EE213 Computer Organization and Assembly Language
Assignment II – Fall 2019
**Open Date**: Friday 22nd November, 2019 (1200HRS)
**Due Date**: Friday 29th November, 2019 (1600HRS)

**NOTE: Students must submit handwritten solution for this activity.**

1. Discuss state of practices of CISC and RISC architectures, your discussion must include assembly language support, registers, architecture styles, practical examples, and applications.

2. Discuss RISC-V architecture in detail.

3. What are different addressing modes in MIPS Assembly? Elaborate with example.

4. What FLAGS are available in MIPS, what are their roles?

5. Draw out data segment, code segment, and stack segment after converting the following program into X86 Assembly Language. Before Drawing code segment, first encode x86 instructions of your program.

```
int number;
int main() {
       int n1=0,n2=1,n3;
       cout<<"Enter the number of elements: ";
       cin>>number;
       cout<<n1<<" "<<n2<<" ";
       for(i=2;i<number;++i)
       {
              n3=n1+n2;
              cout<<n3<<" ";
              n1=n2;
              n2=n3;
       }
   return 0;
   }
```

6. What is the role of INSTRUCTION PREFIX BYTE and SCALE INDEX BYTE when encoding X86 instructions? Elaborate with examples.

7. Encode the given X86 Instructions, your solution must include all possible bytes.
   a. `INC  DWORD PTR [VAR1+1000h]`
   b. `IMUL  BX`
   c. `CMP  CX, 100h`
   d. `SUB  WORD PTR [ESI+EDI+1000h]`
   e. `XCHG  ESI, EDI`
   f. `MOV  [EBX+ESI*4], EDX`

8. Identify the hazards (both structural and data hazards) in the following MIPS code, resolve the problems, and finally draw out the escheduled instructions in their encoded formats:

```
lw        $10, ($1)
lw        $11, ($2)
add       $12, $10, $11
```

```
sw      $12, ($3)
lw      $13, ($4)
lw      $14, ($5)
sub     $15, $13, $14
sw      $15, ($6)
lw      $10, 0($1)
lw      $11, 0($2)
lw      $13, 0($4)
lw      $14, 0($5)
add     $12, $10, $11
sw      $12, 0($3)
sub     $15, $13, $1
```

**********