COMPUTER ORGANIZATION AND ARCHITECTURE

Course Code	: CSE 2151
Credits	: 04

INSTRUCTION EXECUTION AND STRAIGHT-LINE SEQUENCING

- a possible program segment for the task, C=A+B, as it appears in the memory of a computer.
- Straight-line sequencing: using the information in the PC to fetch and execute instructions, one at a time, in the order of increasing addresses.
- Executing a given instruction is a two-phase procedure.
  - Instruction fetch (First phase)
  - Instruction execute (Second phase)

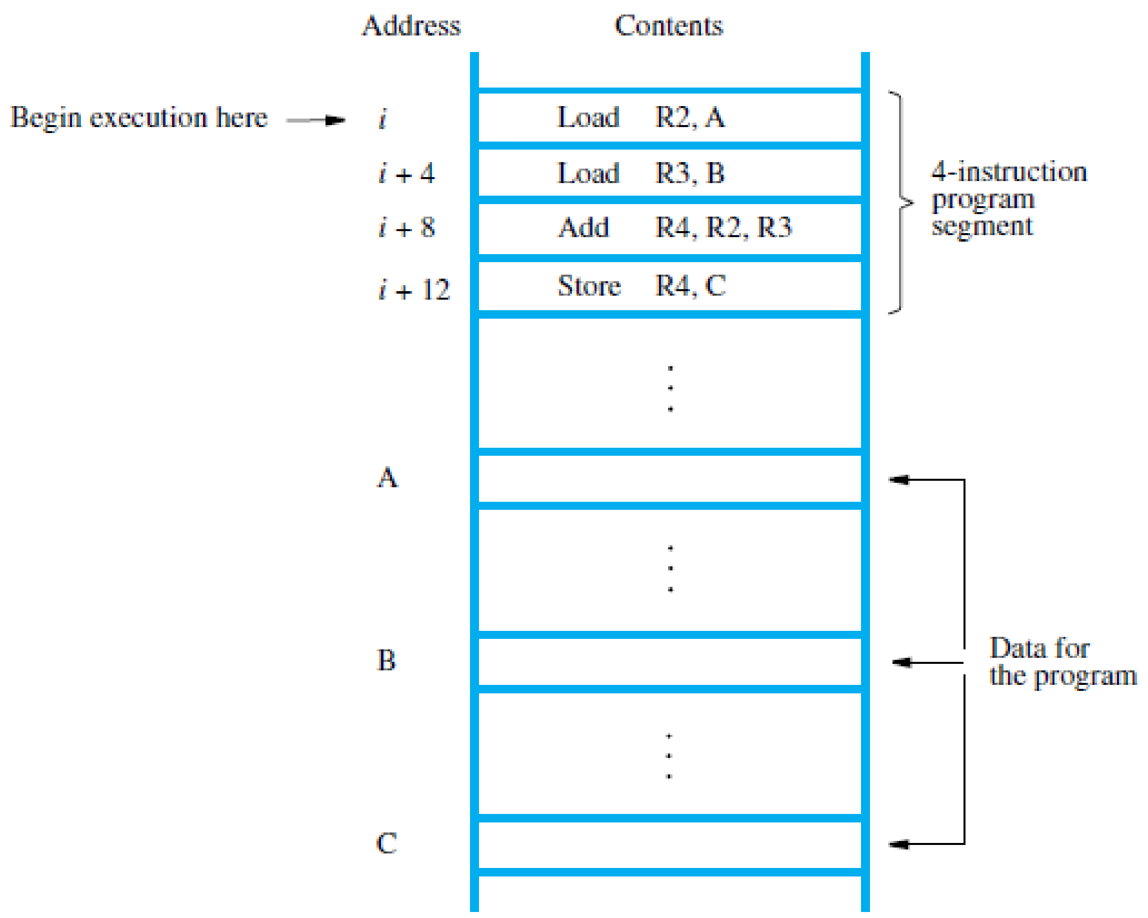

Figure 2.4 A program for C ← [A] + [B].

BRANCHING

- Consider the task of adding a list of n numbers.

