The C Language

Introduction to C language

The C Language

* Made By Dennis Ritchie in 1970s.
* It is used to make games.
Because it is very fast language.
Programs run very fastly. in C Language.

In this Course we learn from basic to Advance

The C Language

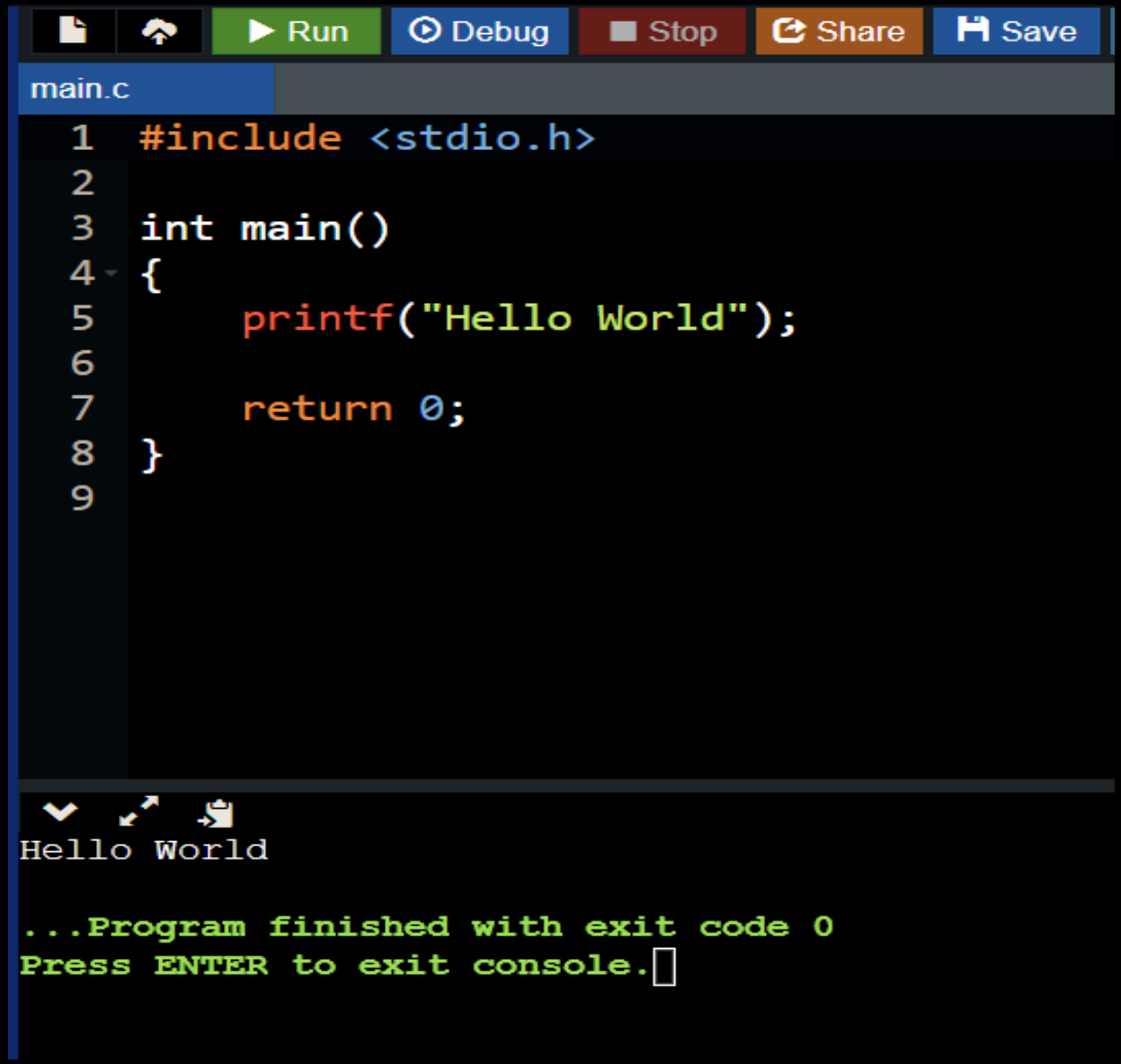
In this video we write our first code

```
#include <stdio.h>
int main()
{
        printf("Hello World\n");
        return 0;
}
```

The C Language

Basic Structure of C Language

The C Language



The C Language

In any C program

Pre processor command, functions, variables, statements, expressions, comments.

