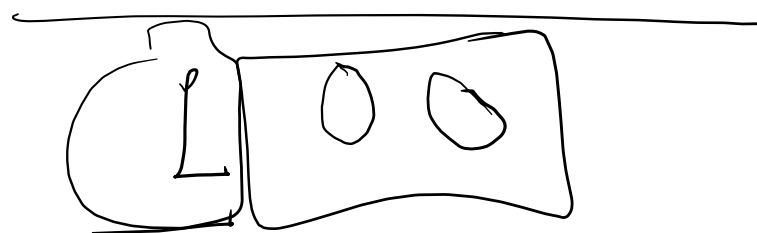
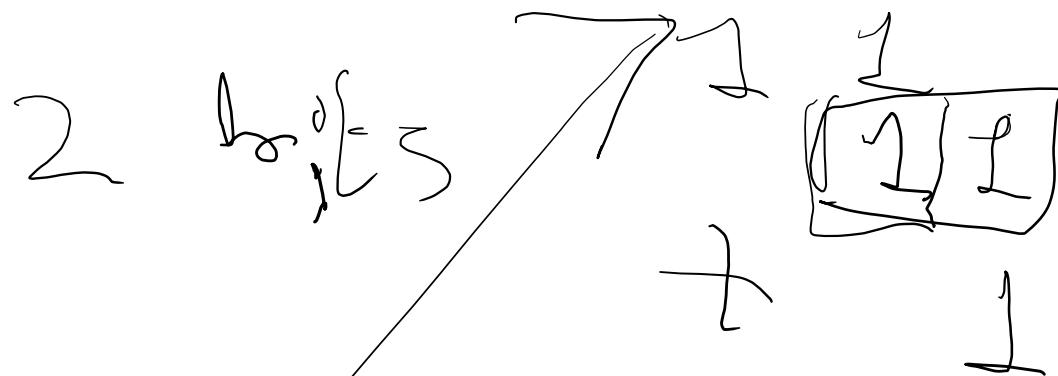