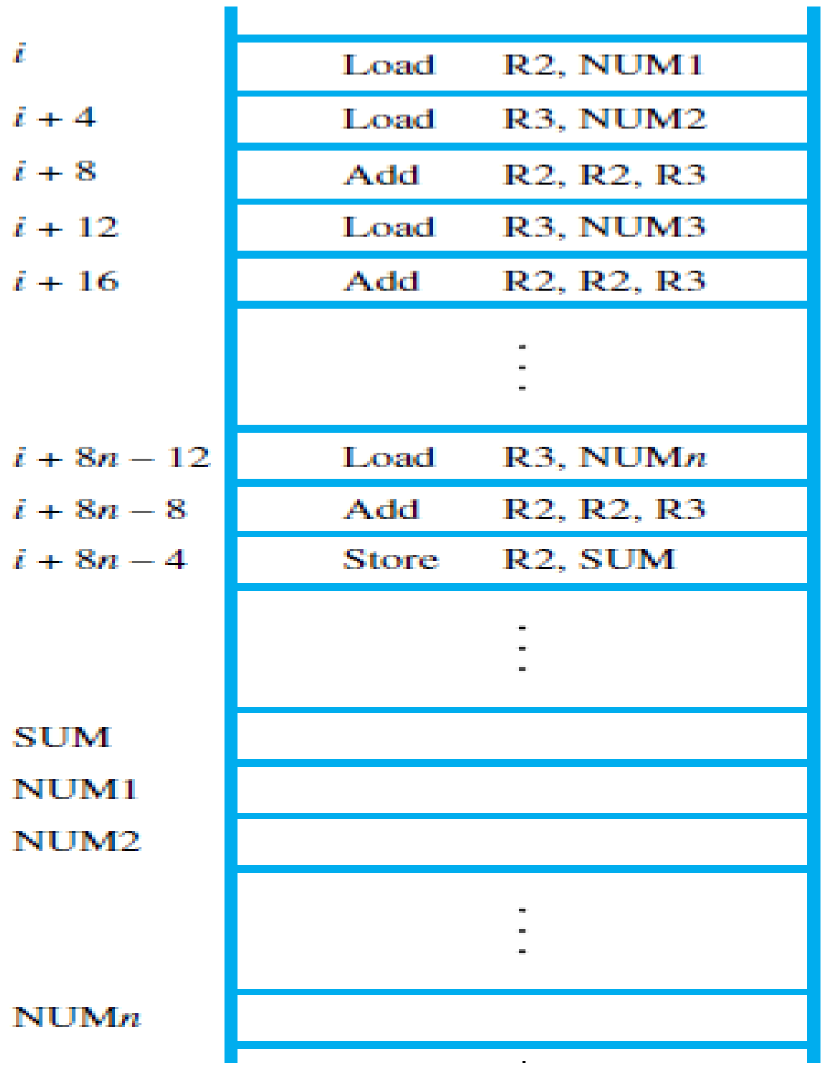

Figure 2.5 A program for adding n numbers.

BRANCHING

- Consider the task of adding a list of n numbers in a loop.