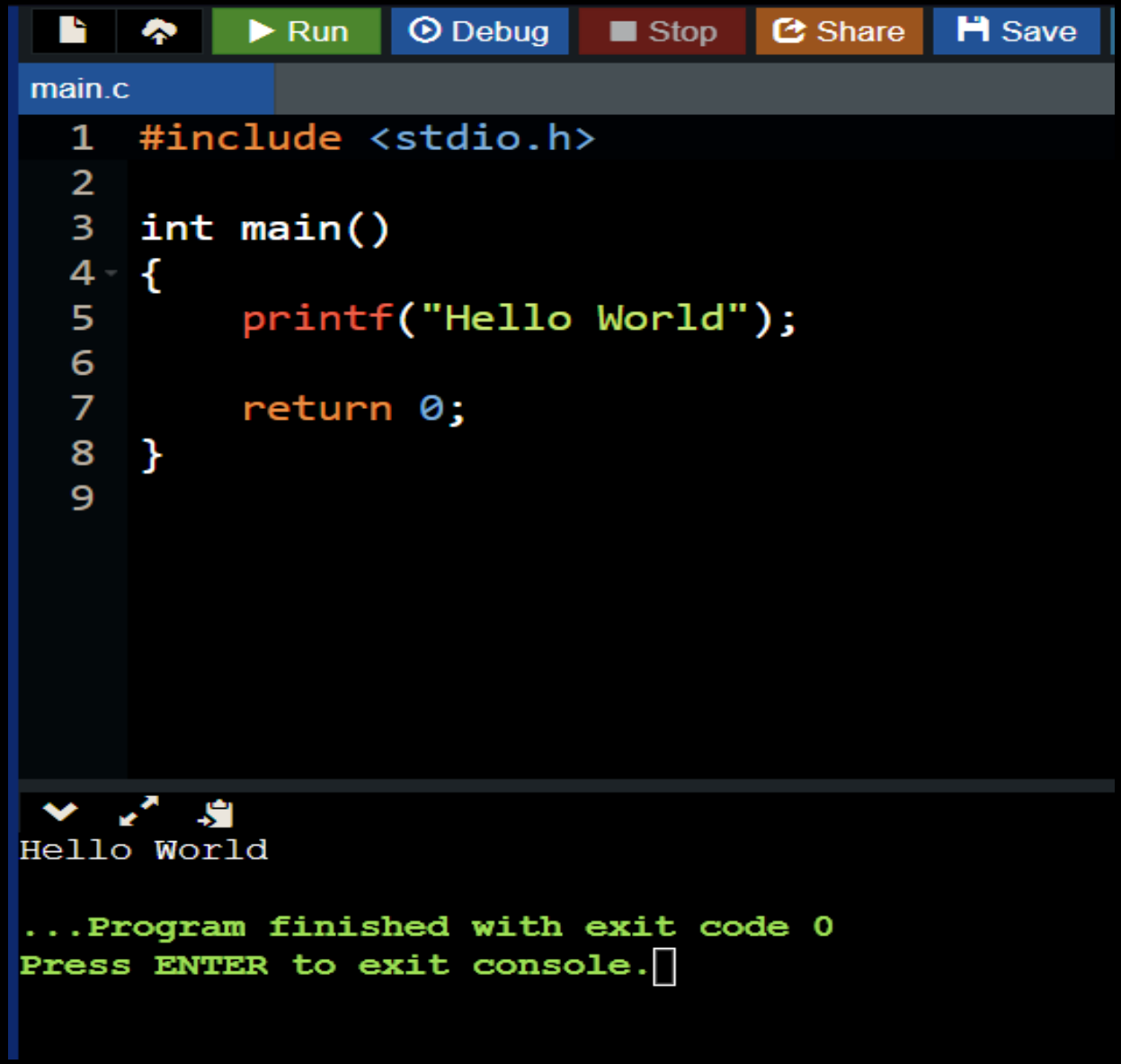
```
#include <stdio.h>  ⟶ pre processor command
                File
int main()  ⟶ function
function → break down a program into parts
printf("Hellow World\n"); → function
return 0; → give integer value to operating System
```

