Peer Instruction #3:
C Programming and Transistors

Is the following code legal in C, Java, both languages or neither language?

```
double circleArea(double radius) {
    return (3.14159 * radius * radius);
}
```

A. C
B. Java
C. Both
D. Neither

C versus Java

What does the C code shown below print to the console?

```
int i0 = 256;
double d0 = 34.5678910;
printf(
   "%d, 0x%04x, %.3f\n",
    i0, i0, d0
);
```

A. 256, 0x0100, 4.568
B. 256, 0x0100, 34.568
C. 256, 0x100, 4.568
D. 256, 0x100, 34.568
E. Will not compile

printf Formatters

What does the C code shown below print to the console?

```
int i = 1234;
int *p = &i;
printf("%d\n", *p);
```

A. value of i
B. address of i
C. value of p
D. address of p
E. Will not compile

C Pointers

What does the C code shown below print to the console?

```
int i= 0x1234;
int mask = 0x0F0F;
printf("%x, ", i & mask);
printf("%x", , i | mask);
printf("%x\n", ~mask & 0xffff);
```

A. 204, 1f3f, f0f0
B. 1030, ffff, f0f0
C. 1f3f, 1234, edcb
D. 204, f2f4, edcb
E. None of the above

Bit Manipulation

Which C statement checks the value of the integer and returns a true (nonzero) if bit 8 is set and false (zero) otherwise?

```
int i = 0x5678;
1)  return (i & 0x0100);
2)  return ((i >> 8) & 1);
3)  return (i & (1 << 8));
4)  return (i & 256);
```

A. 1
B. 2
C. 3
D. 4
E. All of the above

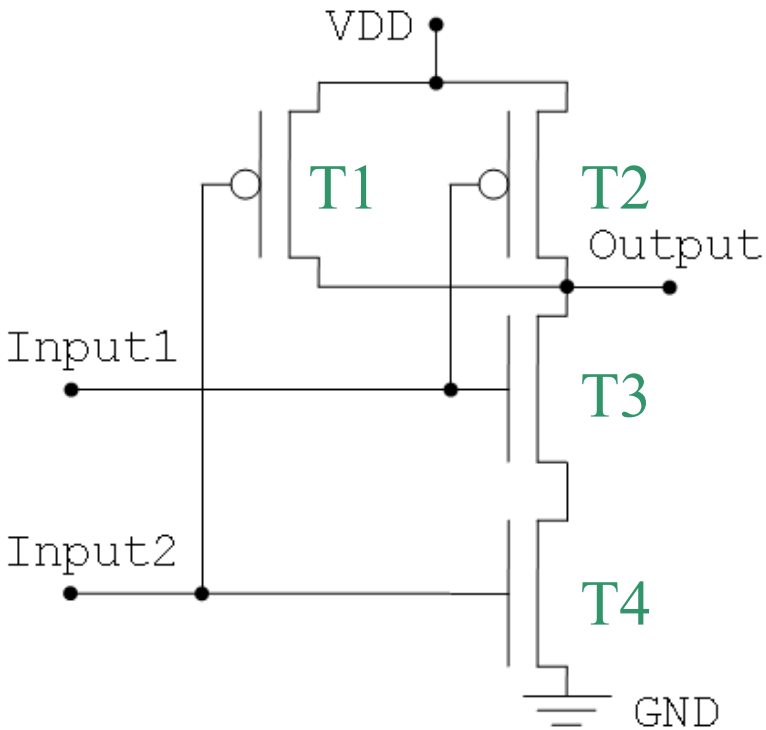
Bit Manipulation

Is a p-type transistor closed or open when there is voltage applied to the gate? What about an n-type transistor?

A. closed, closed
B. open, open
C. open, closed
D. closed, open
E. Depends on whether the transistor is attached to power or ground.

Transistors

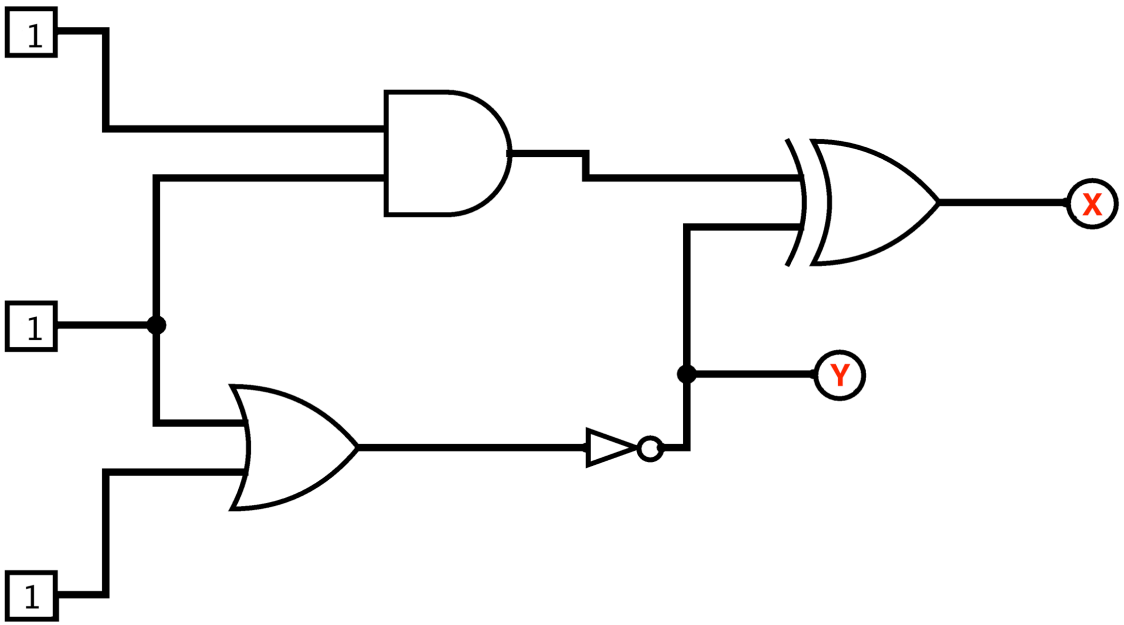
Which list shows the correct status of each transistor (T1,T2,T3,T4) and the Output, when Input1 = 1 and Input2 = 0?



A. closed,open,closed,open,0
B. closed,open,closed,open,1
C. closed,closed,open,open,1
D. open,open,closed,closed,0
E. None of the above

Gates

What are the outputs X, Y in the combinational logic circuit below, given the inputs shown.



A. 0, 0
B. 0, 1
C. 1, 0
D. 1, 1
E. None of the above

Combinational Logic