Computer Organization
Lab 1 - MIPS Assembly

教授:蔡文錦
TAs:林浩君、薛乃仁、吳年茵

Objectives

In this lab, we are going to learn how to write assembly code of MIPS architecture, and understand the difference between it and high-level languages such as C/C++. We will give an example to you.

1. Learn how to write **assembly code** and understand how it works
2. Learn how to test your code using a **MIPS simulator** - SPIM

Example: Factorial

- This is an example about computing factorial n!, where n is a given positive integer.
- The attached files **factorial.c** and **factorial.s** are the given example code.
- Please check the above files before moving on to the tasks, for you to get familiar with assembly code.

```
> gcc factorial.c -o factorial
> ./factorial
Please input a number: 5
The result of factorial(n) is 120
```

Tasks

For each task, the corresponding C code is given, you need to translate it into mips assembly, and each task account for 25% of your score.

1. Prime Number
2. Simple Calculator
3. Drawing Triangles
4. Fibonacci Sequence

Task 1. Prime Number(25%)

We want you to write a program that can determine whether a given input number is a prime number or not.

Input: A positive integer n

Examples:

```
> ./prime
Please input a number: 7
It's a prime
> ./prime
Please input a number: 9
It's not a prime
```

Task 2. Simple Calculator (25%)

A pretty simple calculator that can do addition、subtraction and multiplication on two given numbers and show the result.

Input: A option to determine the action and two numbers, n1 and n2, to perform the calculation.

Example:

```
> ./calculator
Please enter option (1: add, 2: sub, 3: mul): 1
Please enter the first number: 5
Please enter the second number: 3
The calculation result is: 8
```

Task 3. Drawing Triangles (25%)

Draw triangles with given parameters

Input: An option to determine version of triangle, and height of triangle

Example:

```
> ./triangle
Please enter option (1: triangle, 2: inverted triangle): 1
Please input a triangle size: 5
    *
   ***
  *****
 *******
*********
```

```
> ./triangle
Please enter option (1: triangle, 2: inverted triangle): 2
Please input a triangle size: 3
*****
 ***
  *
```

Task 4. Fibonacci Sequence (25%)

Read a number from standard input. Output the n-th item of Fibonacci sequence. You are **required to use recursive function** to solve this task.

Input: A positive integer n.

Example:

```
> ./fibonacci
Please input a number: 10
The result of fibonacci(n) is 55
```

SPIM: A MIPS32 Simulator

- SPIM is a simulator for the MIPS, you can simply install SPIM with following command
  - macOS: ***brew install spim***
    - need to install homebrew first: https://brew.sh/
  - Ubuntu: ***sudo apt-get install spim***
  - centOS: ***sudo yum install spim***
- Then you can using spim using command line interface

```
> spim -file factorial.s
SPIM Version 8.0 of January 8, 2010
Copyright 1990-2010, James R. Larus.
All Rights Reserved.
See the file README for a full copyright notice.
Loaded: /usr/lib/spim/exceptions.s
Please input a number: 5
The result of factorial(n) is 120
```

spim log

program IO

Testing Method

- We provide testing script ***test.sh*** to verify wheter the results are identical between the C code and MIPS assembly code.
- Each problem includes two teat cases, but there will also be hidden cases.
  - You can add additional test cases by yourself.

```
> ./test.sh
Testing Problem: factorial
factorial.1 PASS
factorial.2 PASS
-----------------------------------
Testing Problem: prime
Source asm file not found: ./prime.s
-----------------------------------
Testing Problem: calculator
Source asm file not found: ./calculator.s
-----------------------------------
Testing Problem: triangle
Source asm file not found: ./triangle.s
-----------------------------------
Testing Problem: fibonacci
Source asm file not found: ./fibonacci.s
-----------------------------------
```

```
> ./test.sh
Testing Problem: factorial
factorial.1 PASS
factorial.2 PASS
-----------------------------------
Testing Problem: prime
prime.1 PASS
prime.2 PASS
-----------------------------------
Testing Problem: calculator
calculator.1 PASS
calculator.2 PASS
-----------------------------------
Testing Problem: triangle
triangle.1 PASS
triangle.2 PASS
-----------------------------------
Testing Problem: fibonacci
fibonacci.1 PASS
fibonacci.2 PASS
-----------------------------------
```

Grading Policy

- Each task have 5 hidden cases, and you will get **5 points** for each correct testcase, totally 25 points for each task.
- We will use the same method of ***test.sh*** to judge your result, make sure you can pass the testing script on your local environment.
- **Any assignment work by fraud will get a zero point.**
- **No late submission!**

Submission

- The files you should hand in include:
  - **prime.s, calculator.s, triangle.s, fibonacci.s**
- Compress the above file into one zip file, and name your zip file as **HW1_{studentID}.zip (e.g. HW1_312552014.zip)**
  - Make sure not to add an extra folder layer. You can use the following command to zip your files to ensure the format is correct.

```
> zip -r HW1_312552014.zip triangle.s calculator.s prime.s fibonacci.s
  adding: triangle.s (deflated 73%)
  adding: calculator.s (deflated 70%)
  adding: prime.s (deflated 62%)
  adding: fibonacci.s (deflated 68%)
> unzip -l HW1_312552014.zip
Archive:  HW1_312552014.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
     1664  2024-07-05 17:55   triangle.s
     1144  2024-07-05 18:04   calculator.s
      698  2024-07-05 17:55   prime.s
     1083  2024-07-05 18:04   fibonacci.s
---------                     -------
     4589                     4 files
```

compress files into zip file

check zip file format

- **Wrong format will have 10% penalty.**

Reference

- MIPS32 Instruction Set Quick Reference
- System Calls Table
- https://letmegooglethat.com/?q=chatgpt