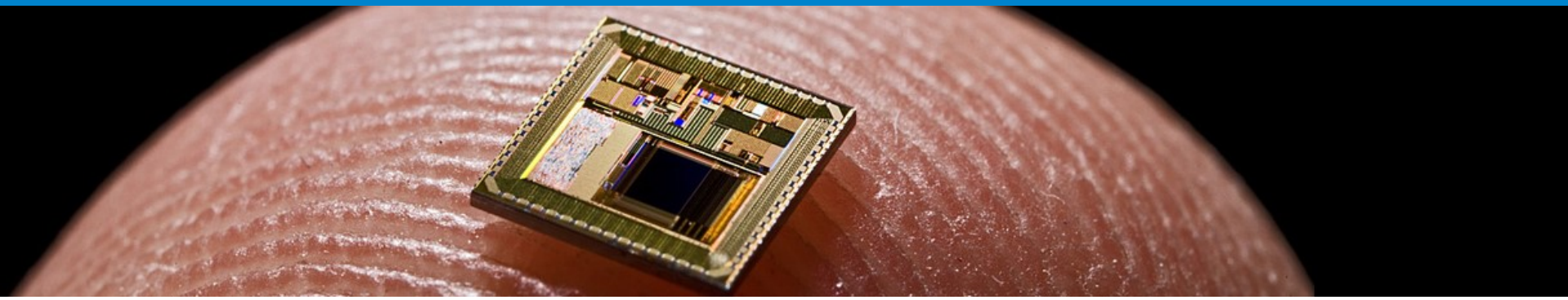
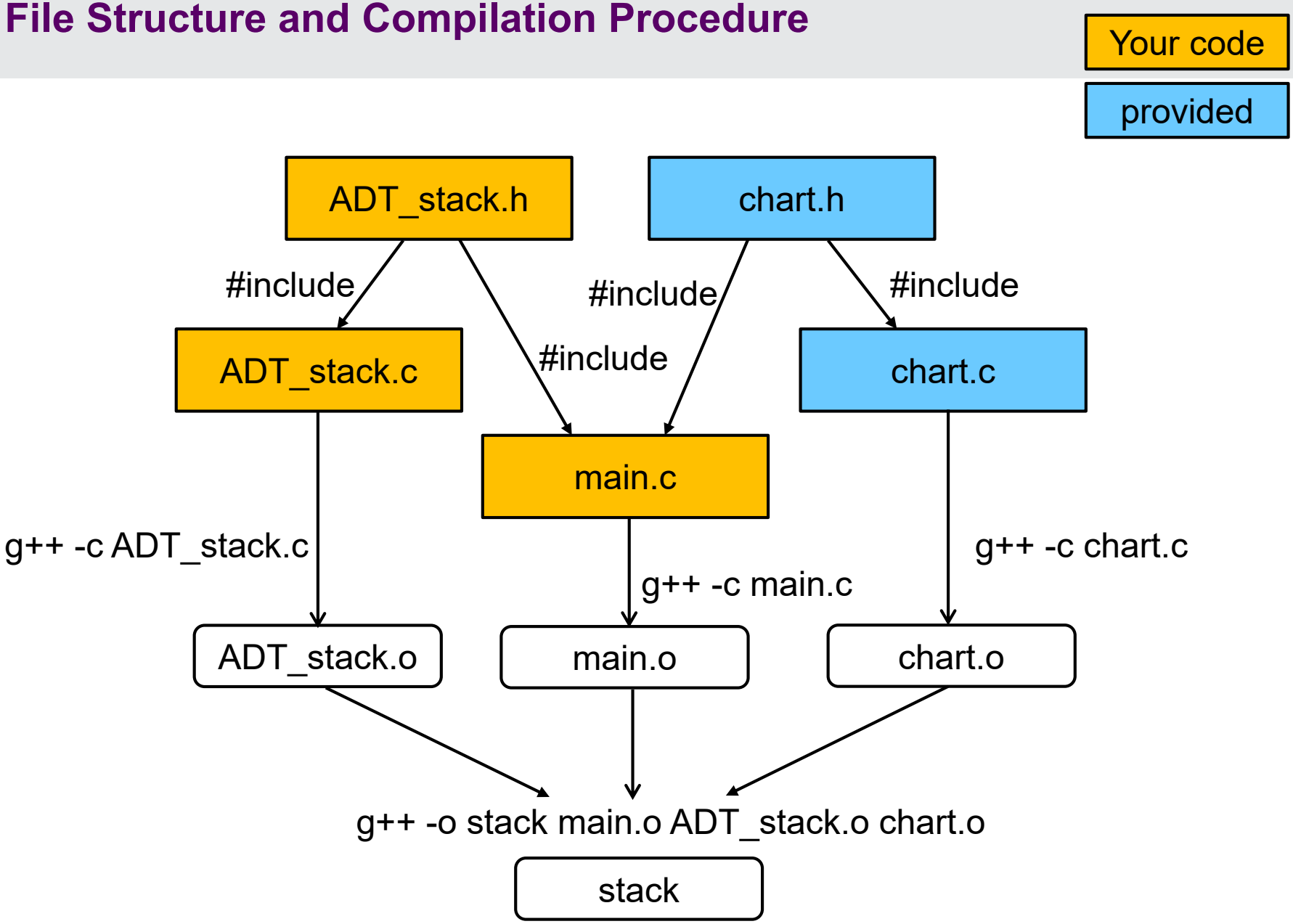
Data Structure
HW1
- inversion using stack

