Palindrome

243 ← Number

342 ← Reverse

if Number EQUALS Reverse then

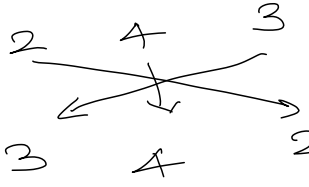
Its a Palindrome

else

Its NOT Palindrome.

Reverse a Number

Number → 243



$243 \Rightarrow 2\times100 + 4\times10 + 3\times1$

$342 \Rightarrow 3\times100 + 4\times10 + 2\times1$

→ From number take a digit out, one at a time. → Divide by 10

→ Add the digit to the reverse.

Reverse → 0

243 DIV 10 → Q: 24, R: 3

Reverse → 3

Number → 24

24 DIV 10 → Q : 2
→ R : 4

Reverse → 3 ⇒ 3 × 1

34 ⇒ 3 × 10 + 4 × 1

Multiply Reverse by 10 and add remainder digit

Reverse = 3 × 10 + 4 = 34
Number = 2 → 30 + 4 = 34

2 DIV 10 → Q : 0
→ R : 2

Reverse → 34 ⇒ 3 × 10 + 4 × 1

342 ⇒ 3 × 100 + 4 × 10 + 2 × 1

Reverse = 34 × 10 + 2 → 340 + 2 = 342
= 342

Number = 0 ⇐ STOP, when Number becomes 0.

Reverse a Number

1. Get a Number.
2. Set Reverse to 0.
3. Repeat following steps while Number is not zero.
   4. Digit = remainder of Number DIV 10.
   5. Multiply reverse by 10.
   6. Add Digit to reverse.
   7. Set Number to quotient of Number DIV 10.

8. Step.

Number → 0
Reverse → 0

~~1234~~ ~~123~~ ~~12~~ 1 0
~~12345~~
0 ~~5~~ ~~50~~ ~~54~~ ~~540~~ ~~543~~ ~~5430~~ ~~5432~~ ~~54320~~ 54321
Digit → ~~5~~ ~~4~~ ~~3~~ 2 1

