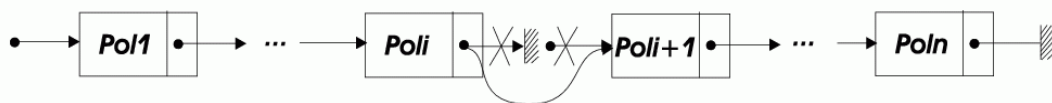


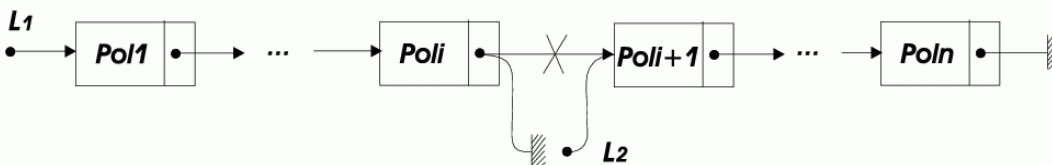
Znenie zadania:

Pridajte podporu operácií CAT a CUT na ÚŠ zoznam. Význam operácií je jasný z obrázkov.
Operácia CAT:

$$L = L1 . L2$$



Operácia CUT:



Prototypy operácií (súbor list.h):

List Cat(List L1, List L2);

List Cut (List L, Position P);

Poznámka: Pre overenie činnosti pridaných operácií doplňte aj modul pre testovanie.

Opis riešenia:

Funkcia Cat slúži na spojenie dvoch zoznamov do jedného. Najprv sa prejde na koniec prvého zoznamu. To sa deje postupným prechádzaním jednotlivých elementov a kontrolovaním ich nasledovníka. Ak je nasledovník rovný NULL, znamená to že daný element je posledný v zozname. Na to aby sa pripojil ďalší zoznam je potrebné ako nasledovníka tohto posledného elementu nastaviť referenciu na prvý element druhého zoznamu, ktorý sa zistí pomocou preddefinovanej funkcie First. Funkcia ako návratovú hodnotu odovzdáva List L1, ktorý už je spojením Listov L1 a L2.

Funkcia Cut slúži na rozdelenie jedného zoznamu na dva a to od elementu na pozícii P. Na vytvorenie druhého zoznamu je potrebné najprv vytvoriť Node. Na ten sa alokuje miesto v pamäti a ako jeho nasledovník sa uloží referencia na nasledovníka z pozície P. Prvý zoznam sa musí ukončiť a tak sa ako nasledovník posledného elementu, ktorý je na pozícii P, nastaví NULL. Funkcia ako návratovú hodnotu odovzdáva List, ktorý vznikol „odseknutím“ od pôvodného Listu.

Pre overenie správnosti funkcií som vo funkcii main() vytvoril 2 Listy. Prvý obsahuje elementy s číselnými hodnotami 0-5 a druhý elementy s číselnými hodnotami 6-10. Následne je volaná funkcia Cat na spojenie týchto zoznamov a funkcia PrintList() na vypísanie vytvoreného zoznamu, ktorý je uložený ako L1. Tento zoznam sa ďalej delí na dva pomocou funkcie Cat. Najprv sa ale udá pozícia na ktorej dôjde k deleniu a to pomocou funkcie Find. Tá je volaná s parametrami elementu za ktorým sa zoznam rozdelí a odkaz na zoznam ktorý sa má deliť. V tomto prípade je element 2 a list L1. Výsledok funkcie Cut je potom zmenený List L1 a vytvorený nový List je uložený ako L2 a oba zoznamy sú vypísané.

Zdrojový kód:

funkcia Cat(List L1, List L2):

```
List Cat( List L1, List L2){
    {
        while( L1->Next != NULL )
            L1 = L1->Next;

        L1->Next = First( L2 );
        return L1;
    }
}
```

funkcia Cut(List L, Position P):

```
List Cut(List L, Position P){
    List Temp;
    Temp = malloc( sizeof( struct Node ) );
    Temp->Next = P->Next;
    P->Next = NULL;

    return Temp;
}
```

funkcia main():

```
int main()
{
    List L1,L2;
    Position P;
    int i;

    L1 = MakeEmpty( NULL );
    P = Header( L1 );
    for( i = 0; i < 6; i++ ){
        Insert( i, L1, P );
        P = Advance( P );
    }
    printf(" List L1: ");
    PrintList( L1 );

    L2 = MakeEmpty( NULL );
    P = Header( L2 );
    for( i = 6; i <= 10; i++ ){
        Insert( i, L1, P );
        P = Advance( P );
    }
    printf(" List L2: ");
    PrintList( L2 );

    Cat( L1, L2);
    printf("\n List L1 (po volani funkcie Cat): ");
    PrintList( L1 );

    P = Find(2, L1);
    printf("\n List L1 (po volani funkcie Cut): ");
    PrintList (Cut(L1, P));

    return 0;
}
```