Oide

Tacú leis an bhFoghlaim
Ghairmiúil i measc Ceannairí
Scoile agus Múinteoirí

Supporting the Professiona
Learning of School Leaders
and Teachers

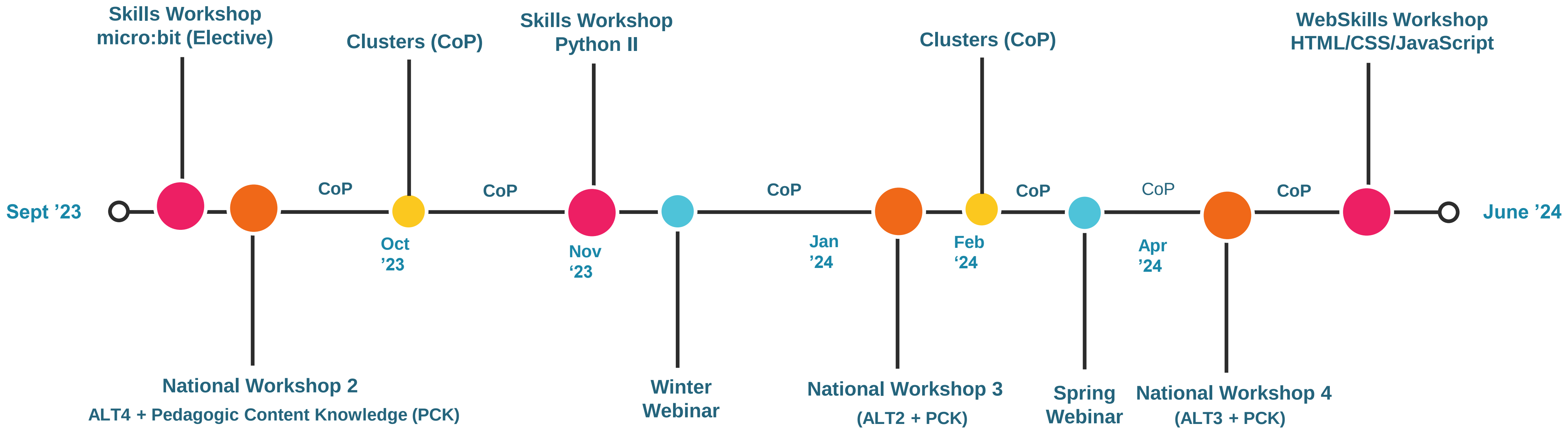
Leaving Certificate Computer Science
National Workshop 2

Day 1

Workshop Overview

Session 1 10:00 - 11:30	Introduction Computer Systems I
Tea/Coffee 11:30 – 12:00	
Session 2 12:00 - 13:30	Computational Thinking II
Lunch 13:30 - 14:30	
Session 3 14:30 - 16:30	PRIMM and Curriculum Planning

Dates for your Diary for 2023/4



Sept ’23

Skills Workshop
micro:bit (Elective)

National Workshop 2
ALT4 + Pedagogic Content Knowledge (PCK)

CoP

Clusters (CoP)

Oct
’23

CoP

Skills Workshop
Python II

Nov
‘23

Winter
Webinar

CoP

Jan
’24

National Workshop 3
(ALT2 + PCK)

Clusters (CoP)

Feb
‘24

CoP

Spring
Webinar

CoP

Apr
’24

National Workshop 4
(ALT3 + PCK)

CoP

WebSkills Workshop
HTML/CSS/JavaScript

June ’24

Next CPD event: Community of Practice cluster meetings – online early November

Introducing Oide

Oide

Tacú leis an bhFoghlaim Ghairmiúil i measc Ceannairí Scoile agus Múinteoirí

Supporting the Professional Learning of School Leaders and Teachers

Lárionad Ceannaireachta Scoile
CSL
Centre for School Leadership

An tSraith Shóisearach do Mhúinteoirí
Junior CYCLE
for teachers

NIPT
An Clár Náisiúnta Ionduchtaithe do Mhúinteoirí
The National Induction Programme for Teachers

PDST
Professional Development Service for Teachers | An tSeirbhís um Fhorbairt Ghairmiúil do Mhúinteoirí

Oide

Supports Provided by Oide

- National Workshops
- Webinars
- School Support
- Scoilnet
- Skills Workshops
- Collaboratives
- Oide website
- CompSci

Purpose for the Day

To allow Phase 5 LCCS teachers to engage with the core concepts of Computer Systems and Computational Thinking.