1 DIV 10 → Q: 0, R: 1

12345 DIV 10 → Q: 1234, R: 5

1234 DIV 10 → Q: 123, R: 4

123 DIV 10 → Q: 12, R: 3

12 DIV 10 → Q: 1, R: 2

Fibo Series

1 1

Number → 10
Reverse → 01
= 1

First term of series = 1

Second term of series = 1

Third term of series = First term + Second term

= 1 + 1 = 2

Fourth term = Second term + Third term

= 1 + 2 = 3

Fifth term = Third term + Fourth term

= 2 + 3 = 5

Sixth term = Fourth term + Fifth term

= 3 + 5 = 8

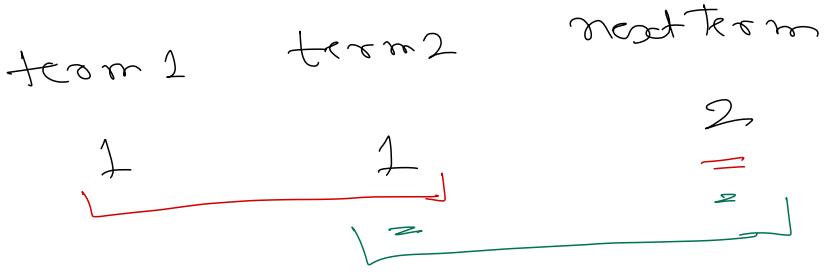
1 1 2

= =

term1 = 1

term2 = 1

next term = term1 + term2

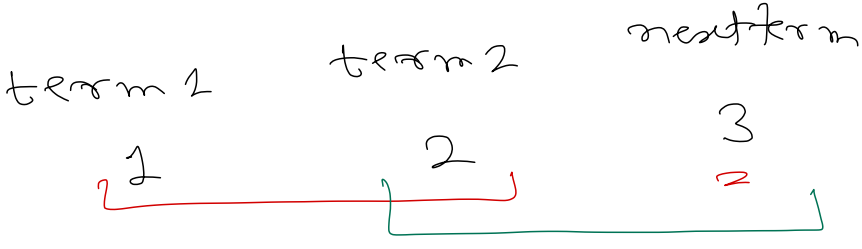


term1 = 1

term2 = 2

nextTerm = term1 + term2

= 1 + 2 = 3



term1 = 2

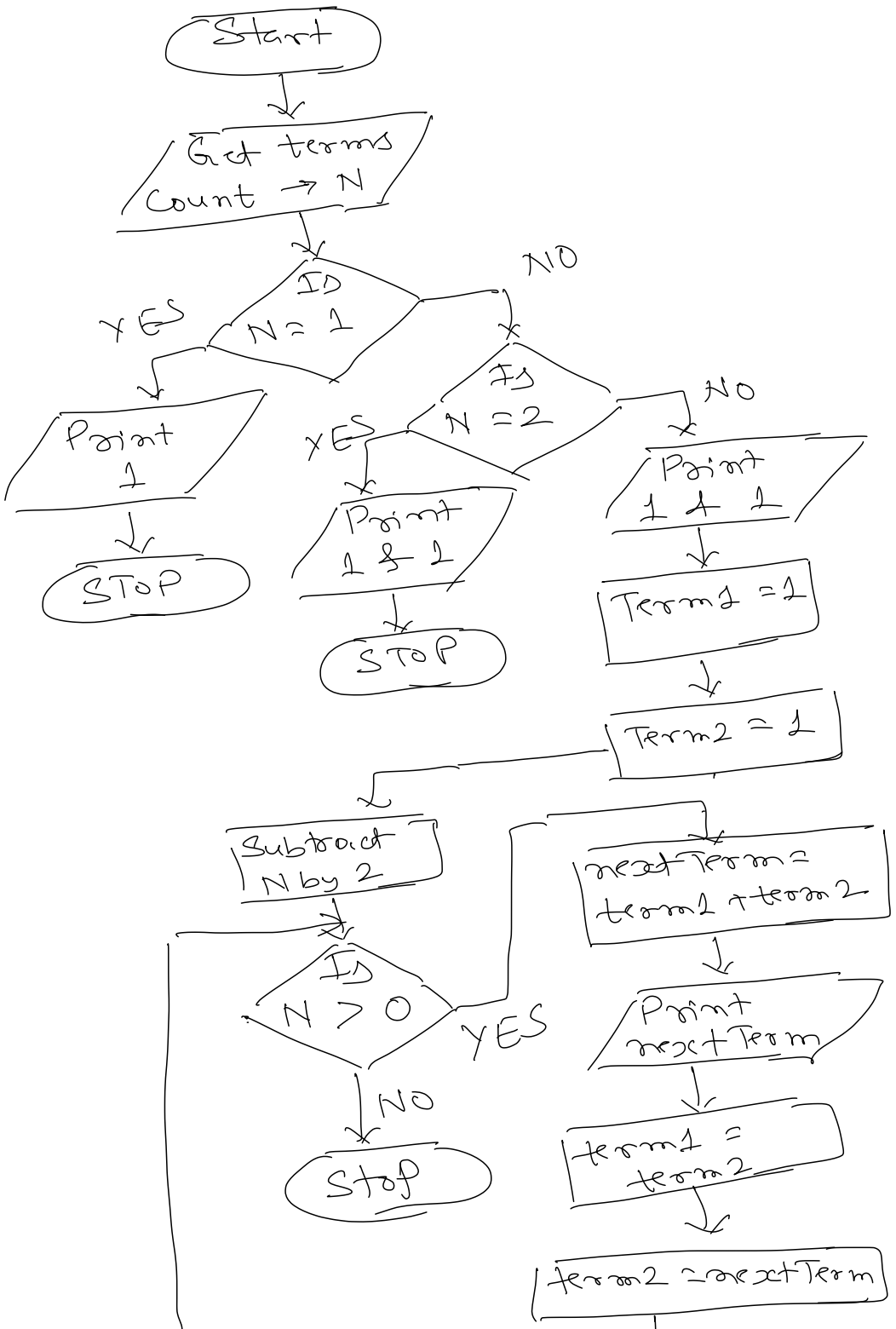
term2 = 3

nextTerm = term1 + term2

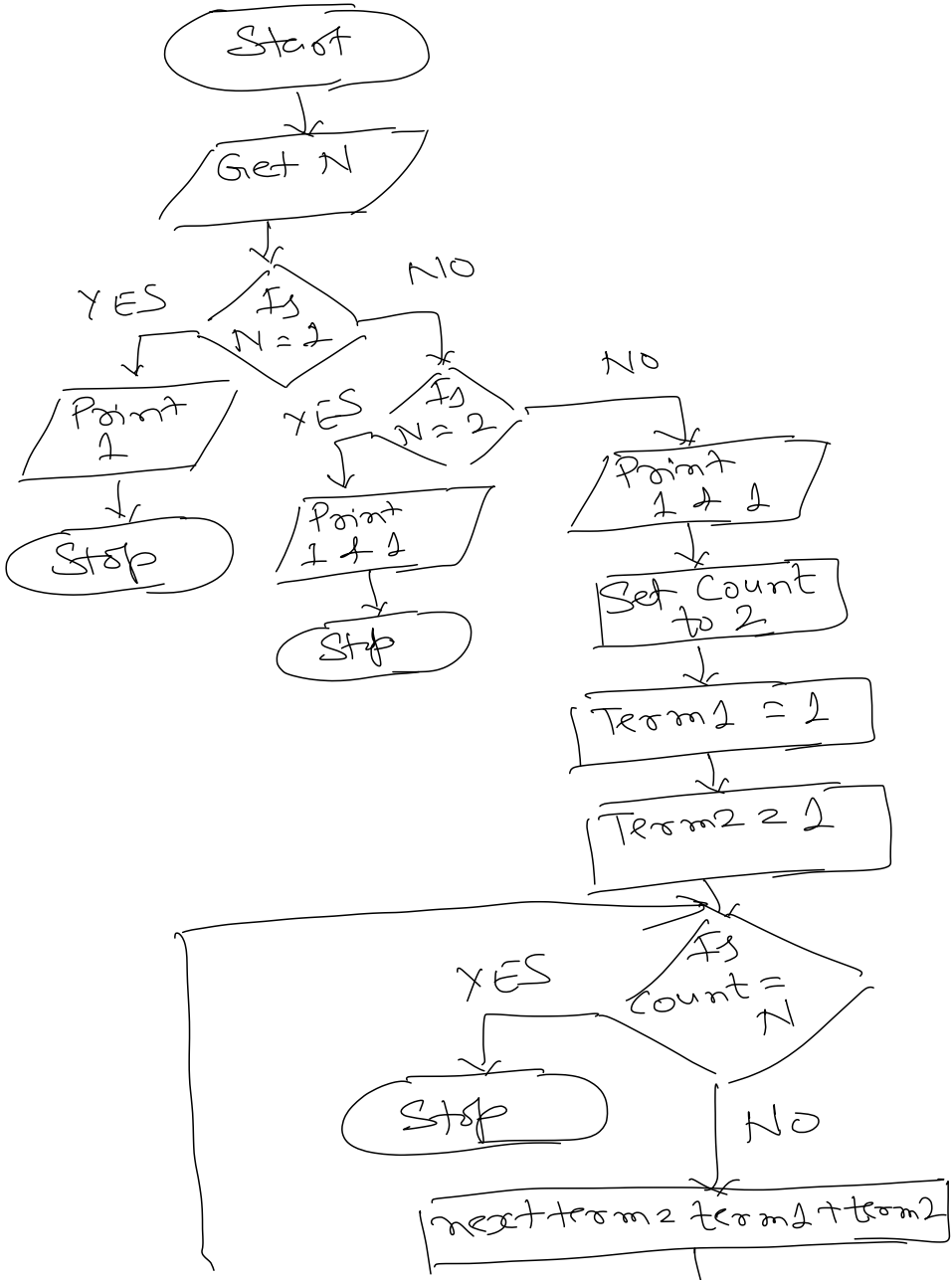
= 2 + 3 = 5

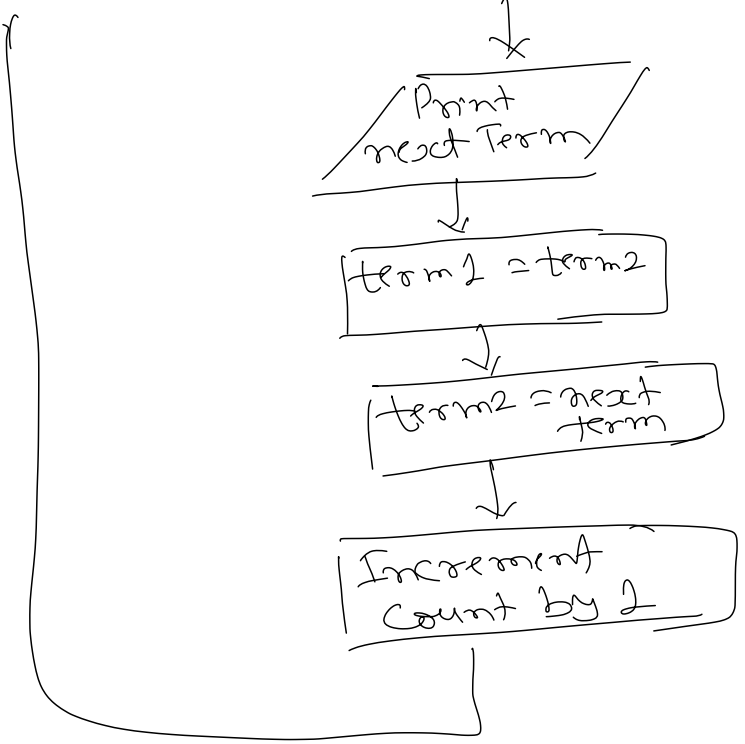
term1 term2

2 3