2017-09-20



Kyungpook National University / Daejin Park

File Structure and Compilation Procedure



© 2017 KNU

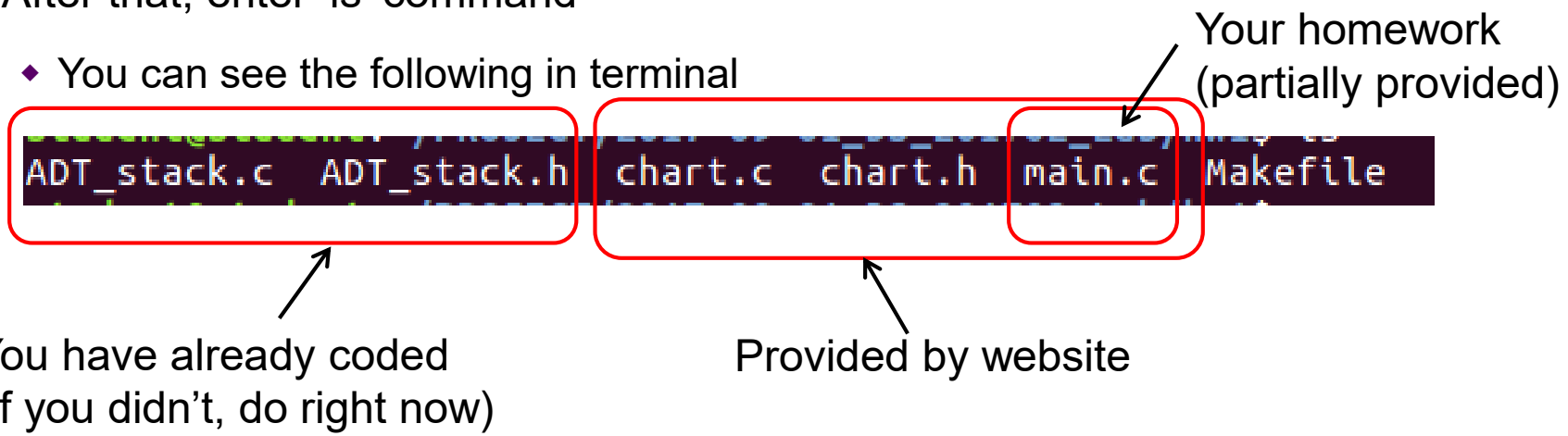
Edit Makefile

```
ADT_stack.o: ADT_stack.c
	g++ -c ADT_stack.c

main.o: main.c
	g++ -c main.c

all: ADT_stack.o main.o
	g++ -o stack main.o ADT_stack.o

run: all
	./stack > chart.html

clean:
	rm -rf *.o stack
```


© 2017 KNU

Download chart.h chart.c,main.c

- Download chart.zip
  - Extract into your folder (three files are extracted, chart.h chart.c,main.c)
  - After that, enter ‘ls’ command
    - You can see the following in terminal



