

C Language

Tokens



Saurabh Shukla (MySirG)

Agenda

- ① What are tokens?
- ② Constants
- ③ variables
- ④ keywords
- ⑤ comments

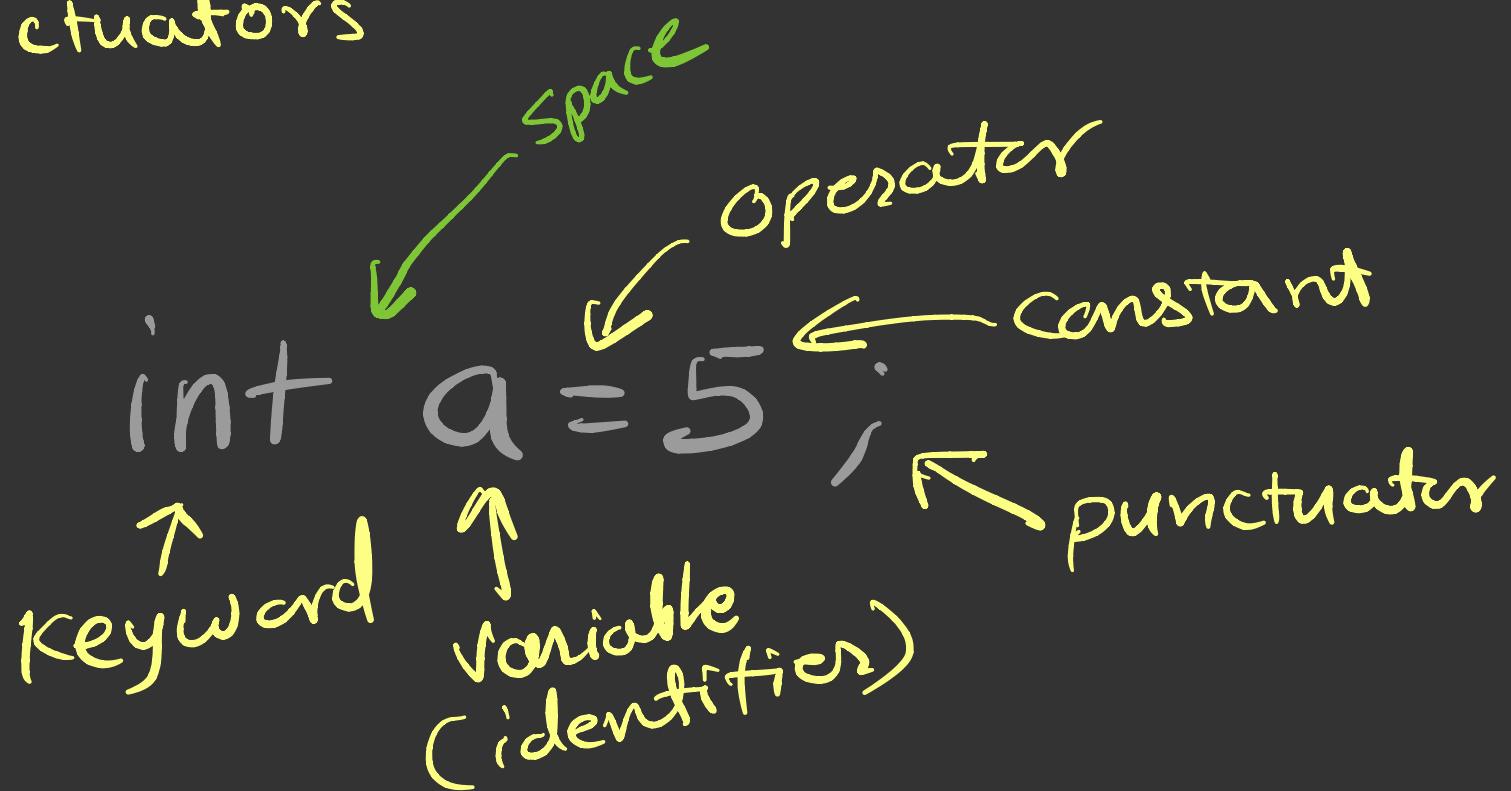
Tokens

- Tokens are also known as lexical elements
- The text of C/C++ program consists of tokens and white space.
- The token is the smallest element of a C/C++ program that is meaningful to the compiler.
- Parser is a program whose job is to separate these tokens.

→ 2nd question

Tokens

- Parser recognizes these kinds of tokens:
 - Constants (Literals)
 - Identifiers (variables, functions ...)
 - Keywords
 - Operators
 - Punctuators



The diagram illustrates the tokens of the C code `int a=5;`. The tokens are:

- `int`: keyword
- : space
- `a`: variable (identifier)
- `=`: operator
- `5`: constant
- `;`: punctuator

White Space

- Tokens are usually separated by white space , which can be one or more
 - Blanks
 - Tabs
 - New lines
 - Form feed
 - Comments

Comments

- Comments are text ignored by compiler
 - Comments are useful for documentation of your code and useful for programmers.

