

HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY



FACULTY OF COMPUTER SCIENCE AND ENGINEERING
COURSE: COMPUTER ARCHITECTURE LAB (CO2008)

Lab 3

floating point instructions & recursive

Ho Chi Minh City, October 23rd 2023



Contents

| | | |
|----------|----------------------|----------|
| 1 | Introduction | 2 |
| 2 | Exercises | 2 |
| 2.1 | Exercise 1 | 2 |
| 2.2 | Exercise 2 | 2 |
| 2.3 | Exercise 3 | 2 |
| 2.4 | Exercise 4 | 3 |



1 Introduction

- The main purpose of this session is to get familiar with floating-point instructions and read/write file.
- Students must submit their answers to the BKeL system no later than the last period of the lab section. Then, the instructor will evaluate all students' work during the lab section's final period. Please note that we will randomly choose ~50% of the questions to mark.

2 Exercises

2.1 Exercise 1

Write a MIPS program that calculates and print either the volume or total surface area of a rectangular box, cube, cylinder or sphere. The user is able to choose which shape and the related parameters to calculate. Note that the parameters can be floating-point numbers.

2.2 Exercise 2

Write a MIPS program to calculate the following integral:

$$f(x) = \int_v^u \frac{ax^4 + bx^3 + cx^2 + d}{e^2} \quad (1)$$

where u, v, a, b, c, d are floating-point numbers chosen by the user, and e equal last digit of your student ID plus 2 (for example, if your student ID is 1234565 then e is 7) For example, if you have the ID 1234565, and user inserted **a=1, b=2, c=3, d=4, u=5, and v=6** then the result should be: **-27.77**

2.3 Exercise 3

In this exercise, students are required to write a recursive program although the problem can be solved by iterations.

Write a MIPS program that print all 10 elements in a single precision floating point array with synthetic data in the reverse order. Bellow is pseudo code of the recursive version:

```
1 float print(float *v, int k){  
2     if (k == 1) cout << v[0];
```



```
3     else {  
4         print(&v[1], k-1);  
5         cout << v[0];  
6     }  
7 }
```

2.4 Exercise 4

Taking the same requirements in Exercise 3, write a MIPS program that is able to find the minimum elements in an array. Bellow is pseudo code of the recursive version:

```
1 float min(float *v, int k){  
2     if (k == 1) return v[0];  
3     float temp = min(&v[1], k - 1);  
4     if (v[0] < temp) return v[0];  
5     else return temp;  
6 }
```