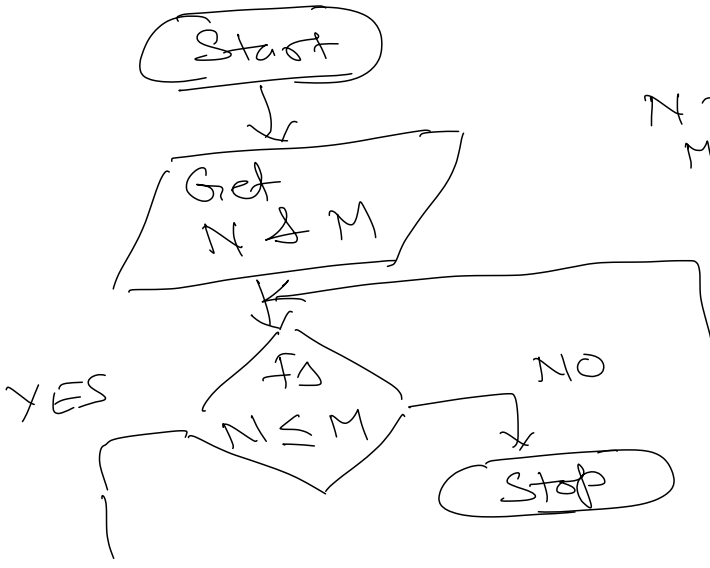

Start
Get terms Count → N
ID N = 1
YES
NO
Print 1
STOP
ID N = 2
YES
NO
Print 1 & 1
STOP
Print 1 1 1
Term1 = 1
Term2 = 1
Subtract N by 2
ID N > 0
YES
NO
Stop
nextTerm = term1 + term2
Print nextTerm
term1 = term2
term2 = nextTerm


subtract
N by 1


Start
Get N
If
N = 2
YES
NO
Print
1
Stop
If
N = 2
YES
NO
Print
1 + 1
Stop
Print
1 + 1
Set Count
to 2
Term1 = 1
Term2 = 1
If
count =
N
YES
NO
Stop
nextterm2 term1+term2

