

```

#include <stdio.h>
#include <malloc.h>

typedef int telem;
typedef struct no {
    telem dado;
    struct no *prox;
} tno;
typedef tno *tpilha;

void criar (tpilha *p) {
    *p = NULL;
}

int vazia(tpilha p) {
    return (p == NULL);
}

int elemtopo (tpilha p, telem *elem) {
    if (vazia(p)) return 0;

    *elem = p->dado;
    return 1;
}

int push (tpilha *p, telem valor) {
    tpilha novo;
    novo = (tno*) malloc (sizeof(tno));

    if (novo == NULL) return 0;

    novo->dado = valor;
    novo->prox = *p;
    *p = novo;

    printf("\n Dado inserido: %d", (*p)->dado);

    return 1;
}

int pop (tpilha *p, telem *valor) {
    tpilha aux;
    if (vazia(*p)) return 0;

    aux = *p;

    *valor = (*p)->dado;
    *p = aux->prox;

    printf("\n Dado retirado: %d", *valor);

    free(aux);
    return 1;
}

int main(int argc, char *argv[])
{
    tpilha P1;
    int x;

    criar(&P1);

```

```
push(&P1,10);  
push(&P1,13);  
push(&P1,15);  
  
pop(&P1, &x);  
  
getchar();  
  
return 0;  
}
```