



SUMMER SESSION II 2013
EECS 10 WEEK1 DISCUSSION2
Che-Wei Chang

PART 1 OF ASSIGNMENT 1

- Print your initials on the screen
 - Example: UCI
 - Tips: Lecture slides page 20, “Hello World” example
 - Simple modification of the “Hello World” example



PART 2 OF ASSIGNMENT 1

- Add two timestamps and print the result
 - Each time stamps contains **hours**, **minutes**, and **seconds**
 - Prompt for **hours**, **minutes**, and **seconds** for each timestamp
 - Read two timestamps through stdin
 - Add two timestamps, and print the result on the screen.

- Example:

- Timestamp1: hour: 3 minute: 14 second: 9
- Timestamp2: hour: 2 minute: 10 second: 8
- Result: hour: 5 minute: 24 second: 17

=====

- Timestamp1: hour: 12 minute: 34 second: 43
- Timestamp2: hour: 4 minute: 42 second: 39
- Result: hour: **16** minute: **76** second: **82**
- Result: hour: **17** minute: **17** second: **22**



PART2 OF ASSIGNMENT 1 (COND.)

- How to read input from stdin ??

- function `scanf`

- lecture 2 slides, page 6 and 7, example “Addition.c”

- `/* Addition.c adding two integer numbers */` ← **comments**

```
#include <stdio.h> ← insert header file
```

```
int main (void) ← main function
```

```
{  
  int input1 ;  
  int input2 ; ← variable declaration  
  int sum ;
```

```
  scanf(“%d”, &input1) ; ← input  
  scanf(“%d”, &input2) ;
```

```
  sum = input1 + input2 ; ← computation
```

```
  printf(“The sum of %d and %d is %d\n”, input1, input2, sum) ; ← output  
  return 0 ;
```

```
}
```



PART2 OF ASSIGNMENT 1 (COND.)

- After read two inputs, how to compute the output timestamp?

- Divide (/)and Modulus (%) operator

ex:

```
int A = 15;
```

```
int B = 4;
```

```
int C, D ;
```

```
C = A / B ; ← What is C ? 3.75 / 3 / 4 ←quotient
```

```
D = A % B ; ← What is D ? 3 ←remainder
```

- Bonus:
 - Add **day** and **week** to the timestamps.



FLOW TO DO YOUR FIRST ASSIGNMENT

- Step1: Login into your account
 - Step1.1: change your password
- Step2: Create directory “*hw1*” for the assignment
- Step3: Create/Edit the c file(s) (initials.c, timestamp.c)
- Step4: Compile the c file(s)
- Step5: run the executable, and check the results
- Step6: Create the script file(s) (ex. initials.script)
- Step7: Briefly describe your work in .txt file
- Step8: Submit your work.



LOGIN INTO YOUR ACCOUNT

- Windows: putty, OpenSSH...
 - Type *zuma.eecs.uci.edu* or *crystalcove.eecs.uci.edu* in the “Host Name (or IP address)” field.
 - Hit “Open”
 - Input your UCINetID and password correctly
- MacOS: Terminal
 - Use the following command:
 - > **ssh** *zuma.eecs.uci.edu* -x -l *YourUserName* or
 - > **ssh** *YourUserName@zuma.eecs.uci.edu*
- Modify your password
 - > **yppasswd**



CREATE DIRECTORY FOR THE HOMEWORK

- Create a new directory
> **mkdir** *name_of_the_directory*
ex: > **mkdir** hw1
- List the files and directories in the current directory
> **ls**
- Change working directory
> **cd** *name_of_the_target_directory*
ex: > **cd** hw1 (go into hw1)
 > **cd** .. (back to upper level)
- Remove a directory
> **rmdir** *name_of_the_directory*
ex: > **rmdir** hw1



EDIT/COMPILE THE C FILE(S)

RUN THE EXECUTABLE

- Using pico to edit your file
> **pico** *filename*
ex: > **pico** initial.c
- Compile your c program
> **gcc** *source_file* **-o** *output_file*
ex: > gcc initials.c **-o** initials
- Run the executable, and check the result
> **./executable**
ex: > ./initials



CREATE SCRIPT FILE FOR THIS ASSIGNMENT

- Make typescript of terminal session
 - > **script**
Script started, file is typescript
 - > compile your c file(s) (see previous slides)
 - > run the executable (see previous slides)
 - > **exit** (or **ctrl-d**)
Script done, file is typescript
- After the command above, file **typescript** will be created.
- Display the content of a file on the screen
 - > **cat** *file_name*
 - ex: > **cat** typescript
- Rename a file to another name
 - > **mv** *file_to_be_renamed* *new_file_name*
 - ex: > **mv** typescrip initials.script



SUBMIT YOUR HOMEWORK

- Describe your implementation in a txt file
ex: “it works”, “it failed, because ...”

(Assuming you are in your *hw1* directory)

- Go back to directory one level higher than current one
> `cd ..` (now you are in the working directory containing *hw1* directory)
- Submit your homework
> `/ecelib/bin/turnin10`
(NOTE: use this command while you are in the directory containing *hw1* directory)



FILES YOU SHOULD SUBMIT...

- **Print Initials**
 - initials.c
 - initials.txt
 - initials.script
- **Timestamp Addition**
 - timestamp.c
 - timestamp.txt
 - timestamp.script
- The names of your files should be exactly the same with the names listed above, or they won't be submitted successfully.

