### Problem 1.

Convert each decimal number to hexadecimal:

26 = _____

60 = _____

175 = _____

255 = _____

---

### Problem 2.

Convert each hexadecimal number to decimal:

$\texttt{1A}_{16}$ = _____

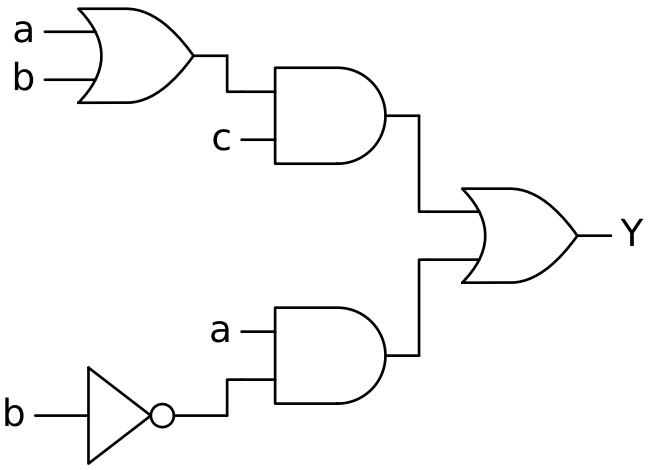
$\texttt{2F}_{16}$ = _____

$\texttt{B4}_{16}$ = _____

$\texttt{FF}_{16}$ = _____

---

### Problem 3.




Write the Boolean expression for output $Y$, then draw its truth table.

---

### Problem 4.

Convert each binary number to hexadecimal:

$11010_2$ = _____

$101111_2$ = _____

$10110100_2$ = _____

$1111_2$ = _____

---

### Problem 5.

Convert each hexadecimal number to binary:

$3C_{16}$ = _____

$A7_{16}$ = _____

$1E_{16}$ = _____

$FF_{16}$ = _____

---

### Problem 6.

Every device on a network has an *IP address*: four numbers separated by dots, each stored in one byte (so 0 to 255). Convert each part of these addresses to 8-bit binary.

`192.168.1.1` = _____ . _____ . _____ . _____

`10.0.0.255` = _____ . _____ . _____ . _____

`8.8.8.8` = _____ . _____ . _____ . _____

Which one of these could *not* be a real IP address, and why?

`192.168.0.1`  `10.0.256.4`  `172.16.0.255`

*Fun fact: 8.8.8.8 is one of Google's real public addresses.*

---

**Problem 7.**

A black-and-white picture is a grid of *pixels*: each pixel is one bit — 1 means a filled (black) square, 0 means blank. Each row of 8 pixels is one byte, given below in hexadecimal. Convert each byte to 8-bit binary, then shade a square wherever there is a 1.

$66_{16}$ = _____

$FF_{16}$ = _____

$FF_{16}$ = _____

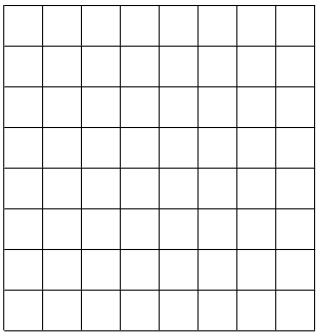
$FF_{16}$ = _____

$7E_{16}$ = _____

$3C_{16}$ = _____

$18_{16}$ = _____

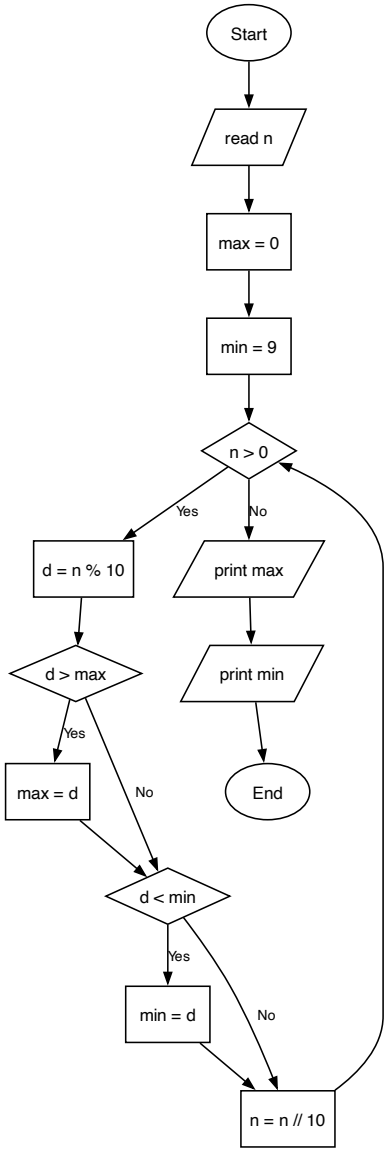
$00_{16}$ = _____



*What picture appears? The whole image is just 8 bytes — 64 ones and zeros. Roughly how many bytes would a picture twice as wide and twice as tall (16 × 16) need?*

## Problem 8.

Trace this algorithm for $n = 5184$, and complete the trace table.




| n | max | min | d | output |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |