



Variables, DataTypes, Arithmetic Operators and Expression



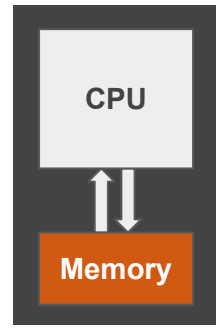
اَللّٰهُمَّ ارْزُقْنِيْ عِلْمًا نَّافِعًا وَاسِعًا عَمِيْقًا

اَللّٰهُمَّ ارْزُقْنِيْ رِزْقًا وَّاسِعًا حَالًا لَا طَيِّبًا
مُّبَارَكًا مِنْ عِنْدِكَ

Memory and Variables

Review: Memory

- When CPU takes input from devices, it stores information into **memory** before processing it.
- CPU stores results of the processing into the **memory**.
- CPU stores information into the **memory** before sending it to output devices.



How to Allocate Memory: Variables

To store data into the **Memory**, we need to reserve the space in the **Memory**.

When the space is reserved, we can store or retrieve data from the **Memory** through its **Memory Addresses**.



30	0xE4A71
5	0xE4A72
6	0xE4A73

Memory

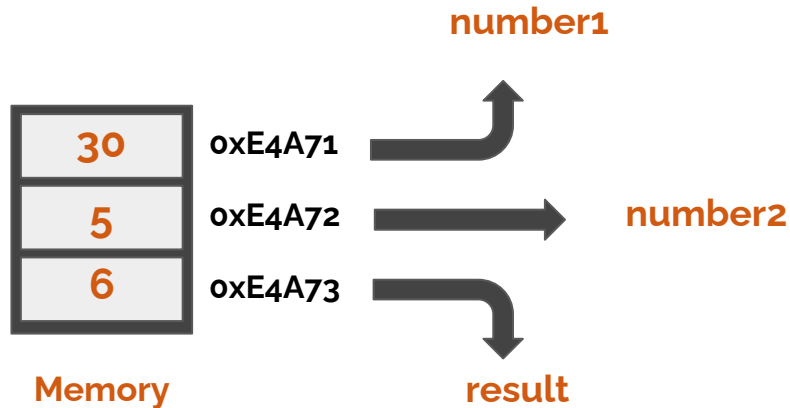


Variable Name

Variables: Names

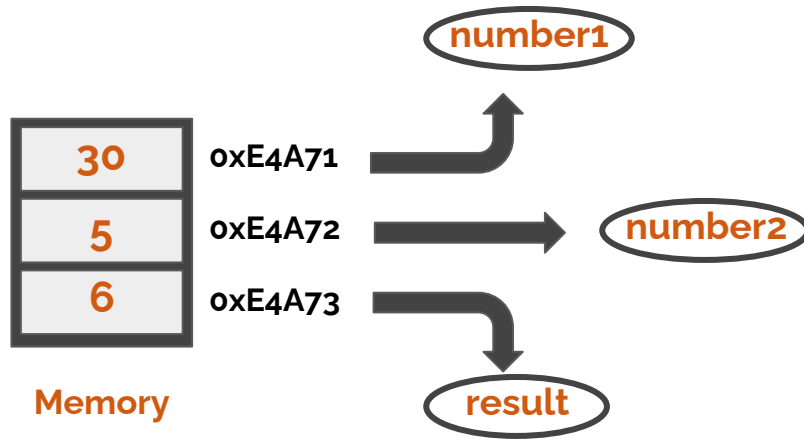
It is difficult to remember the **Addresses** of these **Memory** locations.

High Level Languages allow us to give **Names** to these reserved **Memory** locations.



Variables: Names

These **Names** are called the **Variables**.
Variables are the names that we give to the
Memory Locations.



