

---

## Intro to Computer operations and their operands

---

- **Computer Instructions**

- **Computer instructions** can **perform** the following **operations**:

- **Arithmetic operations** (e.g.: add, subtract, multiply, divide)
  - **Compare** 2 values
  - **Logical operations** (e.g.: and, or, not)
  - **Transfer** data
  - **Branching/jumping** to some **location** in the program
- 
- 
- 

- **Operands**

- **Operands**:

- **Operands** = the **values** used in a **computer instruction/operation**
- 
- 

- A **M68000 instruction** can have:

- **No operands**
  - **1 operand**
  - **2 operands**
- 
- 

- **Some CPUs** has **machine instructions** that takes **3 operands**
- 
- 
- 

- **Types of operands**

- **All computer instructions** uses the following **3 types** of **operands**:

- **Constants**:

- **Constant** = a **fixed (constant) value**

E.g.: 4, -6, 'A'

- **Register operands:**

- **Register operand** = the **value** stored in a **register** of the **CPU**

- **Memory operands:**

- **Memory operand** = a **value** stored in the **computer memory**

- **Size of operands**

- **Fact:**

- **Machine instructions** can **operate** on **operands** of **different sizes**:

- **Operands** of **8 bits**
- **Operands** of **16 bits**
- **Operands** of **32 bits**

- **Very important fact:**

- **Machine instructions** can **only** **operate** on **source operands** that are the **same size** !!!

**Example:**

- The **CPU** can **only** **add**:

- **2 bytes** together, or
- **2 shorts** or
- **2 ints**

- The **CPU** **cannot** add:

- a **byte** and a **short**, or
- a **byte** and a **int**, or

- a **short** and a **int**,
- And so on.