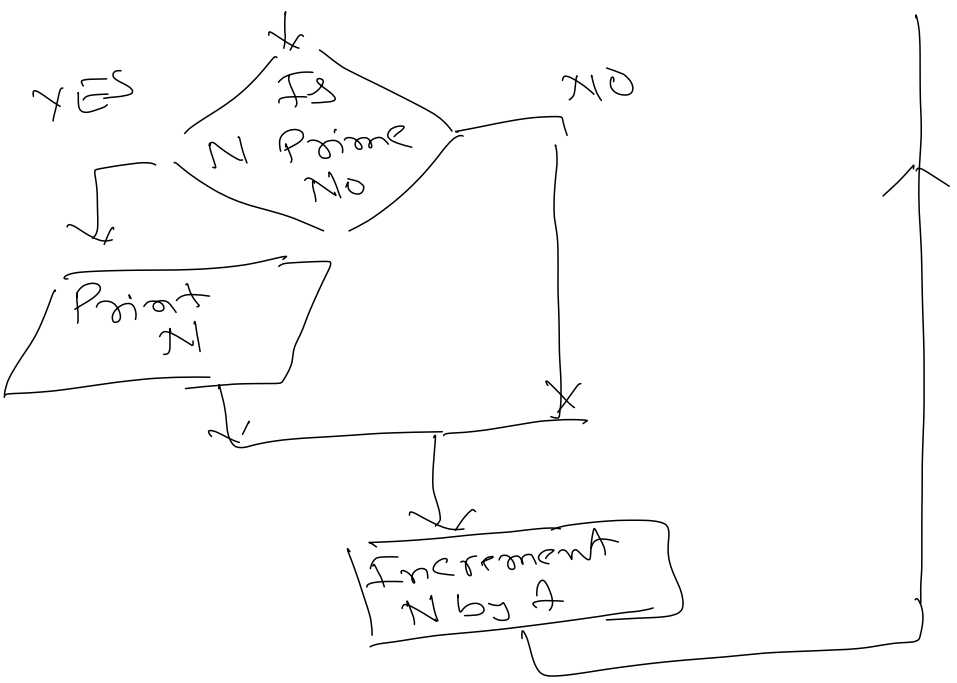


Find prime numbers between two positive integers N & M, including N & M.