Variables: Names

All **High Level Languages** apply some **Naming Rules** on the variables

- The name can **not** have **Spaces**
- The name can **not** start with **Numbers**
- The name can **not** have any **Special Character** (&, !, %, # etc)



Variables: Names

All **High Level Languages** apply some **Naming Rules** on the variables

These are some of the **Valid** names of the **Variables**



number1

num_1

num1

numb2

nu_2

_n2

result_1

Res

_Res

numOne



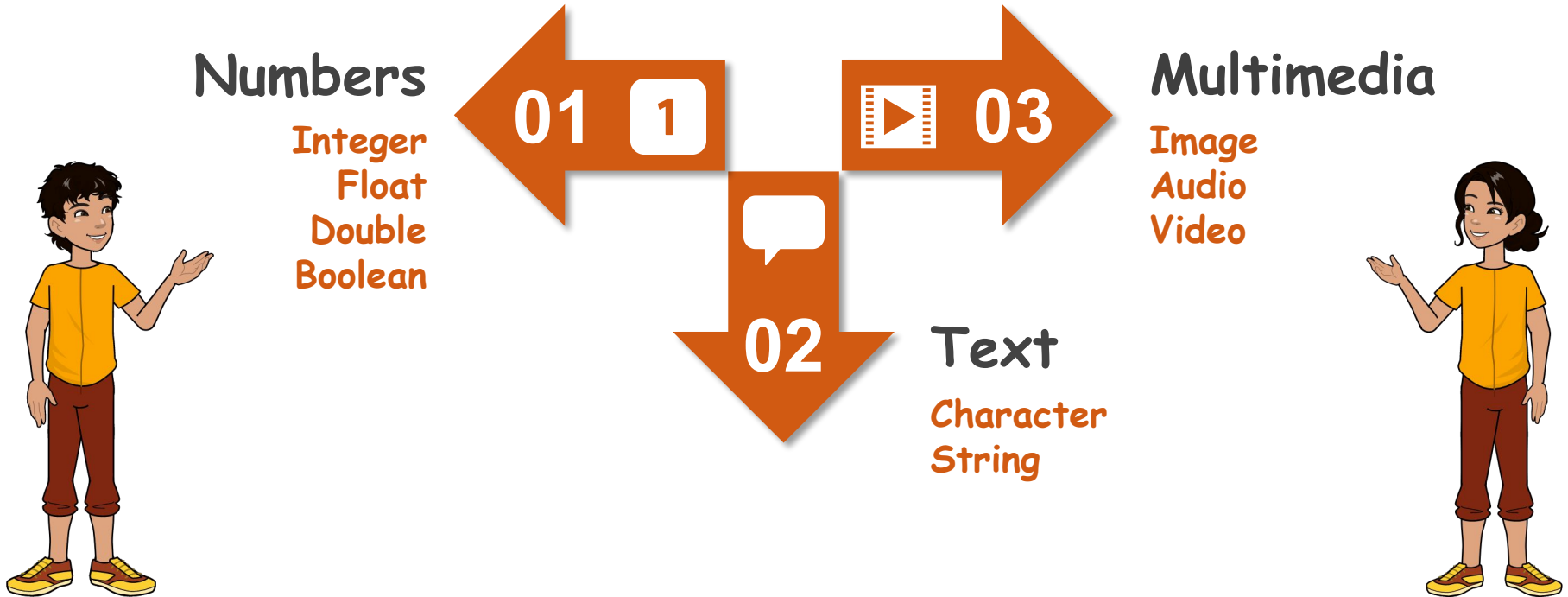
Variable Datatypes

What Type of Data in Memory?

Now, We know that we can deal with memory using **Variables**. But the question is **What type of Data** is in memory?



Variables: Types of Data

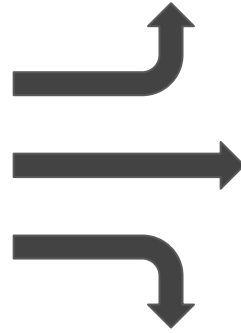


Variables: What Kind of Data Inside



Memory

Integer
number1

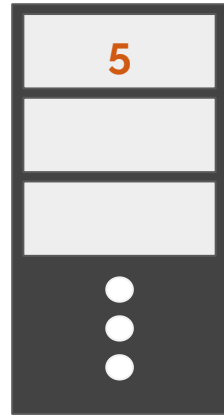


Float
number2

text
String



Variables: What Kind of Data Inside



0xE4A71

0xE4A72

0xE4A73

Memory

number1



Data Types: Why Inform Memory



|| Data Types: Why **Inform** Memory

- To **Adjust Size** of Allocated Memory Cell
- To **Check** the Validity of the Operations



Data Types: **Size** of Memory

Different types of data require **Different sizes** of cells in memory.