for this

The C Language

Basic Syntax of C Language

The C Language



The C Language

Any C program is made with

Constant, identifiers, string literal, symbol, keyword

are Tokens.

Tokens of

```
printf("Hello World\n")
```

are

```
printf
(
"Hello World\n"
);
```

(;) semicolon is used to terminate program

The C Language

main.c

```c
#include <stdio.h>

int main()
{
    printf
    (
        "Hello World\n"
        )
        ;

    return 0;
}

```

The C Language

Key words → reserved words of C programming
Their are 32 general purpose keyword
you can't make their constant, identifier, variable.

Key word → int, retash

identifiers → name of variable or function
A-Z, a-z, _, letters, numbers

@, $, %	Punctuation characters	C not allowed as identifiers

The C Language

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

The C Language

This is very case sensitive programming language

Variable

h → 1st variable

H → 2nd variable

Symbol → &, %, o, t, -, etc.
comments → //, /*
min. one white space character

In our program

int, return → key word

printf, main → function

0 → constant

"Hello World\n" → string literal

The C Language

Variables and Data types in C Language

The C Language

Variables in C

* A name given to a memory location.
* Declared by writing type variable name;

ex. int a;

int a, b;

* Initialized and declared by

type variable name = value;

ex. int a=4;

int a=4, b=6;

The C Language

Rules for defining a Variable in C Language

→ Can contain alphabets, digits and underscore(_)..
→ A variable name can starts with an alphabet and underscore only.
→ Can't start with a digit.
→ No whitespace and reserved keywords are allowed.
→ Valid Variables:

int harshit, float harshit123, char _harshit34.

→ Invalid Variables: int 77harshit, float $harshit,

char long; int harshitithaarsh.

The C Language

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

The C Language

Data Types in C

* Basic Data Type: int, char, float, double
* Derived Data Type: array, pointer, structure, union
* Enumeration Data Type: enum
* Void Data Type: void

The C Language

Operators in C Language

The C Language

Operators In C

* An operator is a symbol used to perform operations in giving programming language.

* In this tutorial series, wel will look into operators used in C Language.

The C Language

Types of Operators in C Language

* Arithmetic Operators
* Relational Operators
* Logical Operators
* Bitwise Operators
* Assignment Operators

The C Language

▷ Arithmetic Operators

+	Addition
−	Subtration
*	Multiplication
/	Division
%	Modulus

The C Language

▷ Relational Operators

==	Is equal to
!=	Is not equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to

The C Language

- Logical Operators

&&	And operator Both true then true
||	OR operator Any one true then true
!	NOT operator Reverse the logical statement

The C Language

- Bitwise Operators

X	Y	X&Y	X|Y	X^Y	~(X)
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

~ one's complement operator

<< left shift operator

>> right shift operator

The C Language

▷ Assignment Operators

=	right side value equal to left side
+=	Add And assignment operator
-=	Subtract AND assignment operator
*=	Multiply AND assignment operator
/=	Divide AND assignment operator

The C Language

- Miscellaneous Operators

Sizeof( )	Returns the size of variable
&	Return the address of a variable
*	Pointer to a Variable
?:	Conditional Expression

The C Language

Format Specifier & Comments in C Language

The C Language

Format Specifier

* Format specifier is a way to tell compiler what type of data is in a variable during taking input and display output to the user.

