Реализирайте програмно всяка от функциите за обхождане на двоично дърво. Разпечатайте резултатите от всяко от обхожданията.
3. Създайте дърво, за което всяко от листата съдържа цифра от 1 до 9, а останалите върхове и корена съдържат операциите +, - и *. Обходете дървото по един от описаните начини, за да получите аритметичния израз във вид, удобен за възприемане от човек (операнд операция операнд). След края на израза изведете знака '=' и отпечатайте стойността на израза.