//==============================================================================

#include <stdio.h>

#include <stdlib.h>

//==============================================================================

typedef struct reg\_pilha{

int info;

struct reg\_pilha \*lig ;

};

typedef struct reg\_pilha \*tpont;

typedef tpont pilha;

pilha p;

//==============================================================================

tpont\* criar (pilha p);

tpont\* push (int x,pilha p);

void vazia (pilha p);

void top (pilha p);

tpont\* pop(pilha p);

void imprimir(pilha p);

//==============================================================================

int main()

{

int choice,x;

for(;;){

printf("\n--------------------MENU---------------------------\n\n");

printf("1 - Para Criar pilha vazia\n");

printf("2 - Para Empilhar(PUSH)\n");

printf("3 - Para Verificar se a pilha esta vazia\n");

printf("4 - Para Desempilhar(POP)\n");

printf("5 - Para Acessar o topo da pilha\n");

printf("6 - Mostrar elementos da pilha\n");

printf("7 - Para Sair\n");

printf("\n----------------------------------------------------\n\n");

printf("Informe A Opcao: ");

scanf("%d", &choice);

switch(choice) {

case 1:

p = criar(p);

break;

case 2:

printf("Digite o numero a inserir\n");

scanf("%d",&x);

p = push(x,p);

break;

case 3:

vazia (p);

system("pause");

break;

case 4:

pop (p);

break;

case 5:

top (p);

system("pause");

break;

case 6:

imprimir(p);

system("pause");

break;

case 7:

exit (1);

}

system("cls");

}

system("pause");

return 0;

}

//==============================================================================

pilha\* criar (pilha p)

{

p = NULL;

return p;

}

//==============================================================================

pilha\* push (int x,pilha p)

{

pilha pont;

if((pont = malloc(sizeof(tpont)))==NULL)

printf("Memoria insulficiente\n");

else{

pont->info = x;

pont->lig = p;

}

return pont;

}

//==============================================================================

void vazia (pilha p)

{

if(p == NULL)

printf("Pilha vazia\n");

else

printf("Pilha nao vazia\n");

}

//==============================================================================

void top (pilha p)

{

if(p!=NULL)

printf("Topo da pilha %d",p->info);

else

printf("Pilha vazia\n");

}

//==============================================================================

pilha\* pop(pilha p)

{

tpont aux;

if(p==NULL){

printf("Nao ha elementos na pilha\n");

return;

}

aux = p->lig;

free(p);

return aux;

}

//==============================================================================

void imprimir(pilha p)

{

tpont aux;

if(p!= NULL)

for(aux = p;aux!=NULL;aux=aux->lig)

printf("%d ",aux->info);

else

printf("Pilha vazia\n");

}