MOV/MVN	LDR	ADR
ARM instruction	Pseudo instruction to load all possible 32-bit constant into register.	Pseudo instruction to load 32-bit address into register
Only 8 bits for immediate data with 4 bits rotate values (x2)	It uses program-relative addressing mode	It uses program-relative addressing mode to jump to the location within the program
Cannot represent all representation of 32 bits numbers; e.g. 0x12341355	It creates a 32-bit constant in a literal pool (usually) at the end of the code, if the value is not within the range of MOV/MVN	< 1020 bytes away for aligned word
	In other words, it requires 1 MOV instruction and 1 32-bit data. (8 bytes)	Always assemble to 1 instruction
	Always work for any numbers. But literal pool must be within 4 KB away from its reference (PC). Otherwise, use LTORG directive to relocate literal pool.	No literal pool constant created. Faster than LDR.