N = 3
M = 5

3 4 5



N → 3 ~~4~~ ~~5~~ 6

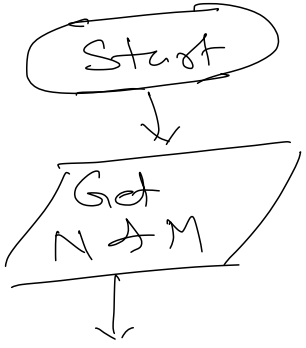
M → 5

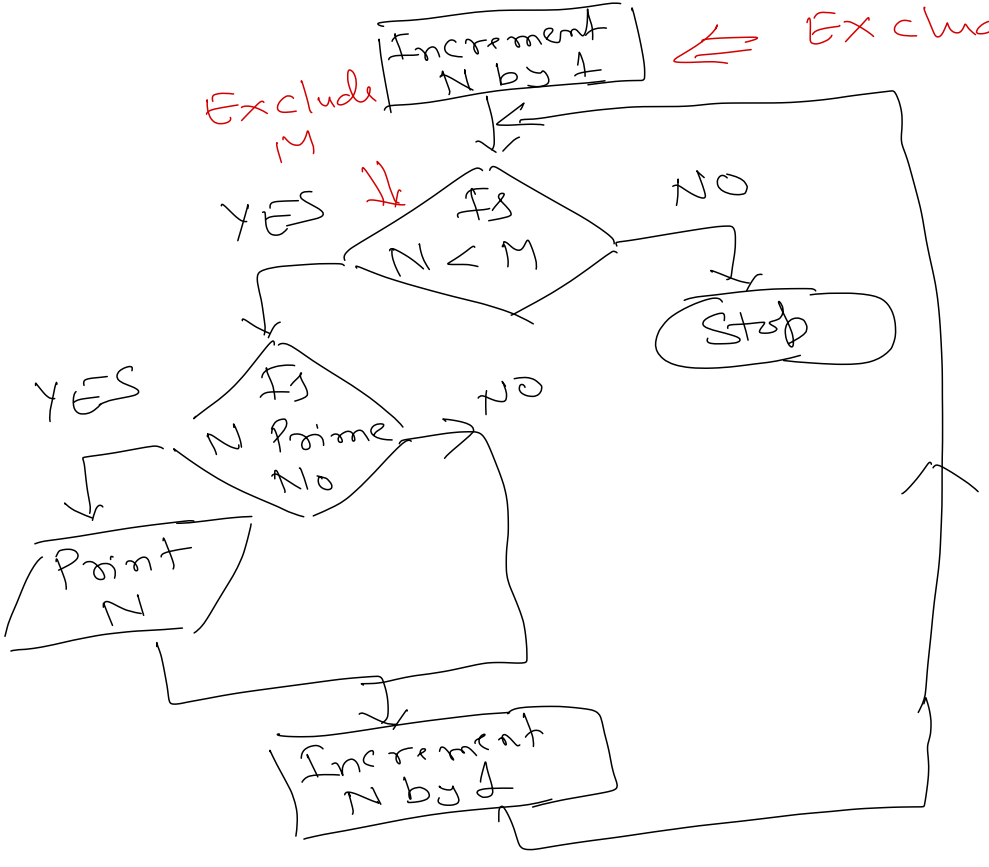
Prime Nos: 3 5

Find Prime numbers between N & M, excluding N & M.

N → 3 4 5 6 7 8

M → 9





$N < M \approx N <= (M-1)$

Reverse a number
Sum of digits
} Positive Numbers only.

Negative Number ?

1. Get Number
2. if Number < 0 then
3. Make Negative Number into Positive.
   ⇒ Multiply Number by -1.

:

