

```
#include <stdio.h>
#include <stdlib.h>

struct elem {
    int      zahl;          /* Listenelement */
    struct elem *next;      /* Wert */
                        /* Zeiger auf naechstes Element */
};

/* Listenzeiger auf oberes Element, am Anfang leer */
static struct elem *liste = NULL, *temp = NULL;

int isFull() { return 0; }          /* Liste niemals voll */

int isEmpty() { return liste == NULL; } /* Liste leer ? */

void push(int wert) {
    struct elem *neu = /* neues oberes Element */
        (struct elem *)malloc(sizeof *neu);
    neu->zahl = wert;    /* Anlegen neues oberes Element */
    neu->next = liste;   /* Wert eintragen */
    liste = neu;        /* zeige auf bisheriges oberes El. */
                        /* zeige auf neues oberes Element */
}

void pop() {
    /* oberes Element entfernen */
    if (isEmpty()) {
        printf("Stack leer\n");
        return;
    }
    temp = liste->next;
    free(liste);    /* Speicherfreigabe oberListenelement */
    liste = temp;   /* zeige auf bisher vorletztes Element */
}

int top() {
    /* Wert des oberen Elementes liefern */
    if (isEmpty()) {
        printf("Stack leer\n");
        return -1;
    }
    return liste->zahl;
}

int popTop() { /* oberes Element liefern und entfernen */
    int wert = top();
    pop();
    return wert;
}
```

```
int main(void) {
    int z=0;

    while (1) {
        printf("int-Zahl (F6 Ende): ");
        scanf("%d", &z); fflush(stdin);
        if(feof(stdin))
            break;
        push(z); z=0;
    }

    printf("\nInhalt Stack: ");

    while (!isEmpty())
        printf("%d, ", popTop());

    printf("\n");

    getchar();
    return 0;
}
```