E3 Postfix Evaluation

#include<stdio.h>

#include<stdlib.h>

#include<stdio.h>

#include<string.h>

int size=10;

struct stack

{

char \*A;

int top;

}\*a1;

void create()

{

a1=(struct stack\*)malloc(sizeof(struct stack));

a1->A=(char\*)malloc(size\*sizeof(char));

a1->top=-1;

}

void push(int n)

{

if(a1->top==(size-1))

{

printf("stack full\n");

}

else

{

a1->A[++a1->top]=n;

}

}

int pop()

{ int val;

if(a1->top==-1)

{

printf("stack empty\n");

return -1;

}

else

{ val=a1->A[a1->top];

a1->top--;

return val;

}

}

int main()

{

void create();

int pop();

void push(int);

void traverse(struct stack \*a1);

char s[20],a,b;

int k=0,t;

create();

printf("enter the postfix expression\n");

gets(s);

while(s[k]!='\0')

{

switch(s[k])

{

case '0':push(0);break;

case '1':push(1);break;

case '2':push(2);break;

case '3':push(3);break;

case '4':push(4);break;

case '5':push(5);break;

case '6':push(6);break;

case '7':push(7);break;

case '8':push(8);break;

case '9':push(9);break;

}

if(s[k]=='+')

{

b=pop();

a=pop();

t=((int)a)+((int)b);

push(t);

}

if(s[k]=='-')

{

b=pop();

a=pop();

t=((int)a)-((int)b);

push(t);

}

if(s[k]=='/')

{

b=pop();

a=pop();

t=((int)a)/((int)b);

push(t);

}

if(s[k]=='%')

{

b=pop();

a=pop();

t=((int)a)%((int)b);

push(t);

}

if(s[k]=='\*')

{

b=pop();

a=pop();

t=((int)a)\*((int)b);

push(t);

}

k++;

}

printf("\nEvaluation result : %d \n",((int)(a1->A[0])));

}