* printf("%a.bf", var); will print 'var' with 'b' decimal points in a 'a' character space

The C Language

Format Specifier	Data Type
%f	float
%d	int
%c	char
%l	long
%lf	double
%Lf	long double

The C Language

Comments

```
//      ⟶    Single Line comment
```

```
/*
        ⟶          Multi-line comment


    */
```

The C Language

Escape Sequences and Constants in C Language

The C Language

Escape Sequences in C

* An escape sequence in C programming is a sequence of characters.
* It doesn't represent itself when used in character or string literal.
* It is composed of two or more characters starting with backslash \. For example: \n represents new line.

The C Language

Escape Sequence	Meaning
\n	New Line
\t	Horizontal Tab
\b	BackSpace
\r	Carriage Return
\a	Audible bell
\'	Printing single quotation
\"	printing double quotation
\?	Question Mark Sequence
\\	Back Slash
\f	Form Feed
\v	Vertical Tab
\0	Null Value
\nnn	Print octal value
\xhh	Print Hexadecimal value

The C Language

Constant in C

* Value or Variable that can't be changed in the program,
  Ex. 17, 23, 'a', 3.47, "Coding Version".

* Ways to define constant in C programming
  → const keyword
  → # define preprocessor

The C Language

Exercise 1 : Multiplication Table

The C Language

```
Enter a number you want table of
31
Table of 31
31x1=31
31x2=62
31x3=93
31x4=124
31x5=155
31x6=186
31x7=217
31x8=248
31x9=279
31x10=310


...Program finished with exit code 0
Press ENTER to exit console.
```

The C Language

Exercise 1 : Multiplication Table Solution

The C Language

```
Enter a number you want table of
31
Table of 31
31x1=31
31x2=62
31x3=93
31x4=124
31x5=155
31x6=186
31x7=217
31x8=248
31x9=279
31x10=310


...Program finished with exit code 0
Press ENTER to exit console.
```

Solution Code Link

Click Here

The C Language

If – Else Statement in C Language

The C Language

If - Else statements in C

* It is a control statement.
* It is used to perform operations based on some conditions

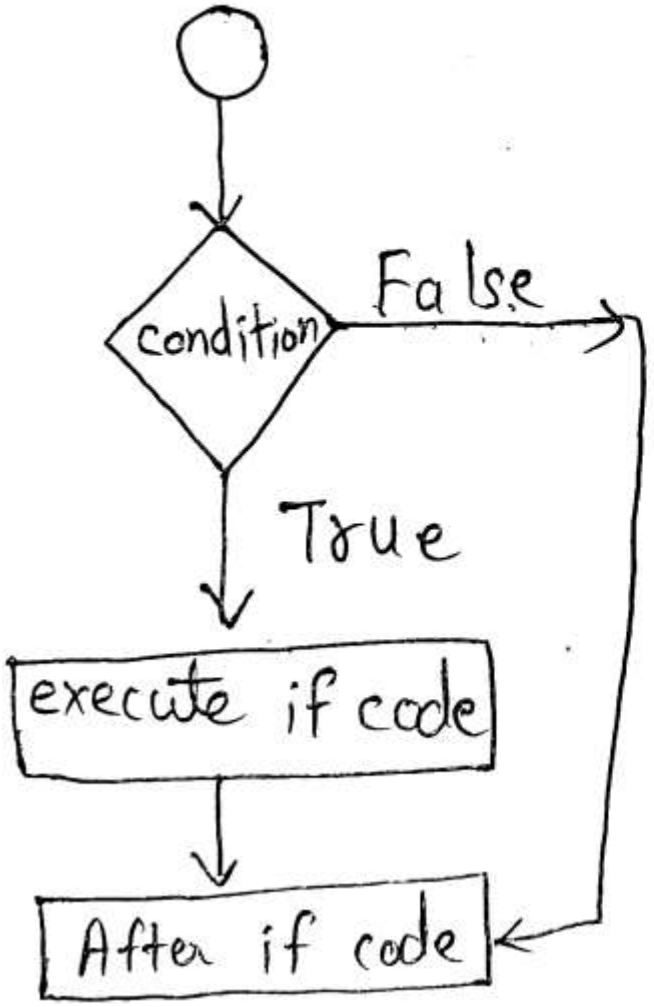
→ Types of statements
- > if statement
- > if else statement
- > if - else if ladder