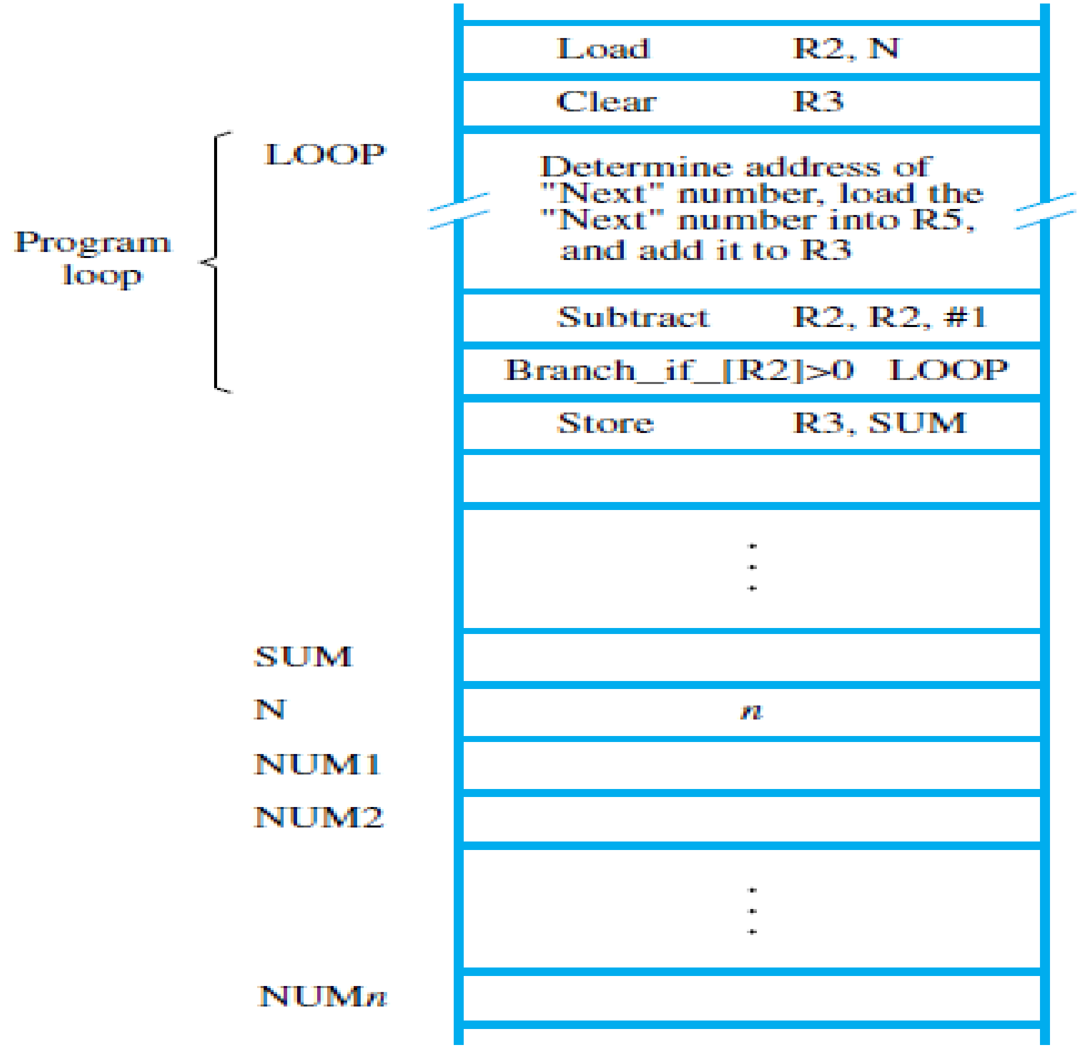

Figure 2.6 Using a loop to add n numbers.

BRANCHING

- Branch_if_[R4]>[R5] LOOP
  - In generic assembly language as: Branch_greater_than R4, R5, LOOP
  - Using an actual mnemonic as: BGT R4, R5, LOOP

GENERATING MEMORY ADDRESSES

- The purpose of the instruction block starting at LOOP is to add successive numbers from the list during each pass through the loop
- must refer to a different address during each pass
- memory operand address cannot be given directly in a single Load instruction in the loop.