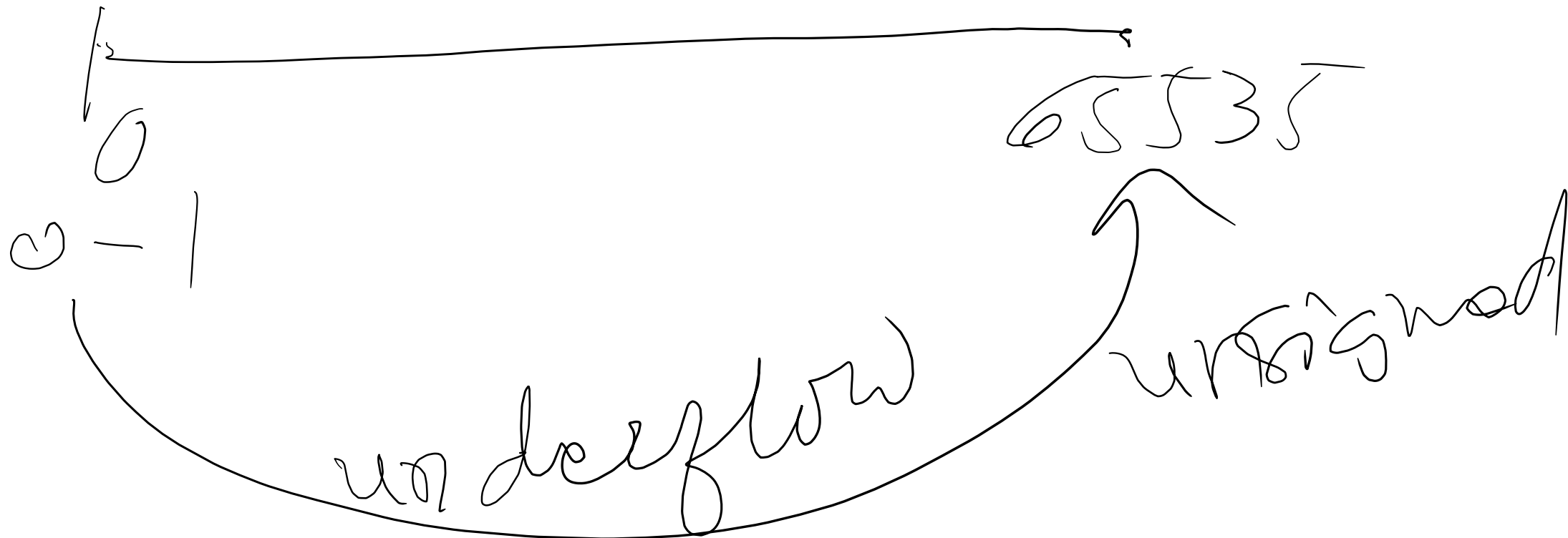
Let's solve it

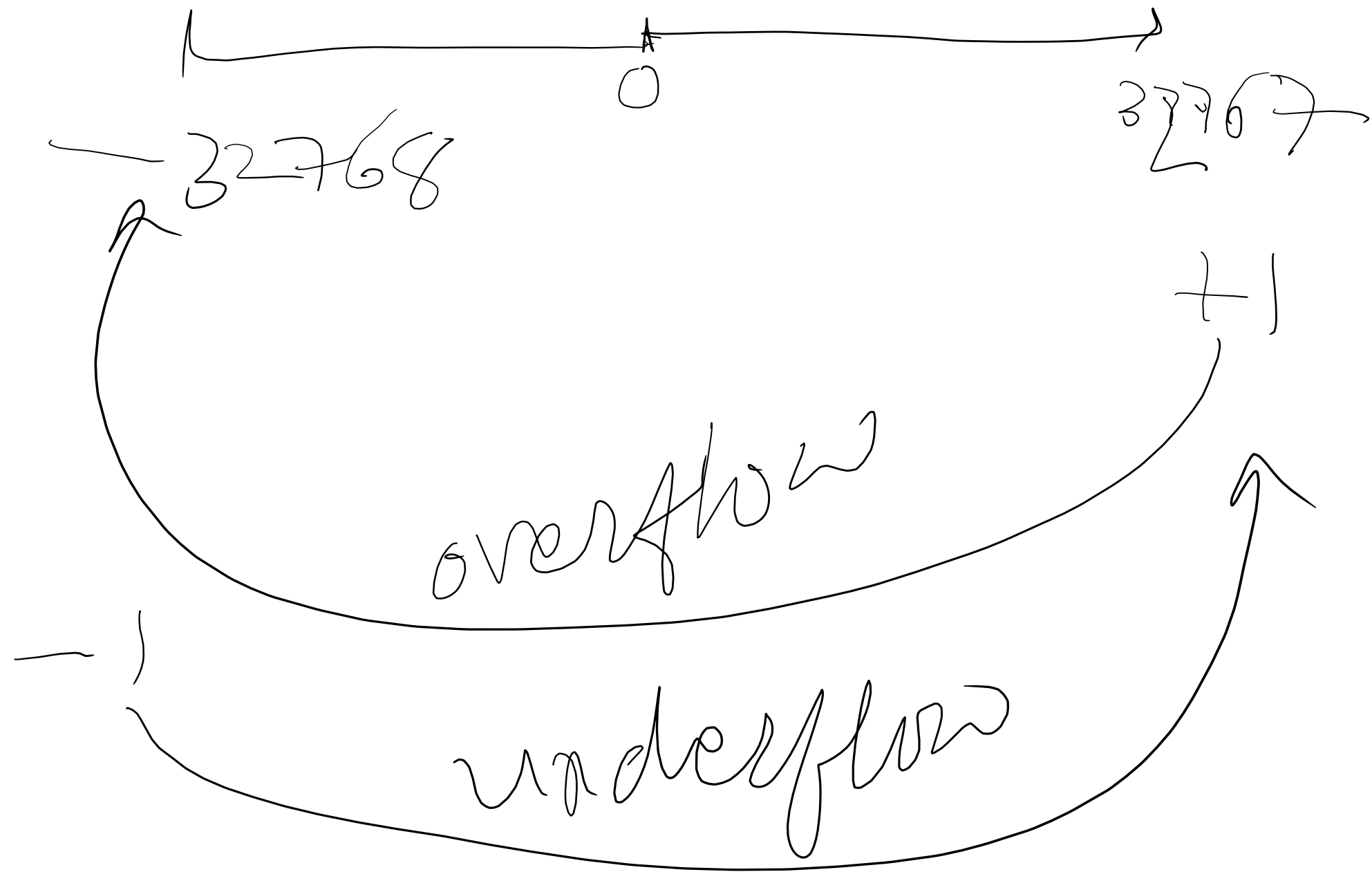
7



unformed





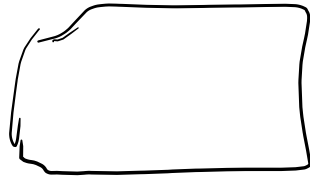


Assignment operation done using '='

int x = 59; 59

Initialization of a variable


int x;




garbage
for

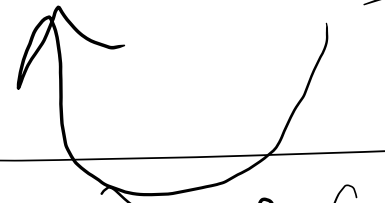
automatic
variable

Equality
Comparison
will be
done
using
operator
'=='

int x = 5; x 

int y; y  garbage

y = x; 


~~printf("x is v.d", x);~~

~~printf("y is v.d", y);~~

copy and paste

It is
NOT
cut
and
paste

$x = x + 1;$ is allowed only
if x is NOT a
constant.

Regular variable;
Error:
↳ value required.

$y = y + 1;$ ✓

is possible

as y is

not a constant.

error: assignment of read-only variable 'x'

~~Over~~ interesting to
memory of ~~y~~

reads data from y memory

$$y = y + 1$$

left Load
hand side
of =
L-Value.

~~right~~
hand side of

=
R-Value

reading
the
data

memory
updateable / writing

$y = y + 1$;

Increments data of y .

provided it is not
constant.

what if y has highest value
 \Rightarrow overflow

what if y has lowest value
next towards zero. \Rightarrow underflow

$$y = y - 1;$$

What if y has lowest value
 \Rightarrow underflow.

char gender = 'm';

gender = 'f';

char
assignment

Assigning float

float f = 2.19;

f = 5.532;

printf as a function

```
int printf(const char *format, ...);
```

```
printf("x is %d & y is %d  
z is %d",
```

displays
on screen.

```
x,  
y,  
z);
```

scanf as a function

int scanf(const char *format, ...);

int x, y, z;

scanf("%d %d %d", &x, &y, &z);

↑
address of
operator

reads from keyboard