C Programming

Language Standards → ANSI

K & R C

ANSI C ←

C99

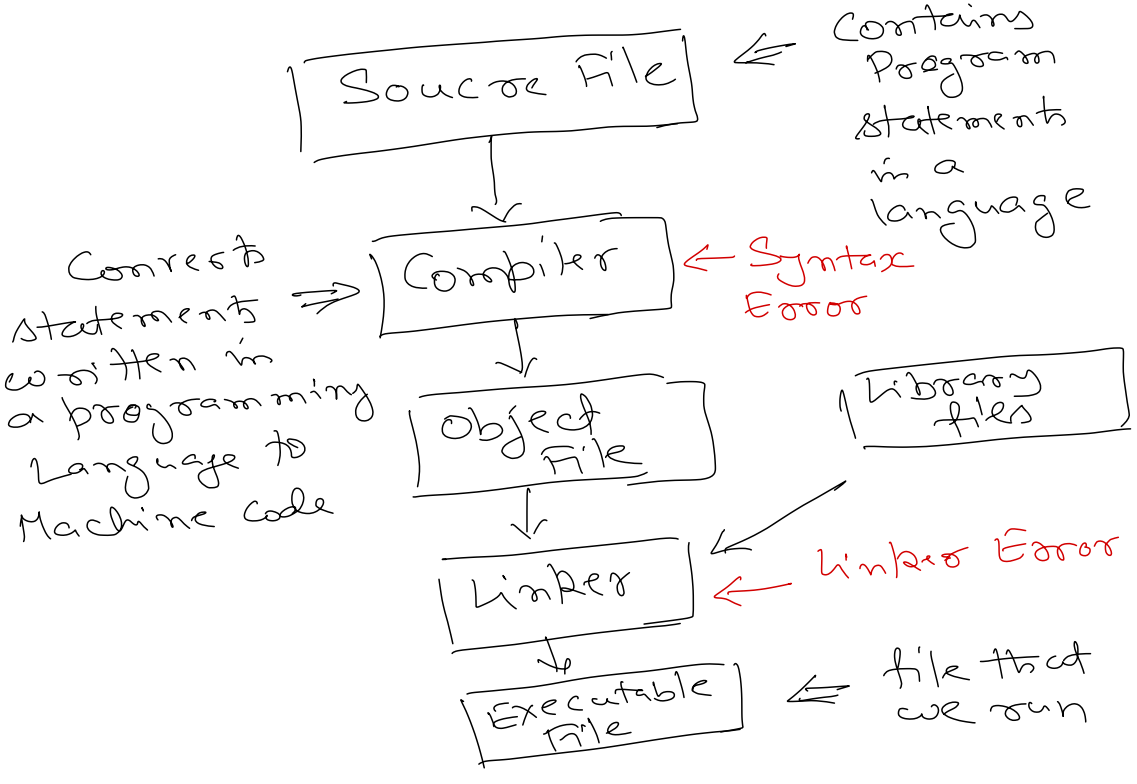
C11

Embeded C

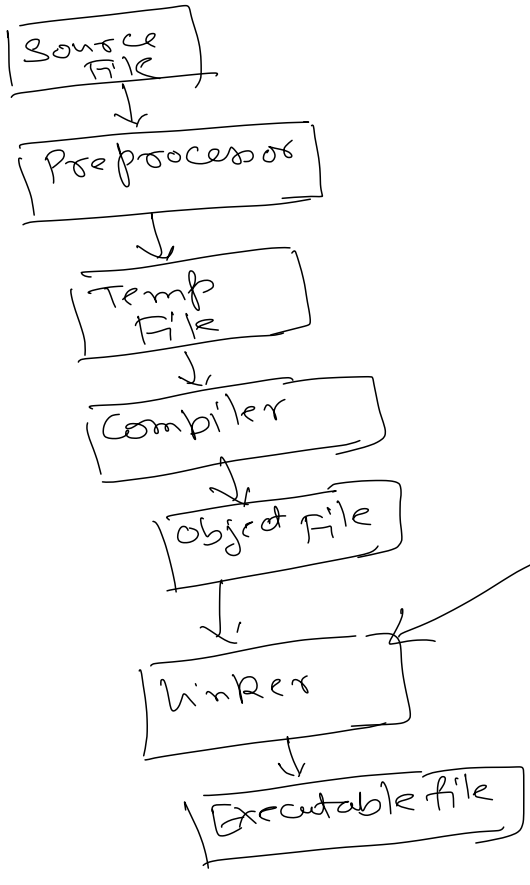
Linux ⇒ gcc

Programming Language

↙ Compiled ↘ Interpreted



C Compilation



Logical Errors

↓

Program not working as expected.

Compiled Language → Source code → Compile → Executable → Execute

Interpreted Language → Source code → Each statement is translated to machine code & executed.