To experience ALT4 (Embedded Systems) through the eyes of the student by engaging with the Design Process.

Key Messages

All learning outcomes (LOs) are interwoven. This means that the specification can be used in many different ways.

ALTs provide an opportunity to teach theoretical aspects of LCCS.

LCCS can be mediated through a constructivist pedagogical approach.

Group work is a key feature in the teaching, learning and assessment of LCCS.

Oide

Tacú leis an bhFoghlaim
Ghairmiúil i measc Ceannairí
Scoile agus Múinteoirí

Supporting the Professiona
Learning of School Leaders
and Teachers

LCCS NW2 Session 1

Number Systems

By the end of this session..

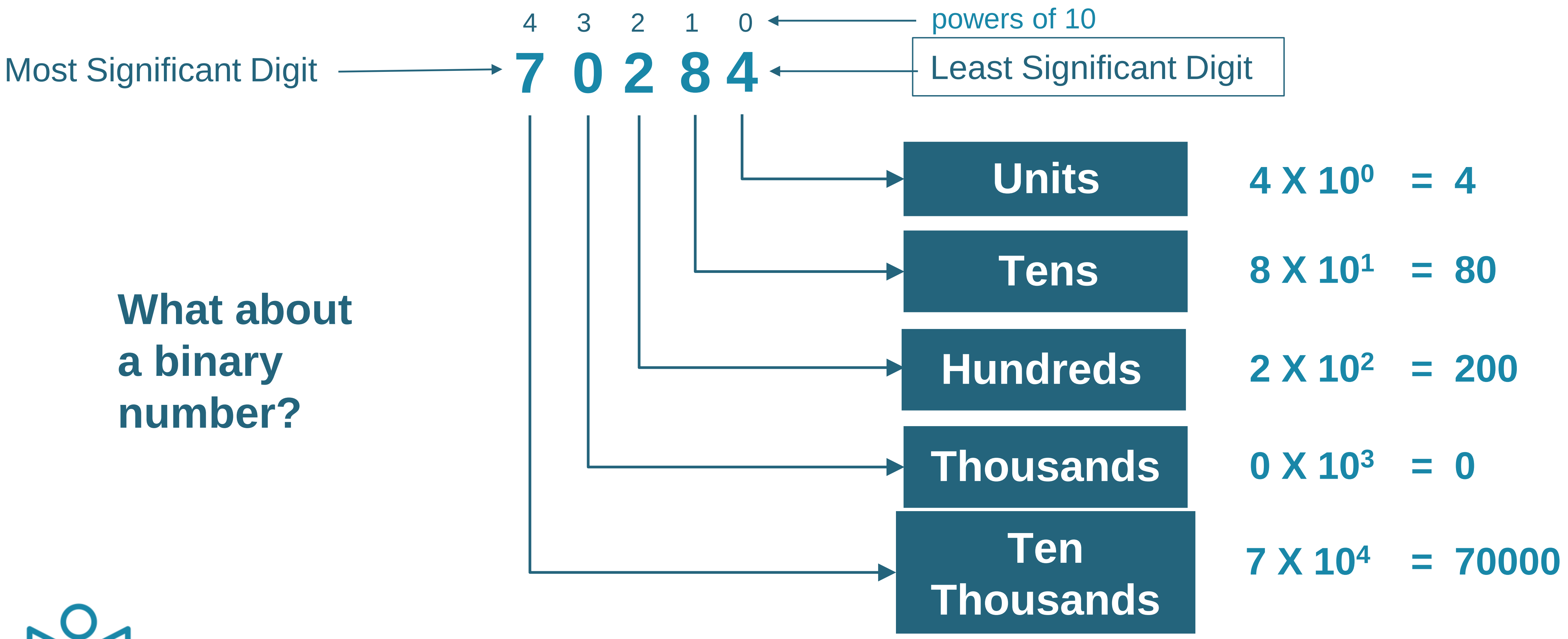
Participants will be enabled to…

- develop an understanding of Computational Thinking concepts such as abstraction, decomposition, algorithmic thinking and pattern recognition
- develop a shared understanding of how *programming as a process* can be used to mediate CT in the classroom
- convert decimal numbers to binary numbers and vice versa

Computational Thinking

"… the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent."
(Wing 2011)

