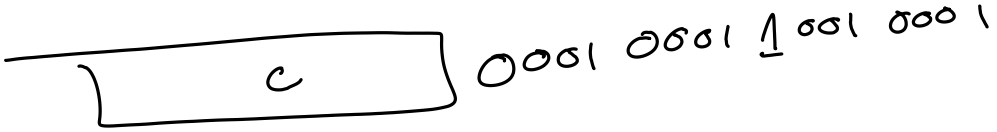
System Verilog

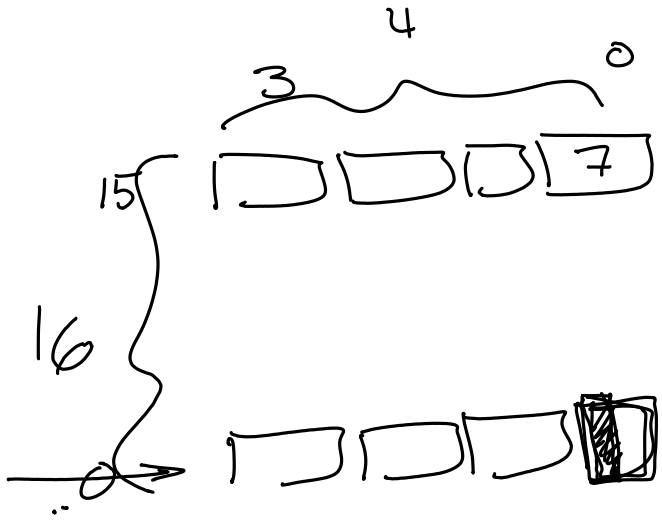
Exercise 1: What are the packed and unpacked dimensions of each declaration?

```
logic [3:0] x ;
logic signed [15:0] y ;
logic [3:0] [7:0] z [15:0] ;
```

	packed	unpacked
x	3:0	none
y	15:0	"
z	3:0, 7:0	15:0




```
z[0][0][7] = 1 ;
```



Exercise 2: What are the signedness, size and value of each constant and each expression above?

```
x = 4'b01xz ; //
x = -1 + 0 ;  // x=15

y = -1 + 4'shf ;  // y=-
x = y ;           // x=1

z[0] = '1 ;         // z|
z[0] = {4{4'b1}} ; // z|
z[0][0][7] = 1 ;    // z|
z[15:0] = '{16{z[0]}} ;
```

16 x 4 x 8 = $2^4 \cdot 2^2 \cdot 2^5 =$

$2^{11} = 2048$

```
logic [15:0] x ;
logic signed [15:0] y ;

x = 16'hfff0 ; // x=65520
y = x >>> 1 ;  // y=0x7ff8
y = signed'(x) >>> 1 ; // y=0xfff8
y = |y ; // y=1

x = 8'h4x ;       // x=X
y = x == 8'h4x ;  // y=X
y = x[6:3] === 7'b100x ; // y=1
y = x ==? 8'h4x ; // y=1
y = {x[7:4],x[6]} ; // y=0x0009
end
```

0100 1

65 4 3 2 1 0

0100 XXX

0000 000

0001 00X

0100 XXX

Exercise 3: Should each of the following nets (or variables) be declared `wire` or `reg`?

```
module test (a,b,c,d,q) ;
dff d0 (clk,d,q) ; // assume only q is an output
assign d = a & b ;
always@* clk = a & c ;
endmodule
```

a net

b net

c net

d reg

clk reg

q net