There are two types of comments

- ① Single line comment // text here
 - ② Multiline comment /*
text line 1
text line 2
....
*/

Sample Program in C

```
int main()
{
    int a,b,c; //variables
    printf("Enter two numbers");
    scanf("%d %d", &a, &b);
    c=a+b;
    printf("Sum is %d", c);
}
/* main function body ends here */
```

Operators

Punctuators

Constants

Identifiers

Keywords

Comments

Information = Constants = Data

primary

Integer 25, -417, 0, 3

Real

4.15, -0.05, 63.1, 3.0

Character

'a' 'A' '5' '+'

X ' -3 ' 57 ' 4.5 '

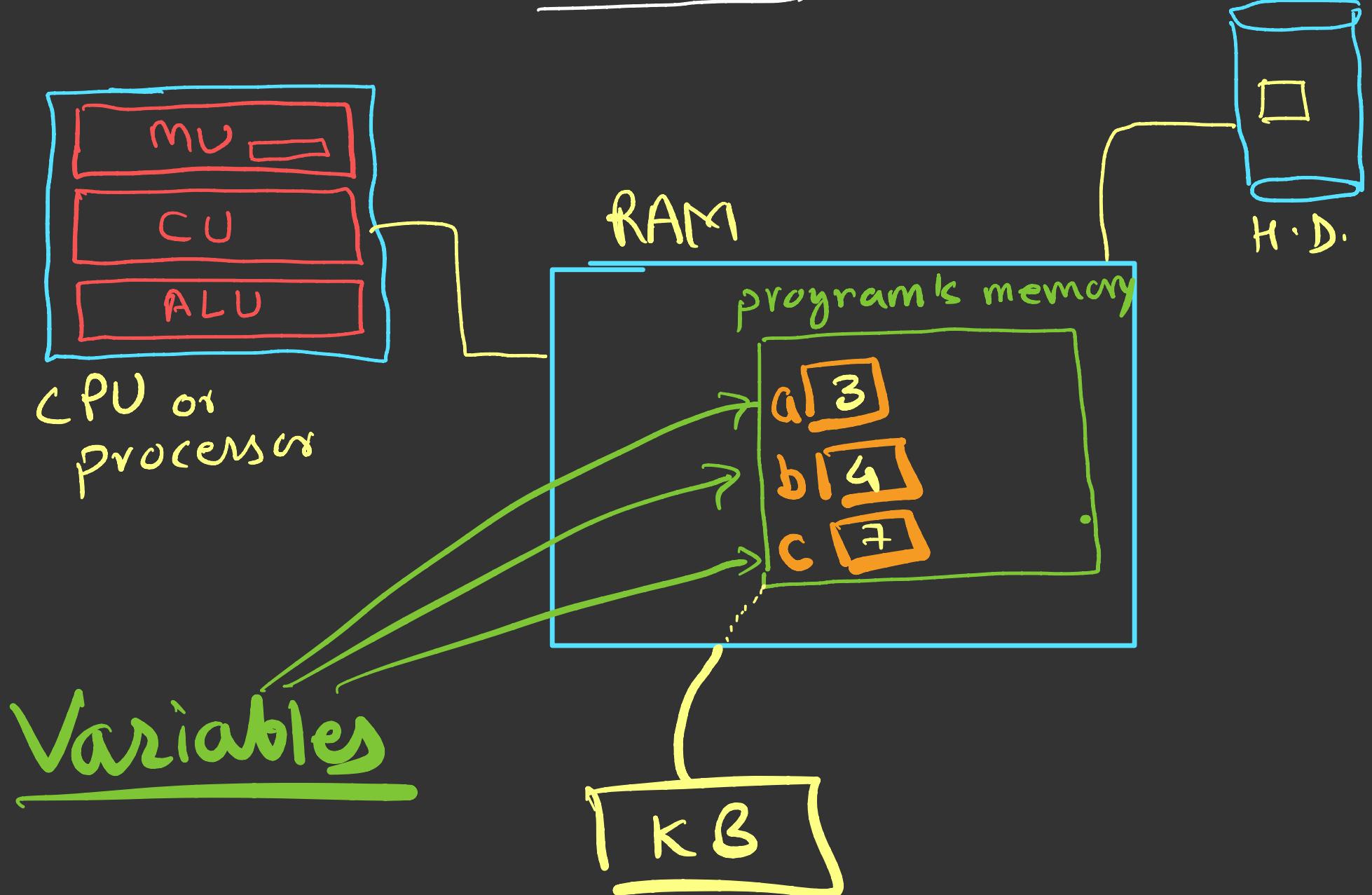
' '

'\n' '\t'

Secondary

- Pointers
- array
- string "Bhopal"
- structure
- union
- enumerator

Variables



Variables

- Variable is a name of memory location where we store program's data during execution of program.
- Variable name is any combination of alphabets (a to z or A to Z), digits (0 to 9) and underscore (_)
- No other symbol is allowed.
- valid variable name cannot start from a digit.
- variable name cannot be a keyword.

a_2 ✓

a_b_2_3 ✓

x-1 ✗

b\$ ✗

s.i ✗

52 ✗

x y ✗

if ✗ (keyword)

_A ✓

IF ✓

2x ✗

c language is a case sensitive
language

C is a case sensitive language

Keywords in C

predefined words / Reserved words

auto
break
case
char
const
continue
default
do

double
else
enum
extern
float
for
goto
if

int
long
register
return
short
signed
sizeof
static

struct
switch
typedef
union
unsigned
void
volatile
while

-Bool
-Complex
-Imaginary
inline
restrict

-Alignas
-Alignof
_Atomic
_Generic
_Noreturn

_Static_assert
_Thread_local

C89
32 keywords

C99
+5

C11
+7