Decomposition of a decimal number



powers of 10

4 3 2 1 0

Most Significant Digit

7 0 2 8 4

Least Significant Digit

Units	$4 \times 10^0 = 4$
Tens	$8 \times 10^1 = 80$
Hundreds	$2 \times 10^2 = 200$
Thousands	$0 \times 10^3 = 0$
Ten Thousands	$7 \times 10^4 = 70000$

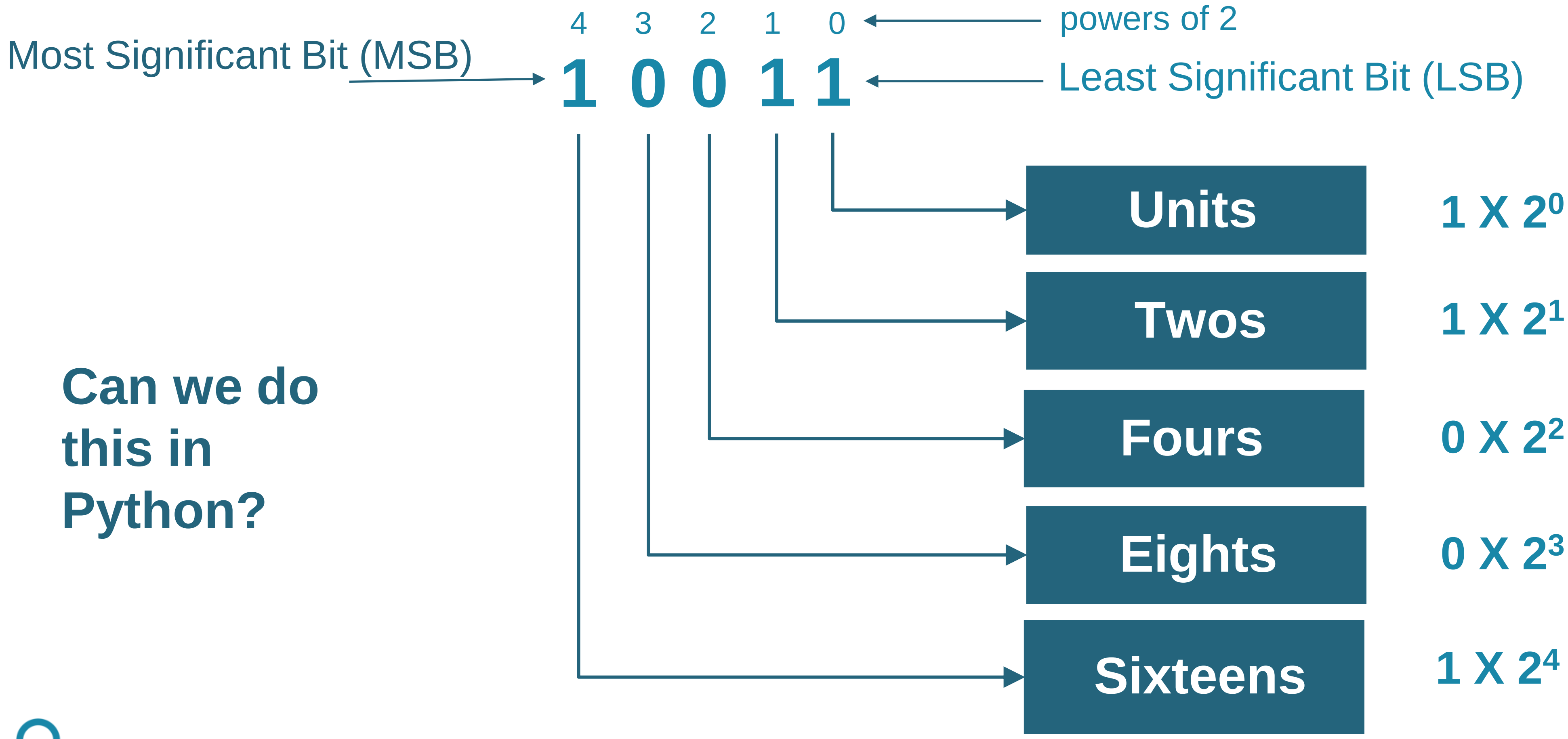
What about a binary number?

etc.

