Variables & Data Types

CS 104 MSG 1.1

Jackson Eshbaugh

Lafayette College

February 3, 2025

Outline

Outline

Jackson Eshbaugh

- Sophomore
  - Computer Science; French

Jackson Eshbaugh

- Sophomore
  - Computer Science; French
- Interested in utilizing computer science to improve other fields of study/research/work
  - i.e., *bioinformatics* or *computational linguistics*

Jackson Eshbaugh

- Sophomore
  - Computer Science; French
- Aspiring computer science professor, teaching in a holistic way and researching within my wide range of interests.

Jackson Eshbaugh

- Sophomore
  - Computer Science; French
- Aspiring computer science professor, teaching in a holistic way and researching within my wide range of interests.
- `jacksoneshbaugh.github.io`

Outline

What is a Variable?

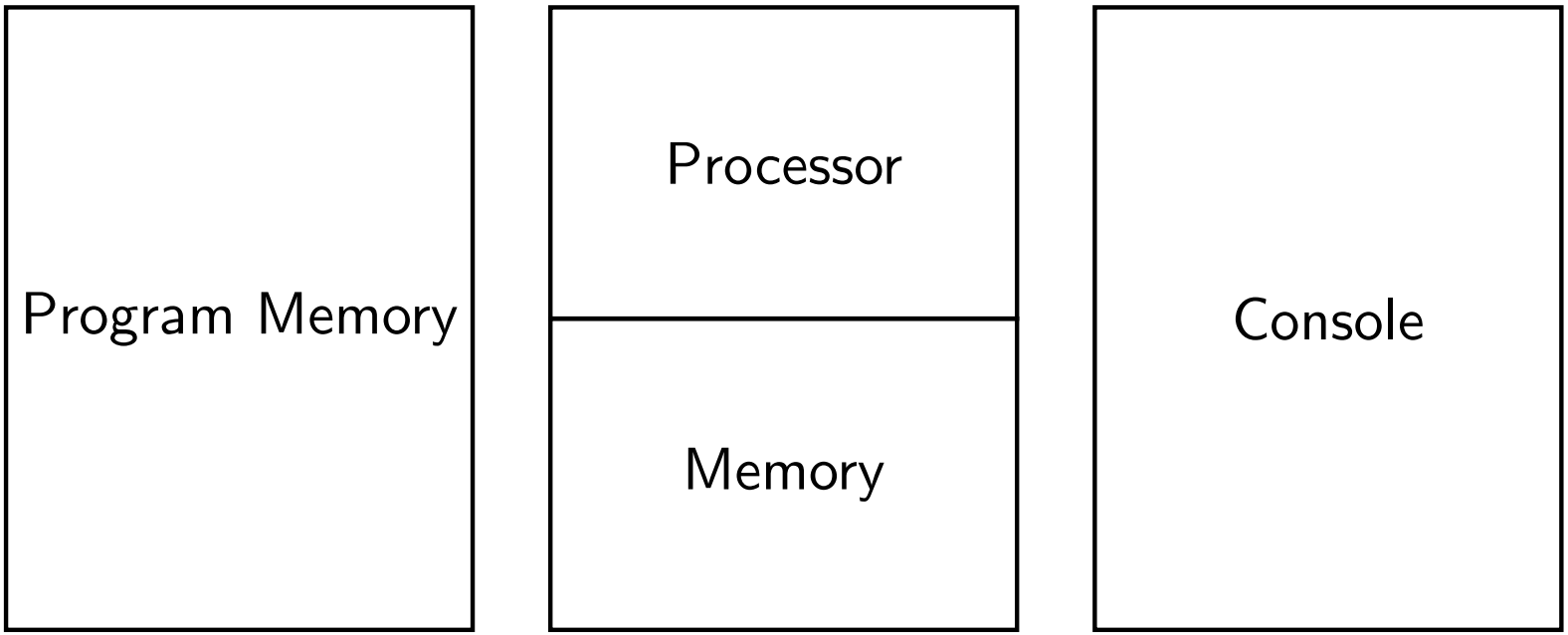
Whiteboard it: *How would you define the term* **variable***?*

What is a Variable?