Memory

0xE4A71

0xE4A72

0xE4A73

0xE4A74

0xE4A75



Data Types: **Validity** of Operations

We also need to **Check** whether an Operation applied on the data is **Valid** or **Not**.

For Example:

- $20 + 20.5$

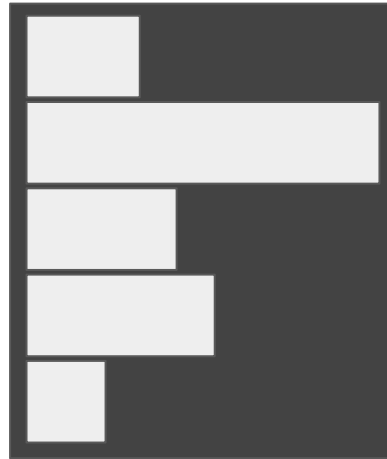


- $20 + \text{Programming}$



Variable Declaration: Reserve Memory

Reserving the memory location through Variables for certain type of data is also called Variable Declaration.



0xE4A71

0xE4A72

0xE4A73

0xE4A74

0xE4A75

Memory



Variable Declaration: Reserve Memory

In many High Level Languages like **C++**, **Java** and **C#** the variable declaration is done as

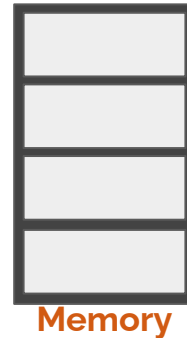
Datatype nameOfTheVariable;

int a;

char letter;

string word;

float decimal;



a
letter
word
decimal

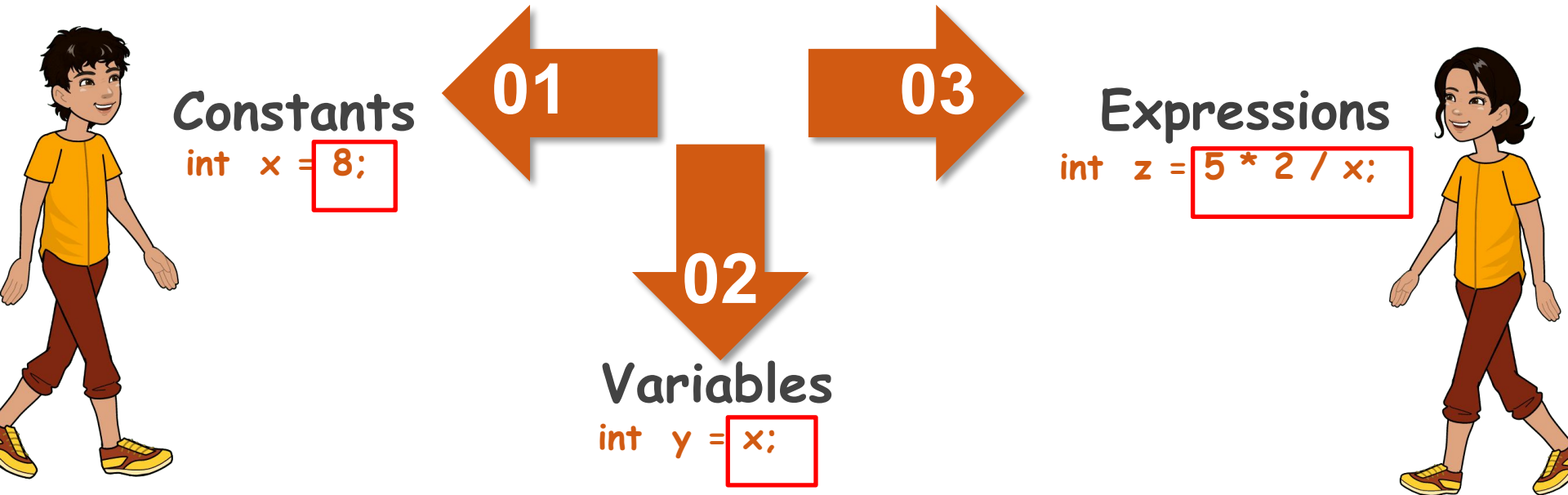


|| Uses of Variables

- **assign** values to these variables according to their
- data types
- **retrieve** values from these variables
- We can apply different **mathematical**
addition, multiplication, subtraction) and other
operations (we will see those in next lecture) on these
variables.

