

```
main.c
```

```
//exercício do editor de texto
```

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include "pilha.h"

int main(void) {
    elem c, x;
    int erro;
    Pilha P;
    Create(&P);
    printf("Digite seu texto: ");
    while ((c = getche()) != '\r') {
        if (c == '#') {
            Pop(&P, &x, &erro);
            if (erro)
                printf("(erro) ");
            else
                printf("(%c desempilhado) ", x);
        }
        else if (c == '@') {
            Empty(&P);
            printf("(pilha esvaziada) ");
        } else {
            Push(&P, &c, &erro);
            if (erro)
                printf("(erro) ");
        }
    }
    printf("\n\nDesempilhando tudo: ");
    while (!IsEmpty(&P)) {
        Pop(&P, &x, &erro);
        if (erro)
            printf("erro ");
        else
            printf("%c", x);
    }
    return 0;
}
```

```
pilha.c
```

```
#include "pilha.h"
```

```
void Create(Pilha *P) {  
    P->topo = -1;  
    return;  
}
```

```
void Empty(Pilha *P) {  
    P->topo = -1;  
    return;  
}
```

```
int IsEmpty(Pilha *P) {  
    if (P->topo == -1)  
        return 1;  
    else  
        return 0;  
}
```

```
int IsFull(Pilha *P) {  
    if (P->topo == TamPilha - 1)  
        return 1;  
    else  
        return 0;  
}
```

```
void Push(Pilha *P, elem *X, int *erro) {  
    if (!IsFull(P)) {  
        *erro = 0;  
        P->topo++;  
        P->itens[P->topo] = *X;  
    } else  
        *erro = 1;  
    return;  
}
```

```
void Pop(Pilha *P, elem *X, int *erro) {  
    if (!IsEmpty(P)) {  
        *erro = 0;  
        *X = P->itens[P->topo];  
        P->topo--;  
    } else  
        *erro = 1;  
    return;  
}
```

`pilha.h`

```
#define TamPilha 100
```

```
typedef char elem;
```

```
typedef struct {  
    int topo;  
    elem itens[TamPilha];  
} Pilha;
```

```
void Create(Pilha*);
```

```
void Empty(Pilha*);
```

```
int IsEmpty(Pilha*);
```

```
int IsFull(Pilha*);
```

```
void Push(Pilha*, elem*, int*);
```

```
void Pop(Pilha*, elem*, int*);
```