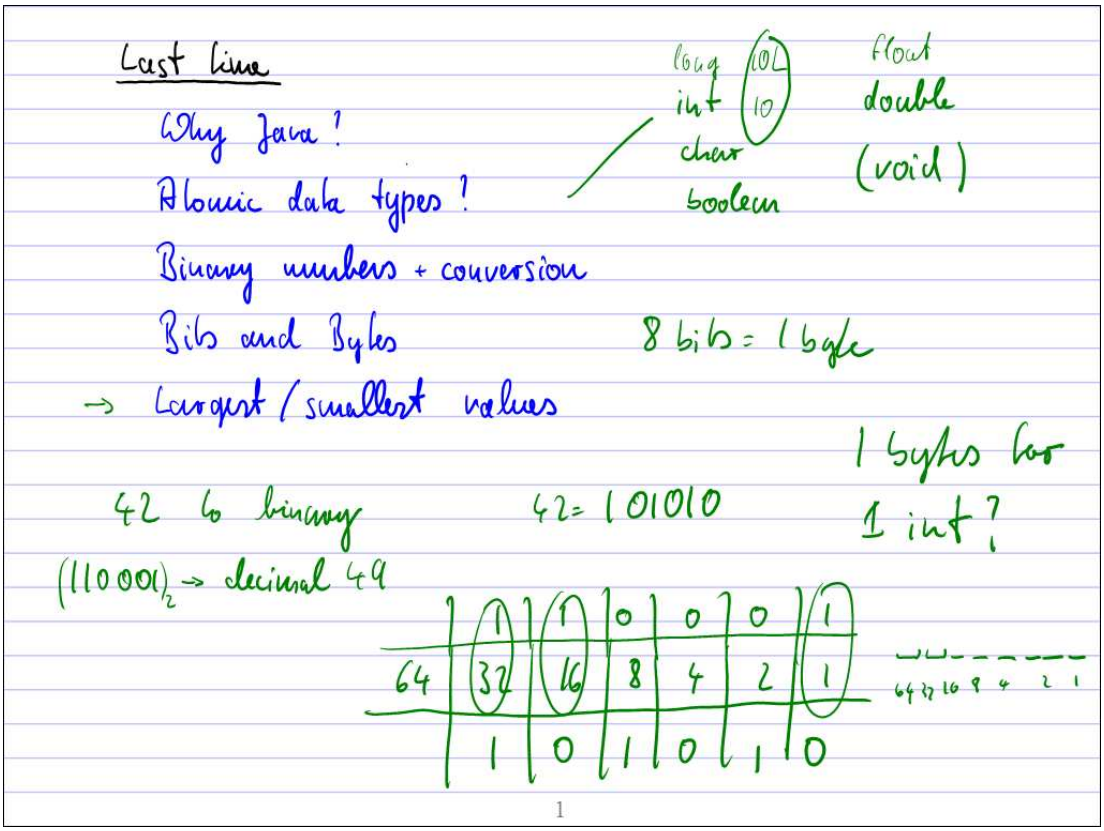
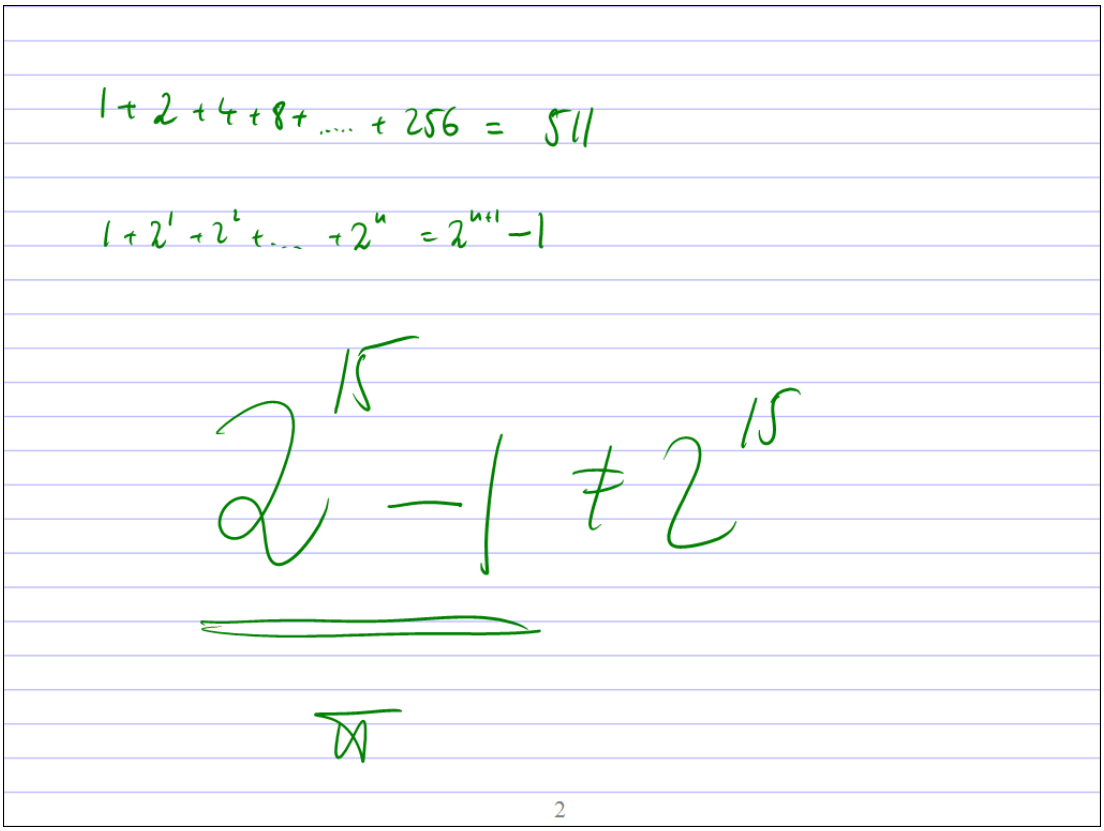
Panel 1



