Cairo University, Faculty of Computers and Information

FACULTY OF COMPUTERS AND INFROMATION, CAIRO UNIVERSITY

CS322: Computer Architecture and Organization Year 2020-2021

First Semester

Assignment 3 – Version 3.0

Course Instructors:

Dr. Mohammad El-Ramly

Revision History

Version 1.0	Dr Mohammed El-Ramly	28 Nov. 2019	Version 1.0
Version 2.0	Dr Mohammed El-Ramly	30 Nov. 2019	More elaboration
Version 3.0	Dr Mohammed El-Ramly	20 Dec 2020	2020 version / only assembly

CS322: Computer Architecture and Organization
Assignment 3 (7 marks) – Version 3.0

Cairo University, Faculty of Computers and Information

احبائى الطلاب ، أشكركم على تعبكم فى هذا المقرر و أدعو لكم بالتوفيق فيه و فى كل المقررات ، و أعتذر عن أى تقصير من جانبى و أرحب بأية مقترحات أو تعليقات (لا تشمل تقليل المنهج) ، و أرجو التزام الأمانة العلمية و عدم تقديم أى حل غير أصلى أو منقول من أى مصدر و إلا تعرض الطالب لعقوبات لا أحب أن أوقعها بأى طالب.

و أود التنبيه أن الظروف الحالية تجعل عليك عبئا أكبر فى الاعتماد على الذات و تحصيل المعلومات.

Objectives

1- Learn MIPS assembly and architecture.
2- Understand how programs and algorithms are represented in assembly.

Instructions

1. **These instructions must be followed to get the full marks. يجب اتباع هذه التعليمات بكل دقة**
2. **Deadline is Sunday 3 Dec. 2021 @ 11 pm**
3. **Form groups of 3 students.**
4. **Please submit only work that you did yourself. If you copy work from a friend or book or the net you will fail the course. تسليم حلول منقولة من أى مصدر يؤدى إلى الرسوب فى هذا المقرر. لا تغش الحل أو تنقله من أى مصدر و اسألنى فى أى شئ لا تفهمه لكن لا تنقل الحلول من النت أو زملائك أو أى مكان و لو عثرت على حلول من أى مصدر أو زميل لا تنقل منها أى شئ وقد أعذر من أنذر ولن يتم التهاون مع الغش مطلقا**

Task 0 (0 marks)

1. Get familiar with MIPS-32 assembly language and architecture.
2. Download QTSPIM assembler for MIPS. Install it and test it on small programs.
3. Read Introduction to SPIM https://ece2020.ece.gatech.edu/readings/assembly/spim.pdf
4. Train on problem solving with MIPS assembly and using QTSPIM assembler.
5. Read this link for help https://www.cse.unsw.edu.au/~cs1521/18s2/notes/C/notes.html
6. Watch these videos in Arabic or any others https://cutt.ly/OhVoBFq

Task 1 (7 marks)

Each team member will solve one set of the following programs. Each one includes 4 programs. **The ID with smallest least-significant digit will solve Set 1.** The next will solve **Set 2,** etc. Use MIPS assembly and deliver a working solution that correctly runs on QTSPIM. Solution must have:

- A header explaining the file content, author, version, date, etc.
- Proper coding style and sufficient comments to explain the solution.
- Must be free of any syntax or logic errors.
- Must work properly on QTSPIM simulator.
- Test your programs with sample data.
- Make any assumptions you need.
- **Do not copy any solution.**

CS322: Computer Architecture and Organization
Assignment 3 (7 marks) – Version 3.0

Cairo University, Faculty of Computers and Information

Set 1:

1- Write, run and test a MIPS program to implement the following C code segment. (1 mark)

```
if (g > h)
     g = g + h;
else if (g < h)
     g = g – h;
else
     g = g * h;
```

2- Develop an algorithm, test it and then implement it in MIPS assembly that takes an array of characters and prints the number of UPPERCASE letters and the number of LOWERCASE letters. (1 mark)

3- Write, run and test a MIPS program to execute the following nested C loop. (1 mark)

```
for(i = 0; i < a; i++)
     for(j = 0; j < b; j++)
        C[2 * i] = i – j;
```

4- Translate the following C program to MIPS assembly. (2 marks)

```
int main() {
     ...
     t1 = sumOdd (10);
     printf ("%d", t1);
     ...
}
int sumOdd (int n) {
     int i, result = 0;
     for (i = 0; i < n; i++)
          if ((i % 2) == 1)
               result += i;
     return result;
}
```

5- In A2 we designed a floating point multiplication unit. When a processor does not support floating point arithmetic with hardware, it is still possible to support floating-point operations using software libraries to simulate what the hardware would do. It is required to write a MIPS function that takes two floating-point numbers using IEEE754 and **multiplies** them and returns the result in a third register. (2 marks)

(Hint this will require separating the mantissa and exponent, adding the leading one, multiplying the mantissas together, adding the exponents, subtracting 127, normalizing the result. etc. Ignore special cases like infinity, NaN, but consider the case of zero.)