Whiteboard it: *How would you define the term* **variable***?*

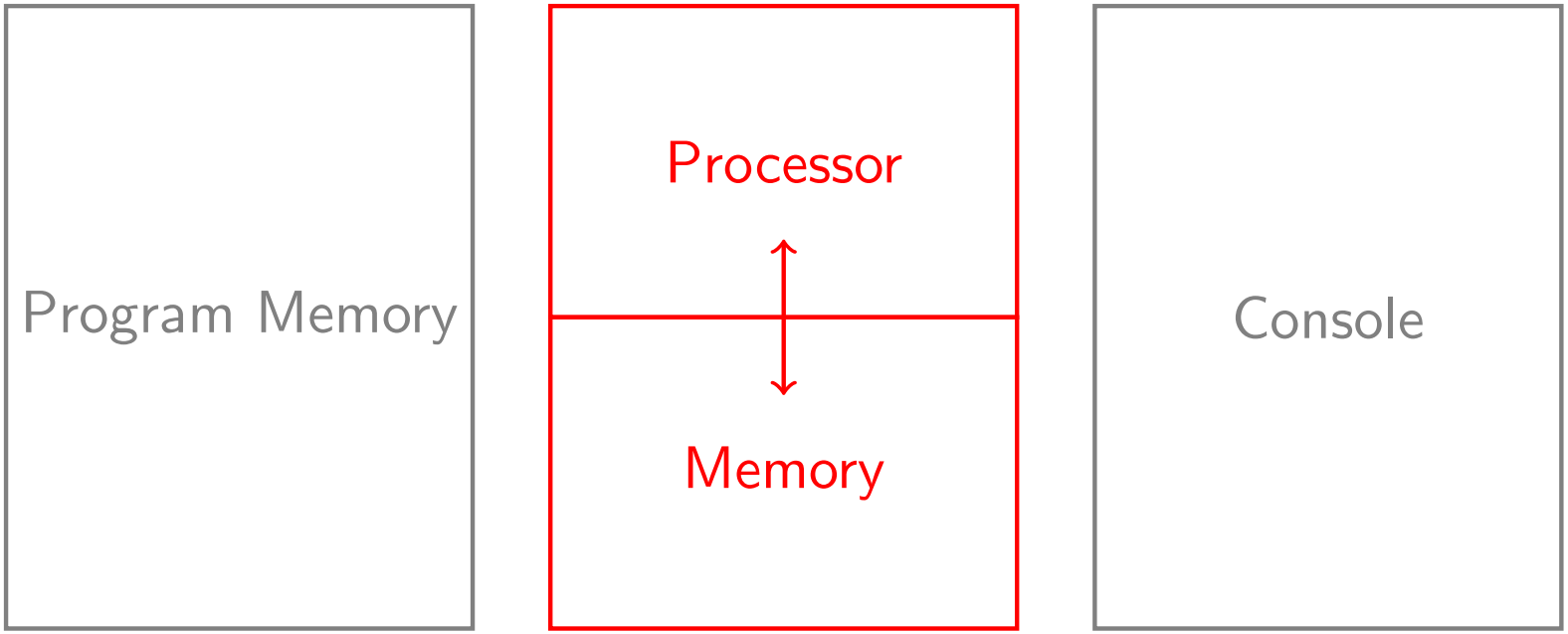
$$x, y, z \ldots$$

Computer Model



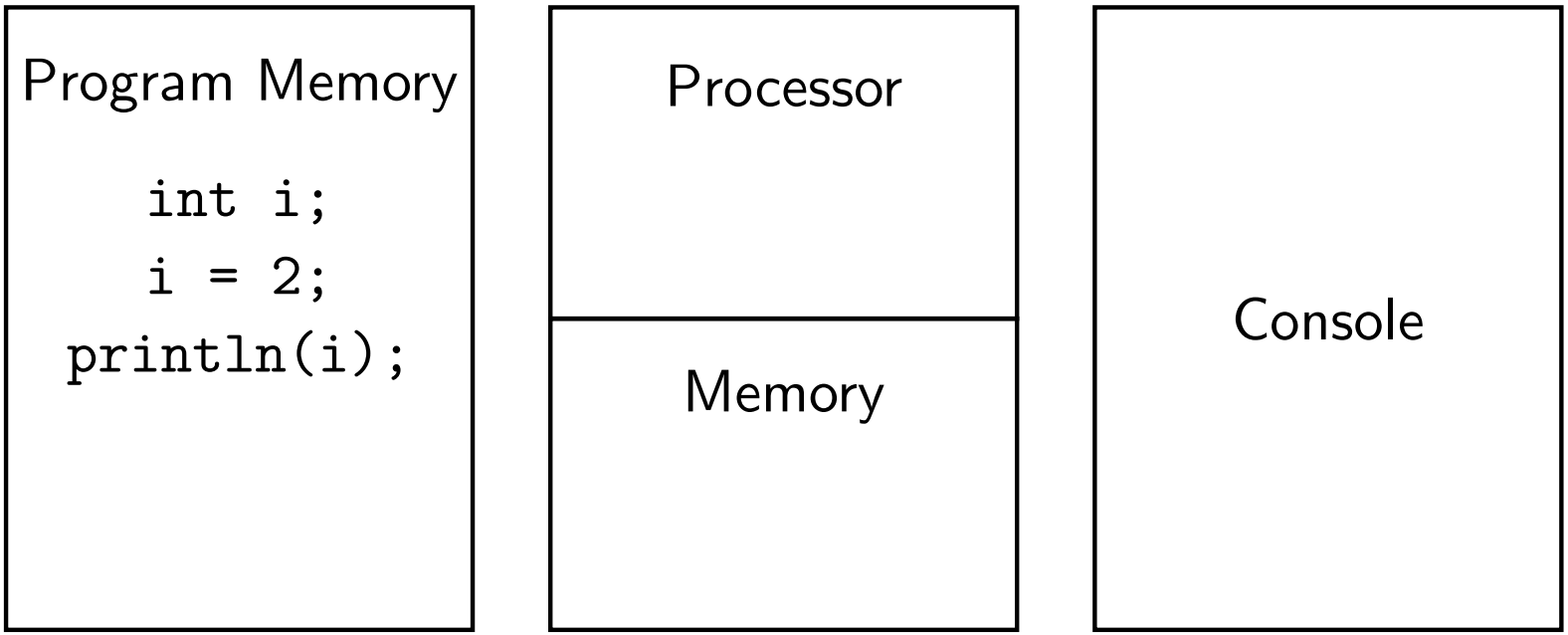
- Recall this model of the computer from class.

Computer Model



- Variables deal with the memory (and the processor).

Variable Example



- Variables are stored in memory.

Data Types

- Computers are mathematical devices at their cores.

Data Types

- Computers are mathematical devices at their cores.
- This implies that all data we encounter must be numerical at its core.

Data Types

- Computers are mathematical devices at their cores.
- This implies that all data we encounter must be numerical at its core.

$$\Rightarrow \textbf{Enter data types.}$$

Outline

Data Types

- Data types define the type of data a variable can hold.
- In other words, the data type of a variable is used to interpret the stored numerical value.

Data Type Examples

Data Type	Represents
`int`	
`float`	
`double`	
`char`	
`boolean`	
`String`	

Table: Data Types and What They Represent

Outline

Operators

- Operators operate on operands—variables or literal numbers.

Operators

- Operators operate on operands—variables or literal numbers.
- + — addition

Operators

- Operators operate on operands—variables or literal numbers.
- + — addition
- - — subtraction

Operators

- Operators operate on operands—variables or literal numbers.
- + — addition
- - — subtraction
- * — multiplication

Operators

- Operators operate on operands—variables or literal numbers.
- + — addition
- - — subtraction
- * — multiplication
