

## ปฏิบัติการที่ 1: การเขียนโปรแกรม MIPS assembly ขั้นพื้นฐาน

ดาวน์โหลด zip ไฟล์ที่มากับปฏิบัติการนี้เพื่อลองรันโปรแกรมตัวอย่าง

ลองใช้ QtSpim รันโปรแกรมต่อไปนี้

- Sum.asm
- FibLT100.asm
- gather\_scatter.asm

เมื่อรันเสร็จสิ้นแล้วดูเหมือนจะไม่มีอะไรเกิดขึ้น นั่นเป็นเพราะโปรแกรมไม่ได้ตอบโต้กับ console ที่แสดงผล ให้นิสิตอ่านโค้ดและดูว่าผลลัพธ์สุดท้ายไปอยู่ที่ใดใน register และ memory และเราจะดูได้อย่างไร นอกจากนั้นให้นิสิตศึกษาวิธีจองที่ใน .data section และการสร้าง array ที่มีค่าเริ่มต้นที่บริเวณนี้ด้วย

จากนั้นให้ลองรันโปรแกรม

- hello.asm
- add\_print\_load.asm
- in\_out.asm

ซึ่งเป็นโปรแกรมที่มีการติดต่อกับ console นิสิตเข้าใจถึงการทำงานของโค้ดทั้งสามนี้อย่างถ่องแท้หรือไม่ และบอกกลไกในการจัดการนำผลมาแสดงที่ console ได้หรือไม่

โปรแกรมสามตัวอย่างหลังนี้มีการเรียกใช้งาน OS library ผ่านคำสั่ง syscall โดยสำหรับ QtSpim เราจะเรียก library ตัวไหนและส่ง argument อะไรเข้าไปได้บ้างมีรายละเอียดตามไฟล์ SPIM-OS-Service.pdf ที่แนบมา

นอกจากนั้นโปรแกรมทั้งสามตัวอย่างนี้เริ่มที่จะเรียกชื่อ register โดยไม่ใช่หมายเลขอีกต่อไป เช่น \$8 ถูกเรียกเป็น \$t1 หรือ \$2 ถูกเรียกเป็น \$v0 ณ ตอนนี้เราอาจจะยังไม่เข้าใจในรายละเอียดในการเรียกชื่อแบบใหม่นี้ แต่ในบทเรียนและปฏิบัติการคราวหน้า เราจะมาทำความเข้าใจข้อตกลงในการเรียกชื่อ register ในลักษณะนี้กัน ในปฏิบัติการนี้ขอให้เรียกชื่อ register เป็นตัวเลขตามที่เราได้ปฏิบัติกันมาก่อนแล้วเสียแต่จะเป็นการเรียก OS library ผ่าน syscall

เริ่มทำปฏิบัติการการเขียนโปรแกรม MIPS assembly 3 ข้อตามรายละเอียดด้านล่างนี้ โดยมีกติกาดังต่อไปนี้

- ในทุกๆข้อ\*ห้าม\*เปลี่ยนแปลงโค้ดภาษาซีที่ให้เราโดยเด็ดขาด โปรแกรม MIPS assembly ในทุกข้อต้องแปลมาจากโค้ดภาษาซีต้นแบบเท่านั้น
- โค้ด MIPS assembly จะต้องมีการมี comment กำกับอย่างละเอียด ขอให้ระลึกถึงกฎนี้เสมอไม่ว่าจะในปฏิบัติการหรือในการสอบ ไม่มี comment = ไม่มีคะแนน

### 1. ArrayCopy.asm

จงเขียนโปรแกรมภาษาซีต่อไปนี้ให้เป็นโปรแกรม MIPS assembly ภาษาซีจะไม่มีการเช็ค overflow เพราะฉะนั้นนิสิตจะต้องเลือกชุดคำสั่งให้เหมาะสม กล่าวคือเราจะต้องใช้ unsigned version ของคำสั่ง MIPS เช่นใช้ addu แทนที่จะใช้ add ศึกษาเรื่องความแตกต่างระหว่างสองคำสั่งนี้จากลิงค์ด้านล่างนี้

[https://chortle.ccsu.edu/AssemblyTutorial/Chapter-13/ass13\\_4.html](https://chortle.ccsu.edu/AssemblyTutorial/Chapter-13/ass13_4.html)

```
int source[] = {3, 1, 4, 1, 5, 9, 0};
int dest[10];

int main ( ) {
    int k;
    for (k=0; source[k]!=0; k++) {
        dest[k] = source[k];
    }
    return 0;
}
```

## 2. ArraySum.asm

จงเปลี่ยนโปรแกรมภาษาซีต่อไปนี้ให้เป็นโปรแกรม MIPS assembly ภาษาซีจะไม่มีการเช็ค overflow เพราะฉะนั้น  
นิสิตจะต้องเลือกชุดคำสั่งให้เหมาะสม

```
int a[20] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19};
int b[10] = {0x7fffffff, 0x7ffffffe, 0x7ffffffd, 0x7ffffffc,
0x7ffffffb, 0x7ffffffa, 0x7fffffff9, 0x7ffffff8, 0x7ffffff7,
0x7ffffff6};

int main() {
    int i, sum = 0;
    for (i=0; i<20; i++)
        sum += a[i];
    printf("Sum a = %d\n", sum);
    sum = 0;
    for (i=0; i<10; i++)
        sum += b[i];
    printf("Sum b = %d\n", sum);
}
```

## 3. LinearSorting.asm

เขียนโปรแกรม MIPS assembly ที่ทำการ sort ข้อมูลในเวลา  $O(n)$  โดย  $n$  คือขนาดของข้อมูลที่ต้องการ sort โค้ด  
ด้านล่างนี้ใช้อัลกอริทึม counting sort นิสิตสามารถศึกษารายละเอียดเกี่ยวกับ counting sort ได้จาก

Wikipedia: [http://en.wikipedia.org/wiki/Counting\\_sort](http://en.wikipedia.org/wiki/Counting_sort)

```
int A[8] = {0, 2, 1, 6, 4, 3, 5, 3};
int B[8] = {0, 0, 0, 0, 0, 0, 0, 0};
int C[7] = {0, 0, 0, 0, 0, 0, 0};

int main() {
    int i;
    int n = 8, k = 7;

    for (i = 1; i < n; i++)
        C[A[i]]++;
    for (i = 2; i < k; i++)
        C[i] = C[i] + C[i-1];
    for (i = n - 1; i >= 1; i--) {
```

```

        B[C[A[i]]] = A[i];
        C[A[i]]--;
    }
    printf("A[] = \n");
    for (i=0; i<n; i++)
        printf(" %d", A[i]);
    printf("\n");
    printf("B[] = \n");
    for (i=0; i<n; i++) {
        printf(" %d", B[i]);
    }
    printf("\n");
    return 0;
}

```

การส่งงาน:

นำไฟล์ที่ได้เขียน MIPS assembly แก้ปัญหาทั้งสามข้ออย่างถูกต้องแล้วมาใส่ไว้ในโฟลเดอร์ชื่อ

studentID1\_firstname1\_studentID2\_firstname2\_lab1

โดย studentID และ firstname คือเลขประจำตัวและชื่อแรกของสมาชิกที่ทำปฏิบัติการร่วมกัน จากนั้น zip โฟลเดอร์นี้แล้วส่ง zip ไฟล์มาที่ Google Classroom ของวิชาก่อนกำหนดส่ง