

Introduction to Computer Organization (ICO, CCIT 4026)

HKU SPACE Community College

AY2526 S2

(Assignment #2)

Instructions:

1. Fill in the information

		(Insert a photo of your student ID card below)
Student Name:		(No mark would be given if no student ID is presented here)
Student ID:		
Class Number:		

2. **TYPE** all your answers on the space provided with the screen captures of your final assembly source codes (with colors) and the input/output messages on this WORD file, then convert to PDF format for submission.
3. Make sure the source codes are readable and neatly formatted. Unclear, incorrectly-sized or codes of poor readability would **NOT** be graded.
4. Add adequate amount of program remarks to improve readability.
5. All problems are expected to be answered by using the **results/instructions** mentioned in the course materials ONLY. If you have to bring in results/instructions out of this scope, prior consent from the lecturer should be sought and a proper citation has to be provided otherwise the answers would be considered as **INCORRECT**.
6. Where applicable, it is okay to run (and single-step) your codes on MARS simulator to confirm its functionality before submission.
7. No late submission will be accepted.
8. Please read the Student Handbook for issues related to academic honesty. The College places very high importance on honesty in academic work submitted by students, and adopts a policy of zero tolerance on cheating and plagiarism. Any related offense may be passed to the Guidance Committee. Possible disciplinary action, including termination of study, may follow.
9. By making a submission, you also confirm to have read and understood the above-listed instructions and HKU SPACE Community College's academic honesty policy, and agree to comply with them. Failure to comply with the above instructions may lead to disqualification of your assignment results and possible disciplinary actions.

Q1. [25 marks]

Suppose \$s0 stores the base address of a word array **A** and \$s1 is associated with **k**, convert the following Python code segment into MIPS instructions.

Notes:

- **NO** Pseudo instruction is allowed,
- \$s0 should not be changed after the instructions are executed,
- No data segment definition is required and
- **No** exit program is necessary.

```

If (k <= 10):
    A[30 + k] = A[30] + k
else:
    k = k + 1
  
```

Answer:

(Copy and paste the screen capture your source code from MARS editor below)

Q2. [25 marks]

Suppose \$t0 stores the base address of word array **A** and \$s0 is associated with *b*, convert the following Python code segment into MIPS instructions.

Notes:

- **NO** Pseudo instruction is allowed,
- No data segment definition is required and
- **No** exit program is necessary.)

```

b = 0
while (b <= 5):
    A[b - 1] = A[b]*7
    b = b + 1
  
```

Answer:

(Copy and paste the screen capture your source code from MARS editor below)

Q3. [25 marks]

Suppose \$t2 stores the base address of word array A and \$s3 is associated with h , convert the following Python code segment into MIPS instructions.

Notes:

- **NO** Pseudo instruction is allowed,
- \$t2 should not be changed after the instructions are executed,
- No data segment definition is required and
- **No** exit program is necessary.

```
for h in range(3, 10, 1):
```

```
    if (h < 5):
```

```
        A[h] = A[h//2]
```

```
    else:
```

```
        A[h] = A[h*2]
```

Answer:

(Copy and paste the screen capture your source code from MARS editor below)

Q4. [25 marks]

Write the following Python code function in MIPS instructions according to the following conditions:

- **Assuming the following registers CAN ONLY BE USED:**
 - **\$s0, \$s1 and \$v0 can both READ/WRITE,**
 - **\$a0 and \$a1 register can ONLY be READ and**
 - **\$sp can be READ/WRITE BUT for stack operation ONLY.**
- **ALL arithmetic and return value are in 32-bit.**
- **Procedure calling convention and callee-registers save/restore MUST be followed.**
- **NO Pseudo Instruction is allowed.**
- **No exit program is necessary.**

```
def my_func(x, y):
    a = y // x
    b = 1
    while (a < 5):
        b = x - a
        a = a + 1
    return (b - y)
```

Answer:

(Copy and paste the screen capture your source code from MARS editor below)

----- END OF ASSIGNMENT 2 -----