16-bit CPU

- This is a 16-bit processor. In particular, all registers are 16-bit, and each instruction is also 16-bit.
- The processor has 8 general-purpose registers $r1 – $r8.
- The data and instruction are stored in two separated memory, each having its own address space. Both the data and instruction memory are addressed by 16-bits, i.e., the size of the memory block at each address is 16-bits. (Notice, this is different from the examples in our lecture, where each memory block is 8-bit.)
- The processor supports the following instructions:

The format of each instruction:

li	0000 - opcode ori rt, $zero, 10 pseudo 0000
add	R-type; format: add $rd, $rs, $rt \| R[$rd] ← R[$rs] + R[$rt]; 0001 - opcode 0001 xxxx(rs) xxxx(rt) xxxx(rd)
and	R-type; and $rd, $rs, $rt \| R[$rd] ← R[$rs] & R[$rt]; 0010 - opcode 0010 xxxx(rs) xxxx(rt) xxxx(rd)
or	R-type; or $rd, $rs, $rt \| R[$rd] ← R[$rs] \| R[$rt]; 0011 - opcode 0011 xxxx(rs) xxxx(rt) xxxx(rd)
load	0100 – opcode Memory to reg 0001 rs rt

store	0101 - opcode Reg to memory
move	0110 – opcode add $rt, $zero, $rs pseudo
addi	0111 - opcode
andi	1000 - opcode
ori	1001 - opcode
ble	Branch if less or equal; 1010 – opcode pseudo
bne	Branch if not equal; 1011 - opcode

jump	1100 - opcode
call	1101 - opcode
rtn	1110 - opcode
halt	1111 opcode