Syntax for if else

```
if (3<10){
  printf("This will execute");
  }
```

The C Language

Flowchart for if statement

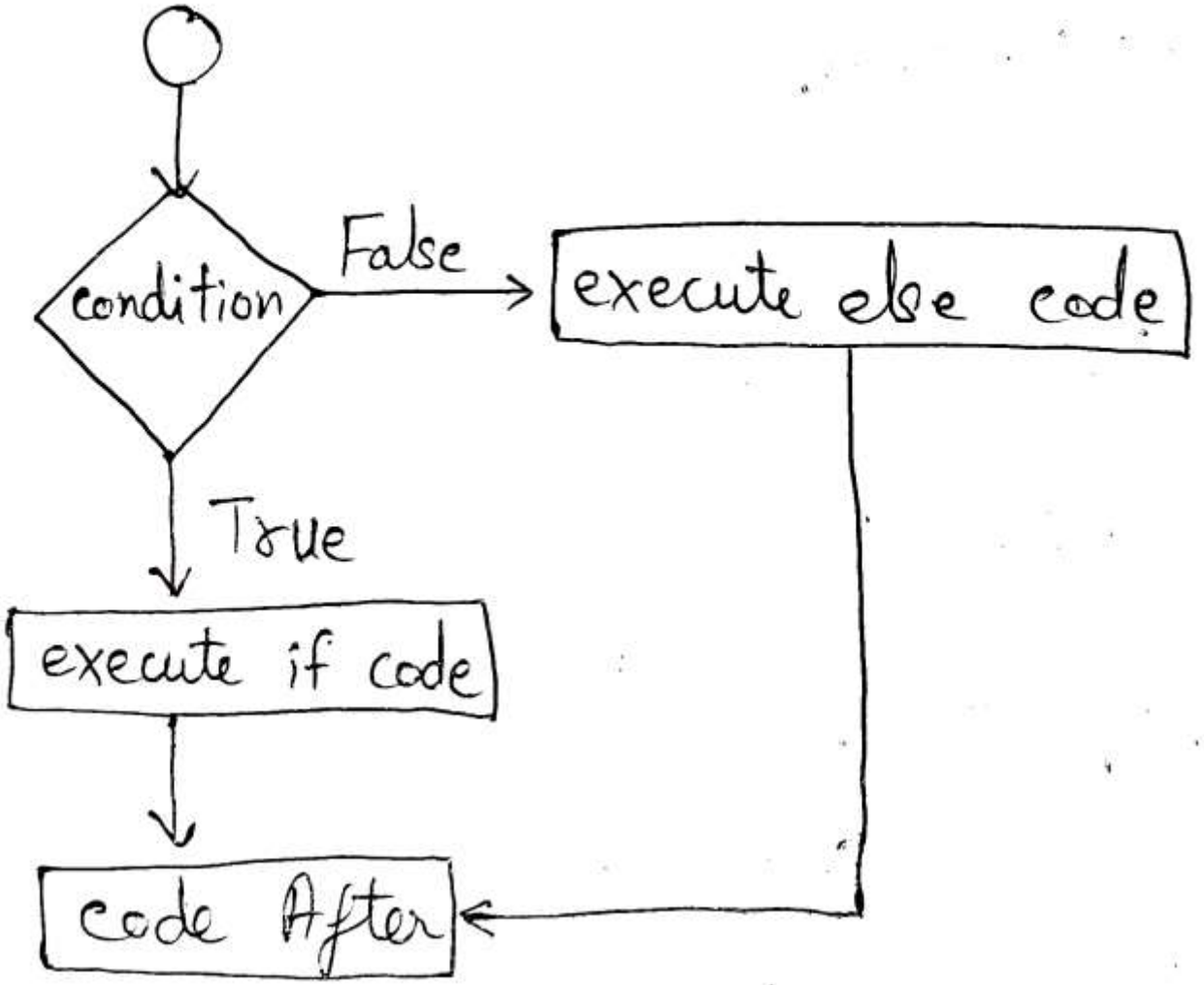


Syntax

```
if (codition){
    code
}
```

The C Language

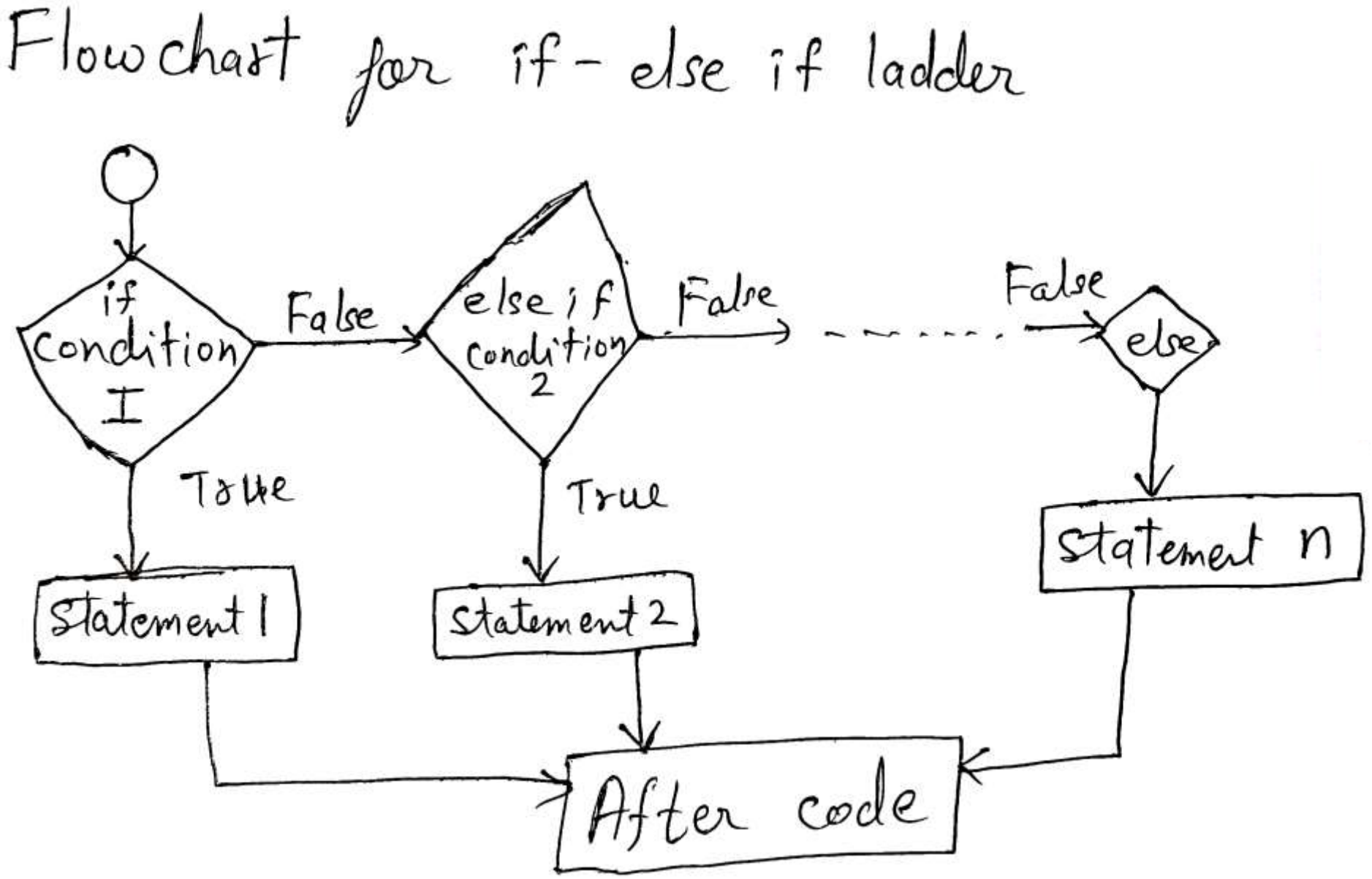
Flowchart for if-else statement



Syntax

```
if (condition){
    code (True) }
else{
    code (False) }
```

The C Language



The C Language

Syntax

```
if (condition 1) {
        code 1 }
else if(condition 2) {
        code 2 }
   !
   !
   !
   else {
        code n
        }
```