70,000 + 0,000 + 200 + 80 + 4 = 70,284

P4

Decomposition of a binary number



4 3 2 1 0 powers of 2

Most Significant Bit (MSB)

1 0 0 1 1

Least Significant Bit (LSB)

Units

Twos

Fours

Eights

Sixteens

$1 \times 2^0 = 1$

$1 \times 2^1 = 2$

$0 \times 2^2 = 0$

$0 \times 2^3 = 0$

$1 \times 2^4 = 16$

Can we do this in Python?

P4

16 + 0 + 0 + 2 + 1 = 19

etc.

$$10011_2 = 19_{10}$$

Convert 19_{10} to base 2

$$\frac{\text{Dividend}}{\text{Divisor}} = \text{Quotient} + \text{Remainder}$$

1. Divide by 2 …. note the remainder
2. The quotient becomes the new dividend
3. Keep dividing …
4. Stop when the quotient reaches zero
5. Read the answer from the bottom up

$$\frac{19}{2} = 9 + 1$$

$$\frac{9}{2} = 4 + 1$$

$$\frac{4}{2} = 2 + 0$$

$$\frac{2}{2} = 1 + 0$$

$$\frac{1}{2} = 0 + 1$$

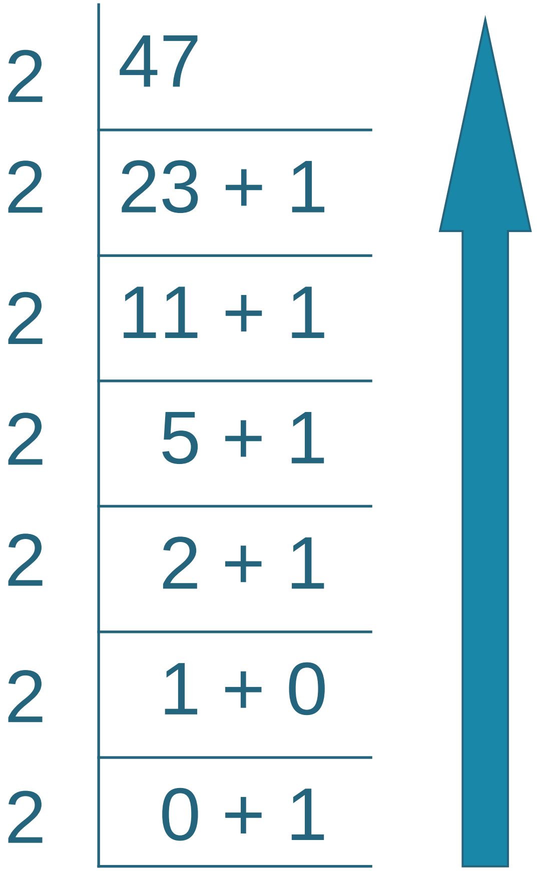
So, $19_{10} = 10011_2$

$$\begin{array}{r|rl} 2 & 19 & \\ \hline 2 & 9 & +\ 1 \\ \hline 2 & 4 & +\ 1 \\ \hline 2 & 2 & +\ 0 \\ \hline 2 & 1 & +\ 0 \\ \hline 2 & 0 & +\ 1 \end{array}$$

Decimal -> Binary (another example)

1. Divide by 2 …. note the remainder
2. The quotient becomes the new dividend
3. Keep dividing …
4. Stop when the quotient reaches zero
5. Read the answer from the bottom up

