

電気情報系実験3

プログラミング例

～fopen fclose Makefile etc...～

2017 年10月18日

M17J4059Y 安場 裕人

Makefile

CC = gcc

CFLAGS = -g -Wall

PROGRAMS = input_data.o count_data.o output_data.o main.o main

all: \$(PROGRAMS)

clean:

rm -f *.o *~

input_data.o: input_data.c parameter.h

\$(CC) \$(CFLAGS) -c \$^

count_data.o: count_data.c parameter.h

\$(CC) \$(CFLAGS) -c \$^

output_data.o: output_data.c parameter.h

\$(CC) \$(CFLAGS) -c \$^

main.o: main.c parameter.h

\$(CC) \$(CFLAGS) -c \$^

main: main.o input_data.o count_data.o output_data.o

\$(CC) \$(CFLAGS) -o \$@ \$^

parameter.h

```
/******  
name; network.h  
abstract; network modules header  
date; 2017/10/16  
writer; Yaseba Hiroto  
  
using packages;  
    standard.h  
  
fixed date;  
  
    *****/  
#include <stdio.h>  
#include <string.h>  
#include <stdlib.h>  
  
/******  
    define numbers  
    *****/  
#define MAX_NAME 30  
#define MAX_DATA 20  
#define MAX_STR 10  
  
/******  
    define structure  
    *****/  
typedef struct Data{  
    char first[MAX_STR];  
    char second[MAX_STR];  
    int third;  
} DATA;
```

main.c

```
/******  
name; main.c  
abstract; test program (function fopen fclose)  
          (and cross compile)  
date; 2017/10/16  
writer; Yaseba Hiroto  
  
using packages;  
    test.h  
  
fixed date;  
  
*****/  
#include "parameter.h"  
  
int main(int argc, char *argv[]){12  
  
    //incorrect calling alert  
    if(argc != 2){  
        printf("main filename.txt.\n");  
        return -1;  
    }  
  
    //variable define  
    //for filename  
    char file[MAX_NAME];  
    //structure of data  
    DATA data[MAX_DATA];  
    //counter of data  
    int count;  
  
    //function define  
    void input_data(DATA * data, char * filename); //from input_data.c  
    int count_data(DATA * data); //from count_data.c  
    void output_data(DATA * data, int c); //from output_data.c  
  
    //catch a filename  
    if((strcpy(file, argv[1])) == NULL) return -1;  
  
    //catch filename's data
```

```
input_data(data, file);

//count data
count = count_data(data);
//output filename' data
output_data(data, count);

//end of program
return 0;
}
```

input_data.c

```
#include "parameter.h"

/*
name; input_data
abstract; this input filename's data to DATA[]
expected data;
char * first \t char * second \t int third
char * first \t char * second \t int third
...
...
...

date; 2017/10/16
writer; Yaseba Hiroto

input; DATA * data
      char * filename
output; DATA * data

fixed date;
*/
void input_data(DATA * data, char * filename){
    //define variable
    //for roop
    int i;

    //file pointer
    FILE *fp;
    //temps
    char temp_first[MAX_STR];
    char temp_second[MAX_STR];
    int temp_third;

    //initialization temp_first
    //strcpy(temp_first, "NULL");

    //open file
    if((fp = fopen(filename, "r")) == NULL) {
        printf("you can't open file\n");
        return;
    }
}
```

```

    }

//roop to file end or MAXDATA
//catch data to temps by fscanf
for(i = 0; (((fscanf(fp,"%s\t%s\t%d\n", temp_first, temp_second, &temp_third)) != -1) &&
                                                    (i < MAX_DATA)) ; i++){

//copy data to DATA * data
    strcpy(data[i].first, temp_first);
    strcpy(data[i].second, temp_second);
    data[i].third = temp_third;
//end roop
}

//insert last data
//first NULL
//scond NULL
//third 0
strcpy(data[i].first,"NULL");
strcpy(data[i].second,"NULL");
data[i].third = 0;

//file close
fclose(fp);

//function catch end
}

```

count_data.c

```
#include "parameter.h"
```

```
/*
```

```
name; count_data
```

```
abstract; this count data in DATA structure
```

```
date; 2017/10/16
```

```
writer; Yaseba Hiroto
```

```
packaged in; test.c
```

```
input; DATA * data
```

```
output; int counter;
```

```
fixed date;
```

```
*/
```

```
int count_data(DATA * data){
```

```
    //define variable
```

```
    //for roop
```

```
    int i;
```

```
    //counter
```

```
    int counter = 0;
```

```
    //roop to find first is NULL or MAX_DATA
```

```
    for(i = 0; ((strcmp(data[i].first, "NULL") != 0) && (i < MAX_DATA)); i++){
```

```
        //counter increment
```

```
        counter++;
```

```
    //roop end
```

```
}
```

```
    //program end return counter
```

```
    return counter;
```

```
}
```


output_data.c

```
#include "parameter.h"

/*
name; output_data
abstract; this output data, repeated to c
output form
page|first  |second  |third|
-----
1|first  |second  |third|
-----
2|first  |second  |third|
-----
...
...

date; 2017/10/16
writer; Yaseba Hiroto

input; DATA * data
      int c
output; void

fixed date;
*/
void output_data(DATA * data, int c){
    //variable define
    //for roop
    int i, n;
    //width;
    int width;
    width = 4 + 1 + 10 + 1 + 10 + 5 + 1;

    //ceiling
    printf("page|first  |second  |third\n");
    for(i = 0; i < width; i++){
        printf("-");
    }
    printf("\n");

    //data writing roop to c or MAX_DATA
    for(i = 0; (i < c) && (i < MAX_DATA); i++){
```

```
printf("%4d|%10s|%10s|%5d\n",i+1 ,data[i].first, data[i].second, data[i].third);
for(n = 0; n < width; n++){
    printf("-");
}
printf("\n");
//data writing roop end
}

//inform data end
printf("data end\n");

//function end
}
```

move.sh

```
#!/bin/sh
```

```
#test using  
rm result.txt  
make clean  
make main
```

```
./main data.txt > result.txt
```

```
cat result.txt
```

data.txt

```
jack   kkk   12  
jon    ggg   37  
bbb    west  49  
koji   quart 89  
clover mike  90
```

result.txt

```
page|first  |second  |third|
```

```
-----  
1|   jack|   kkk|  12
```

```
-----  
2|   jon|   ggg|  37
```

```
-----  
3|   bbb|  west|  49
```

```
-----  
4|  koji| quart|  89
```

```
-----  
5| clover|  mike|  90
```

```
-----  
data end
```