The C Language

Switch Case Statement in C Language

The C Language

Switch Case Statements in C

```
int a=2;
switch(a){
    case 2:
        printf("Value is 2");
        break;
    case 3:
        printf("value is 2");

default:
        printf("Nothing matched")
        }
```

The C Language

Rules for Switch Statements

→ Switch expression must be an int or char.
→ Case value must be an integer or a character.
→ Case must come inside switch.
→ Break is not a must.

Valid

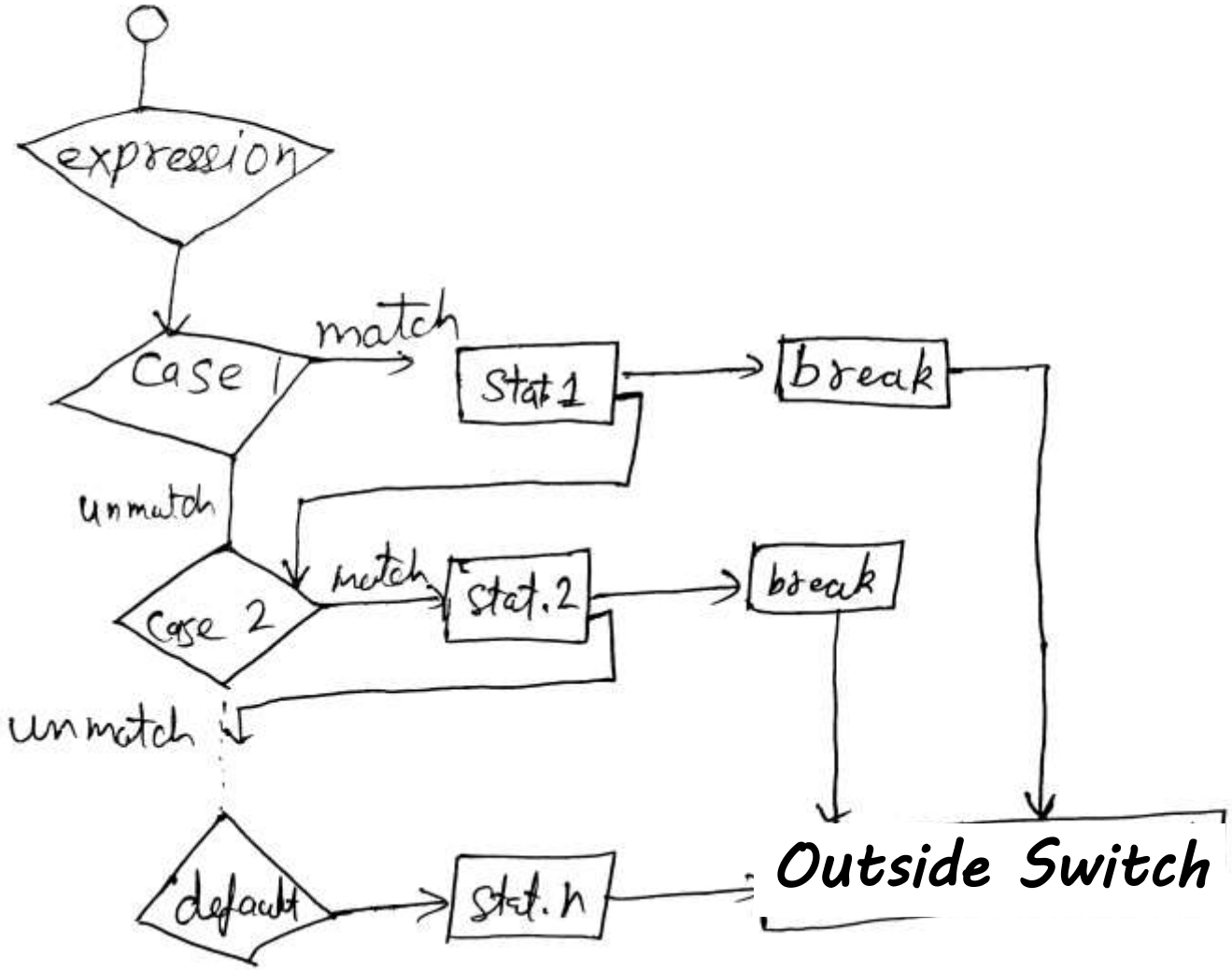
```
int x;
switch(x){
  ==// code+case
  }
```