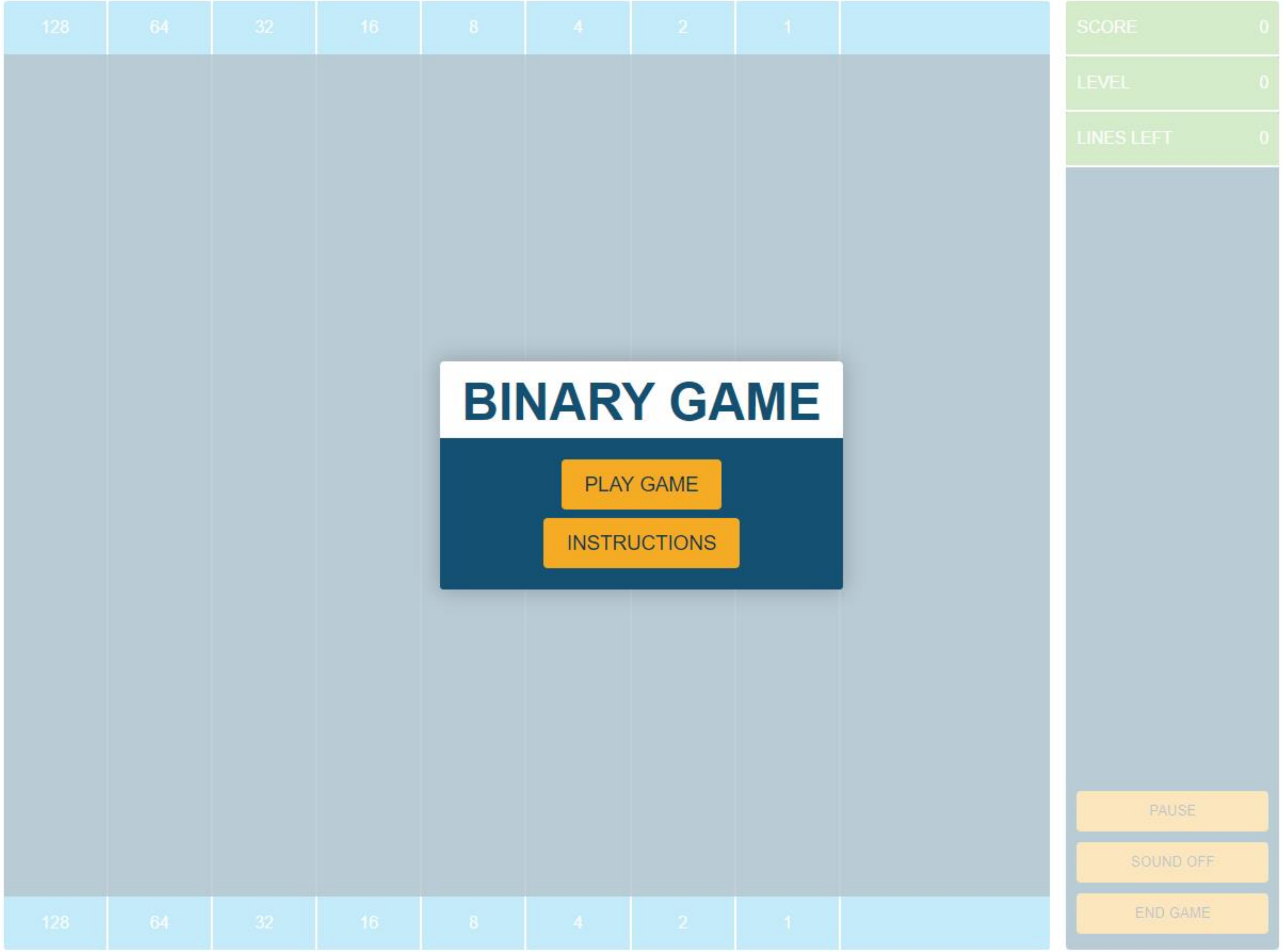
Convert 47_{10} to base 2



2	47
2	23 + 1
2	11 + 1
2	5 + 1
2	2 + 1
2	1 + 0
2	0 + 1

$$47_{10} = 101111_2$$

P5



SCORE 0

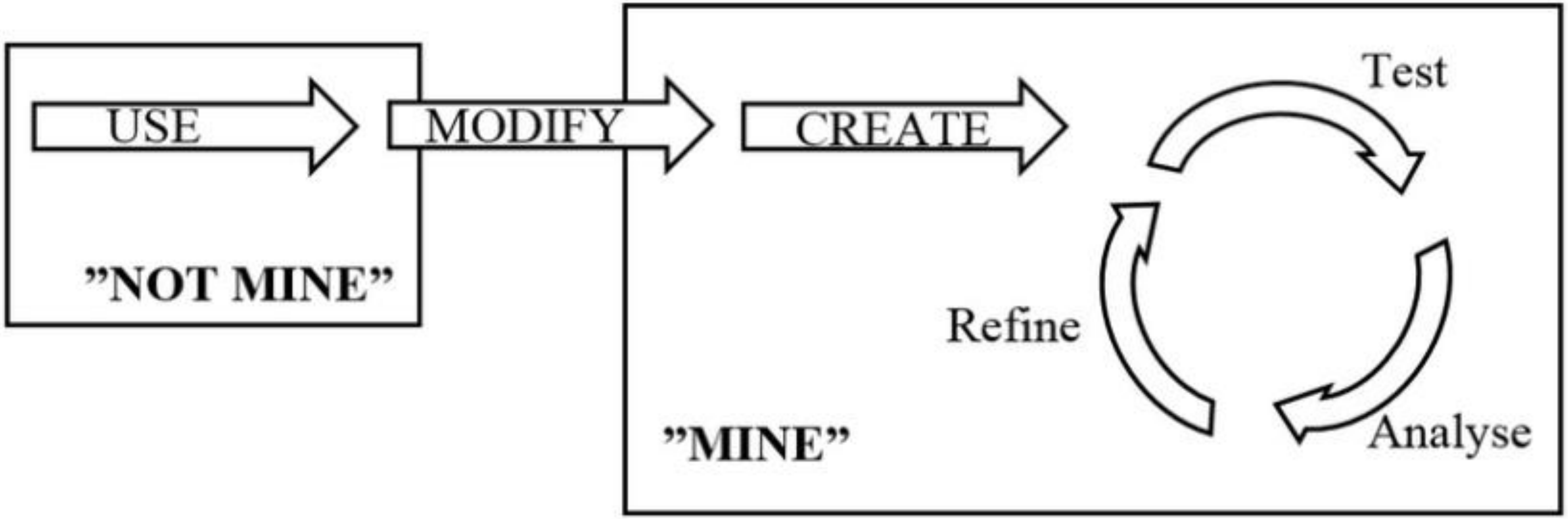
LEVEL 0

LINES LEFT 0

https://learningcontent.cisco.com/games/binary/index.html

Code Along Activity

Use Modify Create



Program Tracing / Debugging

```python
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

This is what is displayed ….

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

```
quotient:   9
```

This is what is displayed ….

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

```
quotient:   9
remainder1: 1
```

This is what is displayed ….

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

```
quotient:   9
remainder1: 1
```

This is what is displayed ….

```
>>> 9 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

quotient: ~~9~~ 4

remainder1: 1

This is what is displayed ….

```
>>> 9 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

```
quotient:    ~~9~~ 4
remainder1:  1
remainder2:  1
```

This is what is displayed ….

```
>>> 9 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

quotient: ~~9~~ 4

remainder1: 1

remainder2: 1

This is what is displayed ….

```
>>> 9 1
>>> 4 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

quotient: ~~9~~ ~~4~~ 2

remainder1: 1

remainder2: 1

This is what is displayed ….

```
>>> 9 1
>>> 4 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

quotient: ~~9~~ ~~4~~ 2

remainder1: 1