C Program → Is a sequence of instructions / statements

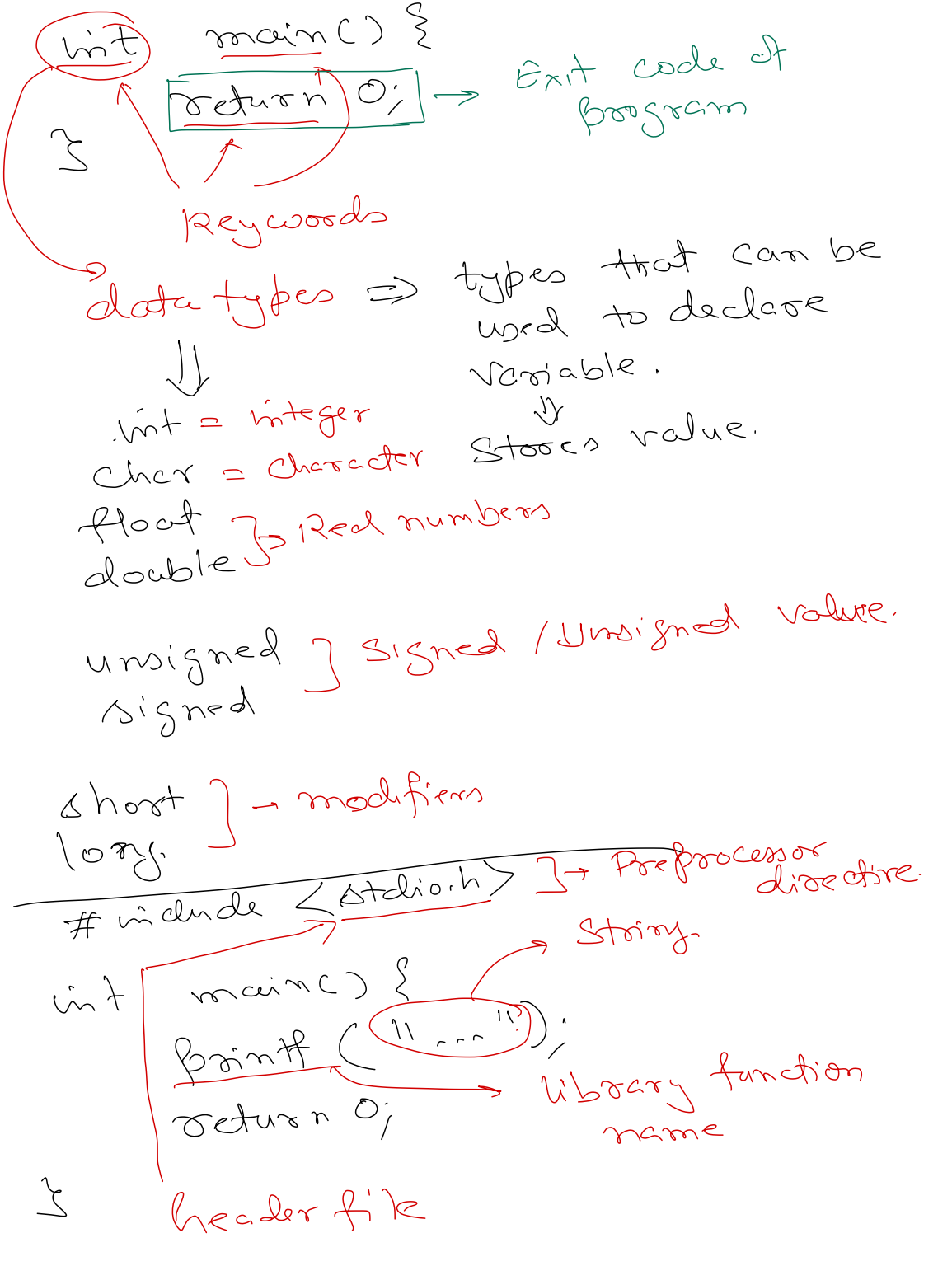
→ Statements are grouped together as functions.

→ Every C program will have a function called main
↓
Entry point of program.

→ C is case sensitive language
"main" diff "MAIN"

→ Keywords ⇒ having predefined meaning in Language

→ operators ⇒ to perform some operations


int
main() {
return 0;
}
Exit code of program
Keywords
data types ⇒ types that can be used to declare variable.
Stores value.
int = integer
char = character
float
double } Real numbers
unsigned
signed } Signed / Unsigned value.
short
long } → modifiers
#include <stdio.h> → Preprocessor directive.
int main() {
printf ("...");
return 0;
}
String.
Library function name
header file