© 2017 KNU

Print student list in forward way (this code is provided, just run)

```
// ADT Stack
#include <stdio.h>
#include "ADT_stack.h"
#include "chart.h"

//#define REVERSE 0 // reverse

int main() {

    // new type definition
    typedef struct {
        char name[20];
        int score;
    } STD;

    // prepare 5 data
    STD student[5] = {
        {"James", 95},
        {"Yoosoo", 87},
        {"Paul", 93},
        {"Peter", 76},
        {"Park", 100}
    };

#ifndef REVERSE
    generate_chart_header();
    //                    me                  my boss             my score
    generate_chart_node(student[0].name, student[0].name, student[0].score);
    generate_chart_node(student[1].name, student[0].name, student[1].score);
    generate_chart_node(student[2].name, student[1].name, student[2].score);
    generate_chart_node(student[3].name, student[2].name, student[3].score);
    generate_chart_node(student[4].name, student[3].name, student[4].score);
    generate_chart_footer();
#else


#endif

    return 0;
}
```

This list has to be commented

The right html file is generated by using code below

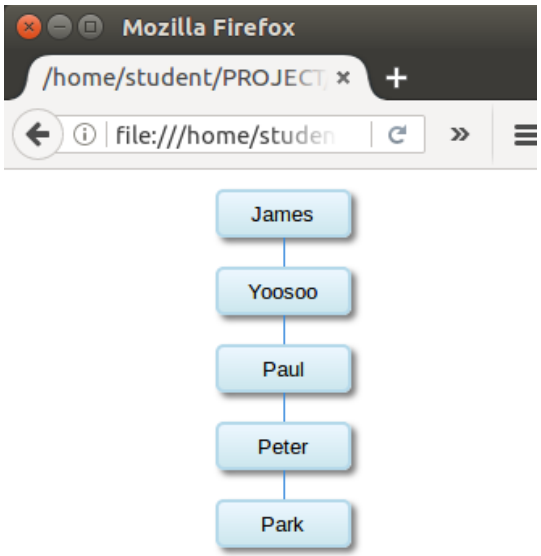
On terminal, type the following:

make all

make run

chart.html

Will be created, open it with browser



© 2017 KNU

Print student list in reverse way (your home, fill out the blank box)

```c
// ADT Stack
#include <stdio.h>
#include "ADT_stack.h"
#include "chart.h"

#define REVERSE 0 // reverse

int main() {

    // new type definition
    typedef struct {
        char name[20];
        int score;
    } STD;

    // prepare 5 data
    STD student[5] = {
        {"James", 95},
        {"Yoosoo", 87},
        {"Paul", 93},
        {"Peter", 76},
        {"Park", 100}
    };

#ifndef REVERSE
    generate_chart_header();
    //                    me                 my boss          my score
    generate_chart_node(student[0].name, student[0].name, student[0].score);
    generate_chart_node(student[1].name, student[0].name, student[1].score);
    generate_chart_node(student[2].name, student[1].name, student[2].score);
    generate_chart_node(student[3].name, student[2].name, student[3].score);
    generate_chart_node(student[4].name, student[3].name, student[4].score);
    generate_chart_footer();
#else

    // start here, for homework
    // Stack Creation
    [                                    ]

    // push them
    int i;
    for(i=0; i<5; i++) {
    [                                    ]
    }

    // pop them
    STD* boss=NULL; // backup previous boss
    STD* std;
    generate_chart_header();
    while([                                    ]) {
    [                                    ]
    }
    generate_chart_footer();

#endif

    return 0;
}
```

Delete comment

Fill out code

On terminal, type the following:

make all

make run

chart.html

Will be created, open it with browser, you can see the reversed student list

Submit (Due: 9/28, PM 10:00)

- Korean student needs to submit two files, into ABEEK website
  - (1) Source code:
    - Compress your homework folder, named hw1_[id].zip
      - For example, **hw1_20161235.zip**
  - (2) Report
    - In addition, **attach the report** (Microsoft word format) to explain your homework in terms of implementation.
- Foreign students have to mail me directly with these two files as attachment
  - boltanut@knu.ac.kr

© 2017 KNU

Kyungpook National University /
Daejin Park

Cloud-Connected IoT System Platform Lab.
http://CCIoTLab.com/come331

To be continued …