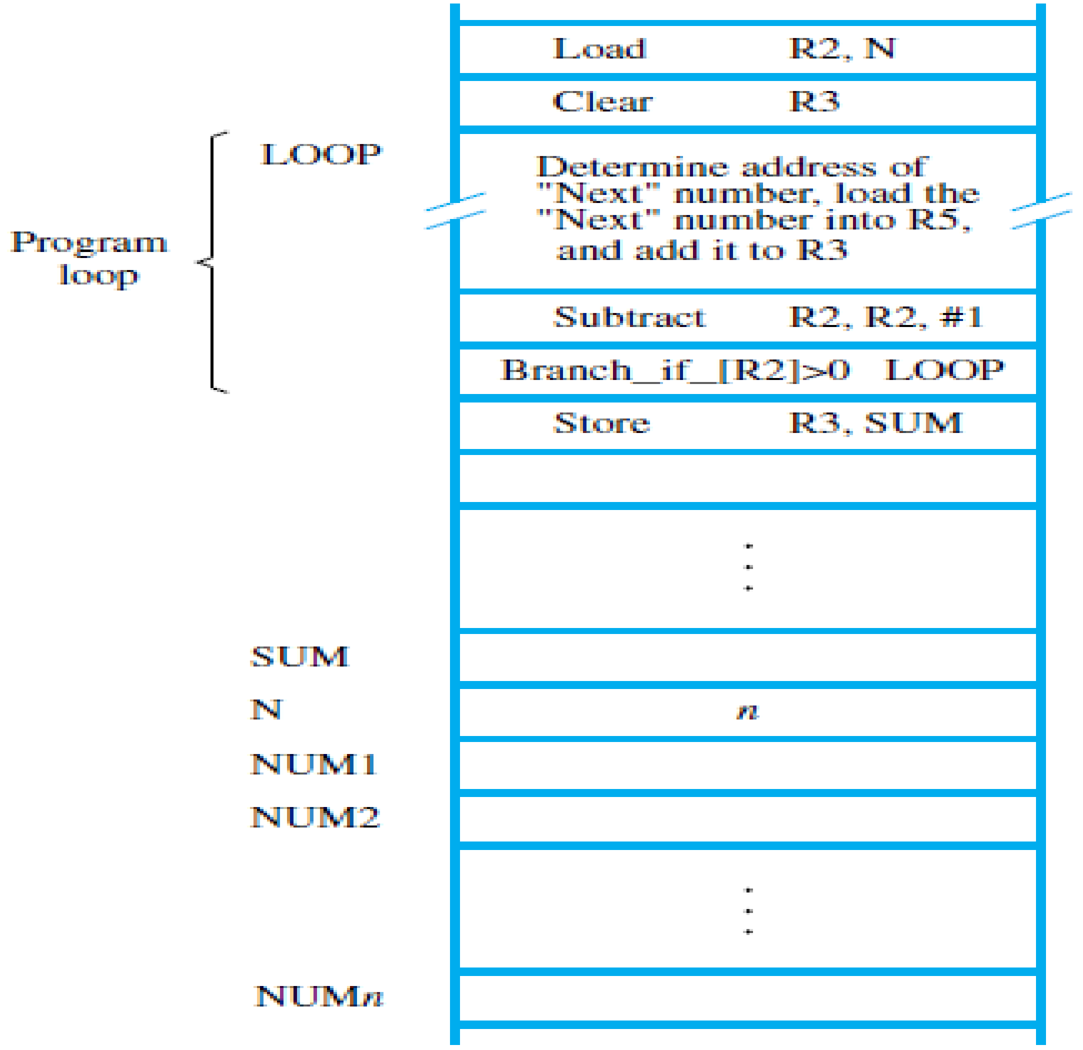

Figure 2.6 Using a loop to add n numbers.

ADDRESSING MODES

- Addressing modes:
  - Different ways for specifying the locations of instruction operands.
- RISC-style processors basic addressing modes Table 2.1
- The assembler syntax defines the way in which instructions and the addressing modes of their operands are specified

