Stack(Con Trỏ)

Gi Chú: Với đề thi có chử Stack, Con Trỏ, Tên Func cũng là tên yêu cầu đề

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<malloc.h>

typedef char ElementType;

typedef struct Node{

ElementType Elements;

Node\* Next;

};

typedef Node\* Position;

typedef Position Stack;

Stack S;

void MakeNull\_Stack(Stack\*S){ // kh?i t?o stack r?ng

(\*S)=(Node\*)malloc(sizeof(Node));

(\*S)->Next = NULL;

}

int Empty\_Stack(Stack S){ // ki?m tra r?ng

return (S->Next == NULL);

}

void Push(ElementType X, Stack\*S){ // thêm ph?n t?

Position T;

T = (Node\*)malloc(sizeof(Node));

T->Elements = X;

T->Next = (\*S)->Next;

(\*S)->Next = T;

}

ElementType Top(Stack S){// l?y ph?n t? d?u tiên

if(Empty\_Stack(S))

printf("Errors !! Stack rong");

else

return (S->Next->Elements);

}

void Pop(Stack\* S){ // xóa ph?n t? d?u tiên

if (!Empty\_Stack(\*S)){

Position T = (\*S)->Next;

(\*S)->Next = T->Next;

free(T);

}

else printf("Ngan xep rong");

}

char BoChuyenDoi(int b){

switch (b){

case 0:

return '0';

case 1:

return '1';

case 2:

return '2';

case 3:

return '3';

case 4:

return '4';

case 5:

return '5';

case 6:

return '6';

case 7:

return '7';

case 8:

return '8';

case 9:

return '9';

case 10:

return 'A';

case 11:

return 'B';

case 12:

return 'C';

case 13:

return 'D';

case 14:

return 'E';

case 15:

return 'F';

}

}

void Doi\_NP ( int n,int CS){

Stack S;

ElementType Du,X;

MakeNull\_Stack(&S);

while(n!=0){

Du=n%CS;

Push(BoChuyenDoi(Du),&S);

n/=CS;

}

while (!Empty\_Stack(S)){

X=Top(S);

printf("%c",X);

Pop(&S);

}

}

int KiemTraDungSai(char X[]){

Stack S;int l,i;

MakeNull\_Stack(&S);

int ktra = 1;

l =strlen(X);

i=0;

while((i<l)&& ktra == 1){

if(X[i]=='('){

Push(X[i],&S);

i++;

}else

if((X[i]==')') &&(!Empty\_Stack(S))){

Pop(&S);

i++;

}else

ktra = 0;

}

if((i == l)&&Empty\_Stack(S)){

return 1;

}else

return 0;

}

int main(){

unsigned int n;

int b;

printf("Nhap n: ");scanf("%d",&n);

printf("Nhap co so: ");scanf("%d",&b);

char Test[256];

printf("Nhap chuoi :");

scanf("%s",&Test);

Stack S;

MakeNull\_Stack(&S);

printf("Ket Qua khi doi la :");

Doi\_NP(n,b);

printf("\n");

if(KiemTraDungSai(Test) == 1){

printf("Dung quy tac toan hoc !!");

}else

printf("Sai quy tac toan hoc !!");

return 0;

}