remainder2: 1

remainder3: 0

This is what is displayed ….

```
>>> 9 1
>>> 4 1
```

Program Tracing / Debugging

```
quotient = 19//2
remainder1 = 19%2
print(quotient, remainder1)

# copy+paste ...
quotient = 9//2
remainder2 = 9%2
print(quotient, remainder2)
# Once ...
quotient = 4//2
remainder3 = 4%2
print(quotient, remainder3)
# Twice ...
quotient = 2//2
remainder4 = 2%2

# Three times ...
quotient = 1//2
remainder5 = 1%2
```

The Notional Machine / Working Memory

quotient: ~~9~~ ~~4~~ 2

remainder1: 1

remainder2: 1

remainder3: 0

This is what is displayed ….

```
>>> 9 1
>>> 4 1
>>> 2 0
```

Group Activity: Breakout

Binary to Decimal (1 of 2)

```
binary_number = 10011
decimal_number = 0

digit0 = 10011 % 10 # lsb
stem = 10011 // 10
print(stem, digit0)
```

How could we develop this Python code to a general solution?

Binary to Decimal

```
# ... convert binary 10011 to decimal ...
# ... the initial number is a string
binary_number = "10011"
# index:         01234

units    = int(binary_number[4])*1
twos     = int(binary_number[3])*2
fours    = int(binary_number[2])*4
eights   = int(binary_number[1])*8
sixteens = int(binary_number[0])*16
decimal = units+twos+fours+eights+sixteens
```

How could we develop this Python code to a general solution?

20 minutes

Group Activity: Feedback

Break

Oide

An Roinn Oideachais
Department of Education

Tacú leis an bhFoghlaim Ghairmiúil i measc Ceannairí Scoile agus Múinteoirí

Supporting the Professional Learning of School Leaders and Teachers