Uses of Variables: Assignment

We can **Assign** a value to variable using **Assignment Operator**.



Uses of Variables: Retrieval

Here, we are **Retrieving** the value of variable `x` and assigning that value to variable `y`

```
int X = 3;
```

```
int y = x;
```



Memory



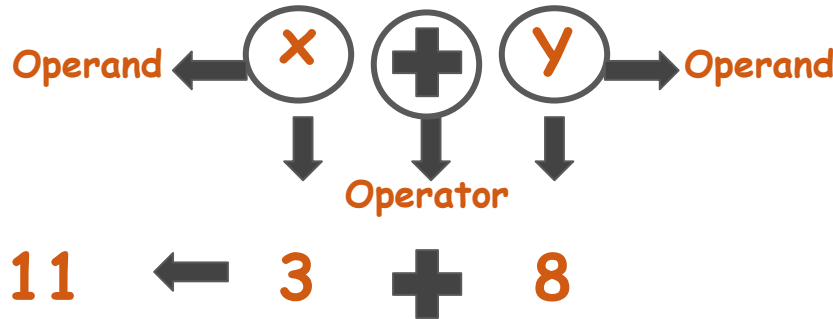
Operations on Variables: Addition

Apply mathematical operation on Variables

```
int x = 3;  
int y = 8;  
int z = x + y;
```

3	x
8	y
11	z

Memory



Arithmetic Operators:

Here is a list of **Arithmetic Operators** that can be used.



Operator	Meaning	Example
+	Addition	$8+2=10$
-	Subtraction	$8-2=6$
*	Multiplication	$8*2=16$
/	Division	$8/2=4$
%	Modulus	$8\%2=0$



Arithmetic Operators:

These operators also called **Binary Operators**.
Why ?

Operator	Meaning	Example
+	Addition	$8+2=10$
-	Subtraction	$8-2=6$
*	Multiplication	$8*2=16$
/	Division	$8/2=4$
%	Modulus	$8\%2=0$



Expressions

An **Expression** is a combination of **Variables**, **Constants** and **Operators**.

For Example

- $8 + 9$ is an expression
- $X/2 - 1$ is also an expression



Expressions

It consists of

- One or more Operands
- Zero or more Operators

For Example:

$X + 10 - Y$



Operands



Operations on Variables: Expression

Expression containing only Constants and Operators

$$2 + 10 + 8$$



Operations on Variables: Expression

Expression containing combination of Variables, Constants and Operators



$$27 \leftarrow \underset{\substack{\downarrow \\ 5}}{\textcircled{X}} + 10 + \underset{\substack{\downarrow \\ 12}}{\textcircled{Y}}$$



Expressions

We can write Expression using **Variables** and **Constants** and **Assign** these **Expressions** to some **Variables**.



int x = 3;

int y = 8;

int z = $y - x + 10$;



Memory

x
y
z



Expressions

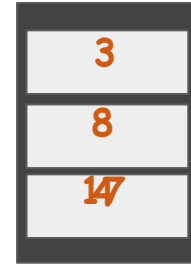
Lets see some more examples



int x = 3;

int y = 8;

int z = 2 * 25 + 7;



x
y
z



Operations on Variables: Expression

$$Z = 10 + 10 * 5$$



Operations on Variables: Expression

$$60 = 10 + 50$$

$$Z = 10 + 10 * 5$$

$$100 = 20 * 5$$



Operations on Variables: Expression

$$Z = 60$$


$$Z = 10 + 10 * 5$$

~~$$Z = 100$$~~



Operations on Variables: Expression

$$Z = 60$$



$$Z = (10 + 10) * 5 \leftarrow$$

$$Z = 1000 * 5 \swarrow \searrow$$



Expression : Precedence Order

Here is the precedence order of **Arithmetic Operators**



Operator	Symbol	Precedence
Parentheses	()	1
Exponential	X^Y	2
Multiplication	*	3
Division	/	3
Addition	+	4
Subtraction	-	4

Expression: PEMDAS RULE

Simply, we can Remember the order of precedence through the PEMDAS Rule.



