Latsol ASD - Struct dan Stack

Agnes Destiny

IF-03-01/1203230092

1. Source code :

```c
#include <stdio.h>

typedef struct Node { // Mendefinisikan tipe data struct 'Node'
  struct Node* link; // Pointer yang menunjukkan ke node berikutnya dalam linked list
  char* alphabet; // Pointer ke karakter atau string kedalam node
} Node;

int main() {
  // Deklarasi node-node
  Node l1, l2, l3, l4, l5, l6, l7, l8, l9;

  // Inisialisasi huruf pada masing-masing node
  l1.link = NULL;
  l1.alphabet = "F";

  l2.link = NULL;
  l2.alphabet = "M";

  l3.link = NULL;
  l3.alphabet = "A";

  l4.link = NULL;
  l4.alphabet = "I";

  l5.link = NULL;
  l5.alphabet = "K";

  l6.link = NULL;
  l6.alphabet = "T";

  l7.link = NULL;
  l7.alphabet = "N";

  l8.link = NULL;
  l8.alphabet = "O";

  l9.link = NULL;
  l9.alphabet = "R";

// Menghubungkan node sesuai dengan urutan yang diinginkan
  l7.link = &l1;
  l1.link = &l8;
```

```
  l8.link = &l2;
  l2.link = &l5;
  l5.link = &l3;
  l3.link = &l6;
  l6.link = &l9;
  l9.link = &l4;
  l4.link = &l7;

//Akses data menggunakan 'printf'
  printf("%s", l3.link->link->link->alphabet); // Print huruf 'I'
  printf("%s", l3.link->link->link->link->alphabet); // Print huruf 'N'
  printf("%s", l3.link->link->link->link->link->alphabet); // Print
  huruf 'F'
  printf("%s", l3.link->link->link->link->link->link->alphabet); //
  Print huruf 'O'
  printf("%s", l3.link->link->alphabet); // Print huruf 'R'
  printf("%s", l3.link->link->link->link->link->link->link->alphabet);
  // Print huruf 'M'
  printf("%s", l3.alphabet); // Print huruf 'A'
  printf("%s", l3.link->alphabet); // Print huruf 'T'
  printf("%s", l3.link->link->link->alphabet); // Print huruf 'I'
  printf("%s", l3.link->link->link->link->link->link->link->link-
  >alphabet); // Print huruf 'K'
  printf("%s", l3.alphabet); // Print huruf 'A'

  return 0; // Mengembalikan nilai 0 unutk menunjukkan bahwa program
  telah dijalankan
}
```

Output :

```
PS C:\AlPro\ASD> cd "c:\AlPro\ASD\" ; if ($?) { gcc tugasoth.c -o tugasoth } ; if ($?) { .\tugasoth }
INFORMATIKA
PS C:\AlPro\ASD>
```

2. Source code :

```c
#include <stdio.h> // Mendefinisikan header file unutk fungsi input-output standar

int twoStacks(int maxSum, int a[], int n, int b[], int m) { // Fungsi twoStacks yang mengambil lima argumen. 'maxSum' merupakan batas jumlah maksimum yang diizinkan, 'a[]' array yang mewakili tumpukan A, 'n' ukuran tumpukan A, 'b[]' Array yang mewaakili tumpukan B, 'm' merupakan ukuran dari tumpukan B
    int sum = 0, count = 0, temp = 0, i = 0, j = 0; // Untuk mendeklarasikan beberapa variabel lokal yang akan digunakan dalam fungsi

    while (i < n && sum + a[i] <= maxSum) { // Melakukan iterasi melalui tumpukan A sampai batas maxSum tercapai
        sum += a[i++];
    }
    count = i; // Menyimpan jumlah elemen yang telah diambil dari tumpukan A

    while (j < m && i >= 0) { // Iterasi melalui tumpukan B, sementara memeriksa apakah jumlah dari kedua tumpukan tidak melebihi maxSum
        sum += b[j++];
        while (sum > maxSum && i > 0) {
            sum -= a[--i];
        }
        if (sum <= maxSum && i + j > count) {
            count = i + j;
        }
    }
    return count; // Mengembalikan nilai 'count' yang merepresentasikan jumlah maksimum elemen yang dapat diambil dari kedua tumpukan melebihi maxSum
}

int main() {
    int g; // Membaca jumlah kasus uji dan melakukan iterasi sebanyak jumlah kasus uji tersebut
    scanf("%d", &g);
    while (g--) {
        int n, m, maxSum; // Unutk membaca ukuran tumpukan A, tumpukan B dan maxSum
        scanf("%d%d%d", &n, &m, &maxSum);
        int a[n], b[m];
        for (int i = 0; i < n; i++) {
            scanf("%d", &a[i]);
        }
        for (int i = 0; i < m; i++) { // Membaca elemen-elemen tumpukan B
            scanf("%d", &b[i]);
        }
```

```
        printf("%d\n", twoStacks(maxSum, a, n, b, m)); // Memanggil fungsi twoStacks untuk menemukan jumlah maksimum elemen yang dapat diambil dari kedua tumpukan, lalu mencetak hasilnya
    }
    return 0; // Mengembalikan nilai 0 untuk menandakan bahwa program telah berakhir
}
```

Output :