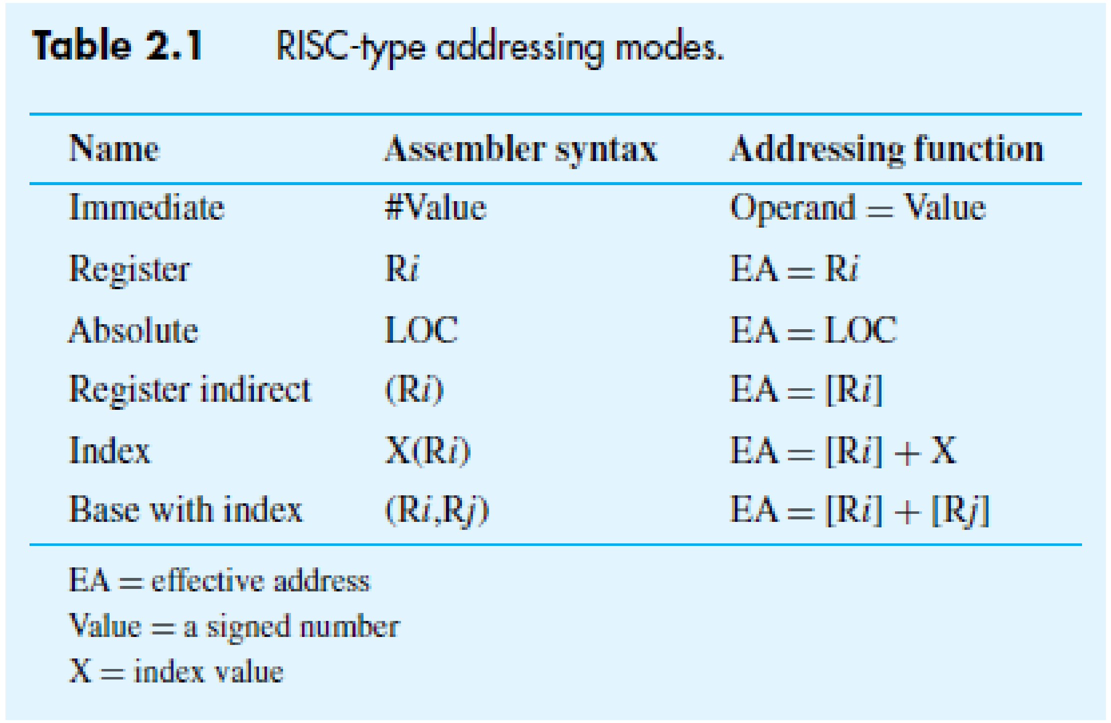

Table 2.1 RISC-type addressing modes.

Name	Assembler syntax	Addressing function
Immediate	#Value	Operand = Value
Register	R*i*	EA = R*i*
Absolute	LOC	EA = LOC
Register indirect	(R*i*)	EA = [R*i*]
Index	X(R*i*)	EA = [R*i*] + X
Base with index	(R*i*,R*j*)	EA = [R*i*] + [R*j*]

EA = effective address
Value = a signed number
X = index value

IMPLEMENTATION OF VARIABLES AND CONSTANTS

- Register mode
  - The operand is the contents of a processor register; the name of the register is given in the instruction.
  - Example: Add R4, R2, R3
- Absolute mode
  - The operand is in a memory location; the address of this location is given explicitly in the instruction.
  - Load R2, NUM1
- Immediate mode
  - The operand is given explicitly in the instruction.
  - Add R4, R6, $200_{\text{immediate}}$
  - Add R4, R6, #200

INDIRECTION AND POINTERS

- Indirect mode:
  - The effective address of the operand is the contents of a register that is specified in the instruction
  - Load R2, (R5)

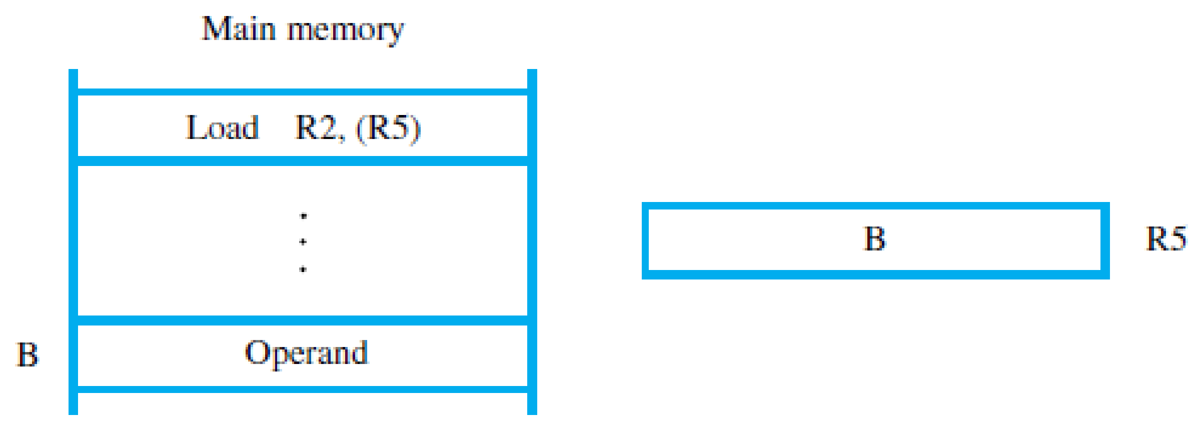

Figure 2.7 Register indirect addressing.

TOPICS COVERED FROM

- Textbook 1:
  - Chapter 2: 2.3, 2.4