Panel 2



Panel 3

All data types stored on a computer have a smallest and largest value. In Java:

Type	Range	
double	largest positive/negative value:	±1.7976931348623157E308
	smallest non-zero value:	±4.9E-324
	significant digits:	16 digits after decimal point
float	largest positive/negative value:	±3.4028235E38
	smallest non-zero value:	±1.4E-45
	significant digits:	8 digits after decimal point
int	largest value	2147483647
	smallest value:	-2147483648
short	largest value	32767
	smallest value:	-32768
long	largest value	9223372036854775807
	smallest value:	-9223372036854775808

3

Panel 4

Quiz #1

① How many bits are one byte?

② If we only stored positive integers, and one integer uses 2 bytes, what is the largest integer we could store?

4

Panel 5

Quiz #1 - continued

③ Convert 39 to a binary number

④ Convert the binary number $(10110111)_2$ to its decimal representation

5

Panel 6

Declare Variables

Every piece of data must be stored in a variable of a specific type!

Definition 1.12: Declaration of Variables

To declare a variable that can store data of a specific type, the syntax:

```
type varName [, varName2,..., varNameN];
```

is used, where `type` is one of the basic data types, `varName` is the name of the variable, and `varName2, ..., varNameN` are optional additional variables of that type. Variables can be declared virtually anywhere in a Java program.

Ex:

```
int x;
double z, y;
```

6

Panel 7

Variable Names:

Definition 1.13: Valid Names for Variables

A variable name must start with a letter, ~~a dollar sign '$'~~, or the ~~underscore character '_'~~, followed by any character or number. It can not contain spaces. The reserved keywords listed in table 1.12 can not be used for variable names. Variable names are case-sensitive.

Java Reserved Keywords					
abstract	boolean	break	byte	case	catch
char	class	const	continue	default	do
double	else	extends	false	final	finally
float	for	goto	if	implements	import
instanceof	int	interface	long	native	new
null	package	private	protected	public	return
short	static	super	switch	synchronized	this
throw	throws	transient	true	try	void
volatile	while				

Table 1.12: Reserved keywords in Java

7

Panel 8

Assign Values to Variables

Definition 1.14: Assigning a Value to a Variable

To assign a value to a declared variable, the assignment operator "=" is used:

```
varName = [varName2 = ... ] expression;
```

Assignments are made by first evaluating the expression on right, then assigning the resulting value to the variable on the left. Numeric values of a type with smaller range are compatible with numeric variables of a type with larger range (compare table 13). Variables can be declared and assigned a value in one expression using the syntax:

```
type varName = expression [, varname2 = expression2, ...];
```

initialize to 0

Ex: int x; double z;

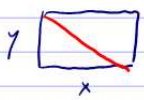
x = 10. ~~0~~; z = 10;

or: int x = 10; double z = 10.0

8

Panel 9

What's in a Name:




```
double x, y, z;
x = 10.0;
y = 20.0;
z = 2*(x + y);
w = 0.5 * x * y;
```

```
double width = 10.0, height = 20.0;
double perimeterOfRectangle = 2*(width + height);
double base = width;
double areaOfTriangle = 0.5 * base * height;
```

Software Engineering Tip: Variables serve a purpose and the name of a variable should reflect that purpose to improve the readability of your program. Avoid one-letter variable names. Do not reuse a variable whose name does not reflect its purpose.

Whenever possible, assign an initial value to every variable at the time it is declared. If a variable is declared without assigning a value to it, all basic types except `boolean` are automatically set to `0`, `boolean` is set to `false`, and all other types are set to `null`.

Declare variables as close as possible to the code where they are used. Do not declare all variables at once at the beginning of a program (or anywhere else).

9

Panel 10

Doing Math:

Basic Math: + - * / % remainder

Task: start BlueJ, write a program to add two variables x = 10, y = 20 and print the answer!

10

Panel 11

HW: ① Figure out what % does?

② Also, compute

```
int x = 5;
int y = 2;
int z = y/x;
```

Explain!

11