201533661 이승수’s hw3 (dew day 11/30)

<hw1>

#pragma warning(disable:4996)

#include <stdio.h>

int pack(char c1, char c2, char c3, char c4)//return을 long으로 하니깐 함수type도 long이 되어야하지 않나요?

{

long sum=0;

sum = (c1 << 24) | (c2 << 16) | (c3 << 8) | c4;

return sum;

}

void main()

{

char a, b, c, d;

int i;

for (i = 0; i < 5; i++)//5 case

{

printf("\n%d.Put (a b c d) the 1byte character:",i);

scanf("%c %c %c %c", &a, &b, &c, &d);

getchar();

printf("%d", pack(a, b, c, d));

}

getchar();

getchar();

getchar();

getchar();

getchar();

}

<hw2>

#pragma warning(disable:4996)

#include <stdio.h>

#include <stdlib.h>

struct STACK

{

int key;

struct STACK\* nextptr;

};

int is\_empty(struct STACK \*head)

{

if (head == NULL)

return 1;

else

return 0;

} // 1 2 3 4 5 6

void push(struct STACK\*\* Preptr, int value, struct STACK \*headStore)

{

if (is\_empty(\*Preptr))//head가 들어갔을 때

{

struct STACK \*Newptr = (struct STACK\*)malloc(sizeof(struct STACK));

\*Preptr = Newptr;

Newptr->key = value;

Newptr->nextptr = NULL;

headStore = \*Preptr;

}

else

{

struct STACK \*Newptr = (struct STACK\*)malloc(sizeof(struct STACK));

Newptr->key = value;//새로 push해서 만든 ptr의 key에 value

Newptr->nextptr = NULL;

Newptr->nextptr = \*Preptr;

\*Preptr = Newptr;

//ptr nextptr NULL

}

}//ok

int pop(struct STACK\* Curptr)//&head~마지막 전

{

int temp;

while (((Curptr)->nextptr) != NULL)

{

Curptr = Curptr->nextptr;

}

Curptr = NULL;

temp = Curptr->key;

free(Curptr);

return temp;

}

void display(struct STACK \*head)

{

struct STACK \*Curptr = NULL;

Curptr = head;

if (is\_empty(head))

{

printf("\nStack is empty");

}

while (Curptr != NULL)

{

printf("%d<-", Curptr->key);

Curptr = (Curptr->nextptr);

}

printf("NULL");

}

int print()

{

int key;

printf("\nPut key\_value(1:push key,2:pop key,else:end stack)");

scanf\_s("%d", &key);

return key;

}//end stacking function

void main()

{

struct STACK\* start = NULL;

struct STACK \*headStore = NULL, \*TempSTACK1, \*TempSTACK2;

FILE \*inF, \*outF;

int i;//

char buf[5], temp1, temp2;

inF = fopen("postfix.txt", "r");

while (!feof(inF))

{

fscanf(inF, "%c", buf);

if (buf[0] = '.')

{

printf("\n%d", start->key);//result

}

else if (buf[0] == '\*' || buf[0] == '+' || buf[0] == '-')

{

TempSTACK1 = headStore;

TempSTACK2 = headStore;

temp2 = pop(TempSTACK2);

temp1 = pop(TempSTACK1);

if (buf == '\*')

push(&start, temp1\*temp2, headStore);

else if (buf == '+')

push(&start, temp1&temp2, headStore);

else if (buf == '-')

push(&start, temp1&(~temp2), headStore);

}

else//number

{

push(&start, buf, headStore);

}

display(start);

}

fclose(inF);

}

<hw3>

#pragma warning(disable:4996)

#include <stdio.h>

#include <stdlib.h>

void printEng(char num)

{

switch (num)

{

case '0':

printf("zero ");

break;

case '1':

printf("one ");

break;

case '2':

printf("two ");

break;

case '3':

printf("three ");

break;

case '4':

printf("four ");

break;

case '5':

printf("five ");

break;

case '6':

printf("six ");

break;

case '7':

printf("seven ");

break;

case '8':

printf("eight ");

break;

case '9':

printf("nine ");

break;

}

}

void main()

{

char temp[11];

int i=0;

printf("Enter an integer:");

scanf("%s",temp);

printf("\n%s",temp);

printf("\nThe digit that number is:\n");

while (temp[i]!='\n')

{

printEng(temp[i]);

i++;

}

printf("\nif you add ten to that number,\nthe digit is in the new number are: \n");

i = 0;

temp[1] += 1;

while (temp[i] != '\n')

{

if (temp[i] == '\n')

printEng(temp[i-1]+1);

else

printEng(temp[i]);

i++;

}

getchar();

getchar();

getchar();

getchar();

getchar();

}