Rule	Operator	Symbol
P	Parentheses	()
E	Exponential	x^y
M	Multiplication	*
D	Division	/
A	Addition	+
S	Subtraction	-



Working Examples: Expressions

Lets see some working examples of Expressions



$$Z = 2 + 3 / 4$$



The diagram shows an orange arrow pointing from the division operator in the expression above to the division operator in the expression below. A checkmark is placed above the division operator in the second expression.

$$Z = 2275 \ 0.75$$



Working Examples: Expressions

Lets see some working examples of Expressions



$$Z = 10 - 2 * 4$$


$$Z = 210 - 8$$



Working Examples: Expressions

Lets see some working examples of Expressions

$$Z = 4 + 100 / 20 * 2$$



Working Examples: Expressions

How to solve?

$$Z = 4 + 100 / 20 * 2$$



Working Examples: Expressions

When there are operators with **same precedence order** then we solve the expression from **left to right** (Associativity)

$$Z = 4 + 100 / 20 * 2$$



Working Examples: Expressions

When there are operators with **same precedence order** then we solve the expression from **left to right** (Associativity)

$$Z = 4 + 100 / 20 * 2$$



Working Examples: Expressions

When there are operators with **same precedence order** then we solve the expression from **left to right** (Associativity)

$$Z = 4 + 100 / 20 * 2$$



$$Z = 4 + 5 * 2$$



Working Examples: Expressions

When there are operators with **same precedence order** then we solve the expression from **left to right** (Associativity)

$$Z = 4 + 100 / 20 * 2$$



$$Z = 4 + 5 * 2$$




$$Z = 4 + 10$$



Working Examples: Expressions

When there are operators with **same precedence order** then we solve the expression from **left to right** (Associativity)

$$Z = 4 + 100 / 20 * 2$$


$$Z = 14$$



Learning Objective

Write expression using **Arithmetic Operator, Variables, and constants** while following precedence rule.



Self Assessment

1. What is a **Variable**?
2. How we can store and load data from the **Memory** using variables?
3. From the given table below, tell which **Variable Names** are **Valid** and which are not.

Variable	Valid/Invalid
mul*	
Foo	
Do it	



Self Assessment

4. Define **Variable Declaration**. And **Declare** a variable to store a value of **58.9**
5. Write the **Datatypes** of the following data given in the table

Data	Datatype
400.6	
My name is Kaka	
C	
12	

6. Declare the variables to store the above mentioned data in the variables.

Hint: **float a;** (**a** is a **variable** that will store **float** type of data)



Self Assessment

1. Find **constant**, **variable** and **operator** from the following statements

Statement	Constant	Variable	Operator
Foo = 4 * result			
Var = 5 % 3			
X = num1 - num2			

2. Solve the following **Expressions** and write the answer.

Statement	Answer
Foo = 4 * 10 / 2	
Var = 5 % 3	
X = 5 - 2 + 62 - 2	



Self Assessment

3. **Evaluate** the following expressions and **Write** the answers.

No.	Expression	Answer
1	$2 / 1 + 5$	
2	$3 / 4 + (2 - 1)$	
3	$7 + (600 - 100) * 8$	
4	$500 * 400 / 4 + 10$	
5	$18 / 2 * 18 - 1$	



Conclusion

- We can have multiple uses of variables
 1. Assign Values
 2. Retrieve Values
 3. Apply Mathematical Operations
- Assignment is done using **Assignment Operator**.
- There are 3 ways in which we can assign values to the variables
 1. Constants
 2. Variables
 3. Expressions
- An **Expression** is a combination of **Variables**, **Constants** and **Operators**.
- **Expressions** are evaluated with the **Precedence** order of Operators.
- The precedence order is given by **PEMDAS** Rule.
- If operators are with **same precedence order** then they are solved from **left to right** (Associativity).