Invalid

```
float x;
switch(x){
  ==// case+code
  }
```

The C Language

Flowchart for Switch case statement



The C Language

Loops in C Language

The C Language

Loops In C Language

→ Why do we programming?

→ Advantages of using loops

* Code reusability
* making games
* Save Time

→ How loop works?

loop start ⟶ Check condition —False→ exit loop

↓ True

execute loop

The C Language

* Types of Loops

> do while loop

> while loop

> for loop

The C Language

Do While Loop in C Language

The C Language

Do While Loop

```
do {
  // code to be executed
} while (condition)
```

The C Language

Ex.

```
int i=0

do {
    i=i+1;
    printf("%d", i);
} while (i<10);
```

do while loop executes at least once

The C Language

While Loop in C Language

The C Language

While loop in C Language

```
While (condition)
{ // code to be executed
  }
```

Ex.

```
int i=0
while (i<30) {
 print ("%d", i);
 i=i+1; }
```

The C Language

Do While Loop

```
do {
  // code to be executed
} while (condition)
```

While loop in C Language

```
while (condition)
{ // code to be executed
  }
```

The C Language

For Loop in C Language

The C Language

For loop in C Language

~~For~~ for(Expression 1; Expression 2; Expression 3)

{ // code to be executed }

The C Language

Properties of Expression 1

→ The Expression represents the initialization of the loop variable.

→ We can initialize more than one variable in Expression 1

→ Expression 1 is optional.

Properties of Expression 2

→ It is conditional expression. It checks for a specific condition to be satisfied. If it is not, the loop is terminated.

→ It can have more than one condition. However, the loop will iterate until the last condition becomes false. Other conditions will be treated as statements.

→ It is optional.

The C Language

→ Expression 2 can perform the task of expression 1 and expression 3. That is, we can initialize the variable as well as update the loop variable in expression 2 itself.

→ We can pass zero or non-zero value in expression 2. However, in C, any non-zero value is true and zero is false by default.

The C Language

Properties of Expression 3

→ Expression 3 is used to update the loop variable.

→ We can update more than one variable at the same time.

→ Expression 3 is optional.

The C Language

Break & Continue Statement in C Language

The C Language

Break Statement

→ Used to bring the program control out of the loop.

→ The break statement is used inside loops or switch statement.

→ Break statement can be used with

> Loops

> Switch case expression

The C Language

ex. while (codition){

// user enters his name & keeps playing untill game over

if (name == "Satyam"){

break;

} }

The C Language

Continue Statement

→ Used to bring program control to the next iteration of the loop

→ The continue statement skips some code inside the loop and continues with the next iteration.

→ It is mainly used for a condition so that we can skip some lines of code of a particular codition.

The C Language

GOTO Statement in C Language

GOTO Statement

- Also called jump statement in C Language.
- Used to transfer program control to a predefined label.
- It's use is avoided since it causes confusion for the fellow programmers in understanding the code.
- GOTO statement is preferable when we need to break multiple loops using single statement at the same time.

The C Language

Typecasting in C Language

Typecastin

- gt is used to convert Data Type.
- Change the data type of the output.

Syntax

```
(data type)
value;
```