CS322: Computer Architecture and Organization
Assignment 3 (7 marks) – Version 3.0

Cairo University, Faculty of Computers and Information

Set 2:

1- Write, run and test a MIPS program to implement the following C code segment. (1 mark)

```
if ((g <= h) && (g > 0))
     g = h;
else
     h = g;
```

2- Develop an algorithm, test it and then implement it in MIPS assembly that takes an array of characters and prints the number of vowels ('a', 'e', 'i', 'o', 'u') in it. Consider both upper and lower cases. E.g., "Utility" prints 3. (1 mark)

3- Write, run and test a MIPS program to execute the following nested C loop. (1 mark)

```
for(i = 0; i < a; i++)
    for(j = 0; j < i; j++)
        C[i] += j;
```

4- Translate the following C program to MIPS assembly. **Use the same structure. Do not write a clever shorter program.** (2 marks)

```
int main() {
     ...
     t1 = isOdd (10);
     printf ("%d", (t1 ? "odd" : "even");
     ...
}

int isOdd (int n) {
     return !isEven (n);
}

int isEven (int n) {
     return ((n % 2) == 0));
}
```

5- In A2 we designed a floating point multiplication unit. When a processor does not support floating point arithmetic with hardware, it is still possible to support floating-point operations using software libraries to simulate what the hardware would do. It is required to write a MIPS function that takes two floating-point numbers using IEEE754 and **adds** them and returns the result in a third register. (2 marks)

(Hint this will require separating the mantissa and exponent, adding the leading one, shifting the mantissas with smaller exponent, adding mantissas together, normalizing the result. etc. Ignore special cases like infinity, NaN, but consider the case of zero.)

CS322: Computer Architecture and Organization
Assignment 3 (7 marks) – Version 3.0

Cairo University, Faculty of Computers and Information

Set 3:

1- Write, run and test a MIPS program to implement the following C code segment. (1 mark)

```
if (g >= h)
     g++;
else
     g--;
```

2- Develop an algorithm, test it and then implement it in MIPS assembly that takes an array of characters, then reverses it in memory and prints the reversed version. (1 mark)

3- Write a MIPS program to execute the following nested C loop. (1 mark)

```
for(i = 0; i < a; i++)
    for(j = i; j >= 0; j--)
        C[i] *= j;
```

4- Translate the following C program to MIPS assembly. (2 marsk)

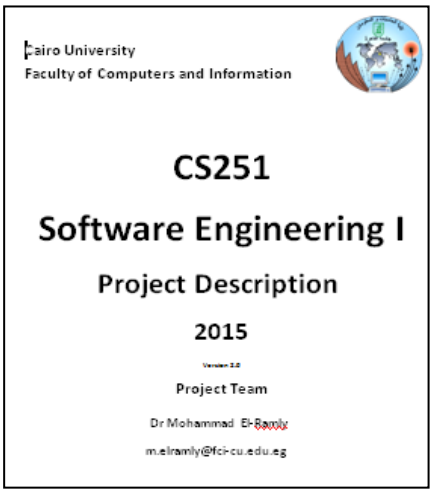
```
int main() {
     ...
     t1 = fact(8);
     t2 = fact(3);
     t3 = t1 + t2;
     ...
}
int fact(int n) {
     int i, result = 1;
     for (i = n; i > 1; i--)
          result  = result * i;
     return result;
}
```

5- Design an algorithm for testing whether a given string is a palindrome. (Recall that a palindrome is a word that is the same forward and backward. For example, the words “wow” and “racecar” are palindromes.) Implement your algorithm using MIPS assembly code. (2 marks)

CS322: Computer Architecture and Organization
Assignment 3 (7 marks) – Version 3.0

Cairo University, Faculty of Computers and Information

Submission Instructions


Cairo University
Faculty of Computers and Information

CS251

Software Engineering I

Project Description

2015

Project Team

Dr Mohammad El-Ramly

m.elramly@fci-cu.edu.eg

1. **Team will submit in acadox.**
2. Submit in acadox a zip file with the following:
3. PDF report with a header as in the figure that includes:
   - Team names, IDs, **and emails,** assignment details, etc.
   - MIPS solutions to the problems given divided into sets with the name of the student did every set.
4. MIPS assembly (.s or .asm files) code nicely organized in folders and named properly.
5. Name file CS322-A3-ID1-ID2.pdf / .zip

```
+
|____ A3
|          +
|          |_ Set1-ID
|          |      +
|          |      |- Problem 1
|          |      |- Problem 2
|          |      |- Problem 3
|          |      |- Problem 4
|          |      |- Problem 5
|          |
|          |_ Set2-ID
|                 +
|          |      |- Problem 1
|          |      |- Problem 2
|          |      |- Problem 3
|          |      |- Problem 4
|          |      |- Problem 5
|          |
|          |_ Set3-ID
|                 +
|          |      |- Problem 1
|          |      |- Problem 2
|          |      |- Problem 3
|          |      |- Problem 4
|          |      |- Problem 5
```

6. **PLEASE do not submit huge files**
7. Team members are expected to help each other but not do work of others.
8. **All team members must understand the details** of all solutions and be able to explain them
9. TA can ask any team member about any of the programs developed and its code**.**

Marking Criterion

Option 1:

1. 1 x 3 1 mark for each **original correct** problem solution (problems 1, 2 and 3)
2. 2 x 2 2 marks for each **original correct** problem solution (problems 4 and 5)
3. -1 for failing to explain the code of other students
4. -1 for low quality code with bad style.
5. -7 for submitting non-original code.