- / — division

Operators

- Operators operate on operands—variables or literal numbers.
- + — addition
- - — subtraction
- * — multiplication
- / — division
- % — modulus

Operator Examples

What will the following code output? What will the memory look like after execution?

```
int i = 12;
int j = 7;
println(3 * x);
int k = i + j - 2;
println(k);

int l = 3;


println(l / 2);
```

Operator Examples

What will the following code output? What will the memory look like after execution?

```
int i = 12;
int j = 7;
println(3 * x);
int k = i + j - 2;
println(k);

int l = 3;


println(l / 2);
```

Dangers of Division

- Depending on the data type, `a / b` yields different results.

Dangers of Division

- Depending on the data type, `a / b` yields different results.
- If *both* `a` and `b` are integer or other non-decimal numerical types, `a / b` yields an integer value, **truncating the decimal.**

Dangers of Division

- Depending on the data type, `a / b` yields different results.
- If *both* `a` and `b` are integer or other non-decimal numerical types, `a / b` yields an integer value, **truncating the decimal.**
- If *either* `a` or `b` are floating point types (`float`, `double`), then `a / b` yields a float.

Modulus

- You've all been using modulus for most of your life:

Modulus

- You've all been using modulus for most of your life: **to tell time**!

Modulus

- You've all been using modulus for most of your life: **to tell time**!

```
15 % 5 =
```

```
8 % 3 =
```

```
3 % 2 =
```

```
12 % 15 =
```

Outline

Q & A: If you need help, let me know. Let's try to help each other out.