Introduction to C Programming

Course Code: CSL 1022

L-T-P 3 – 0 – 2

Course Contents

- **Introduction:**
- **Fundamentals:**
- **Control Statements:**
- **Arrays:**
- **Functions and pointers:**
- **String:**
- **User defined data types and Additional Features of C:**

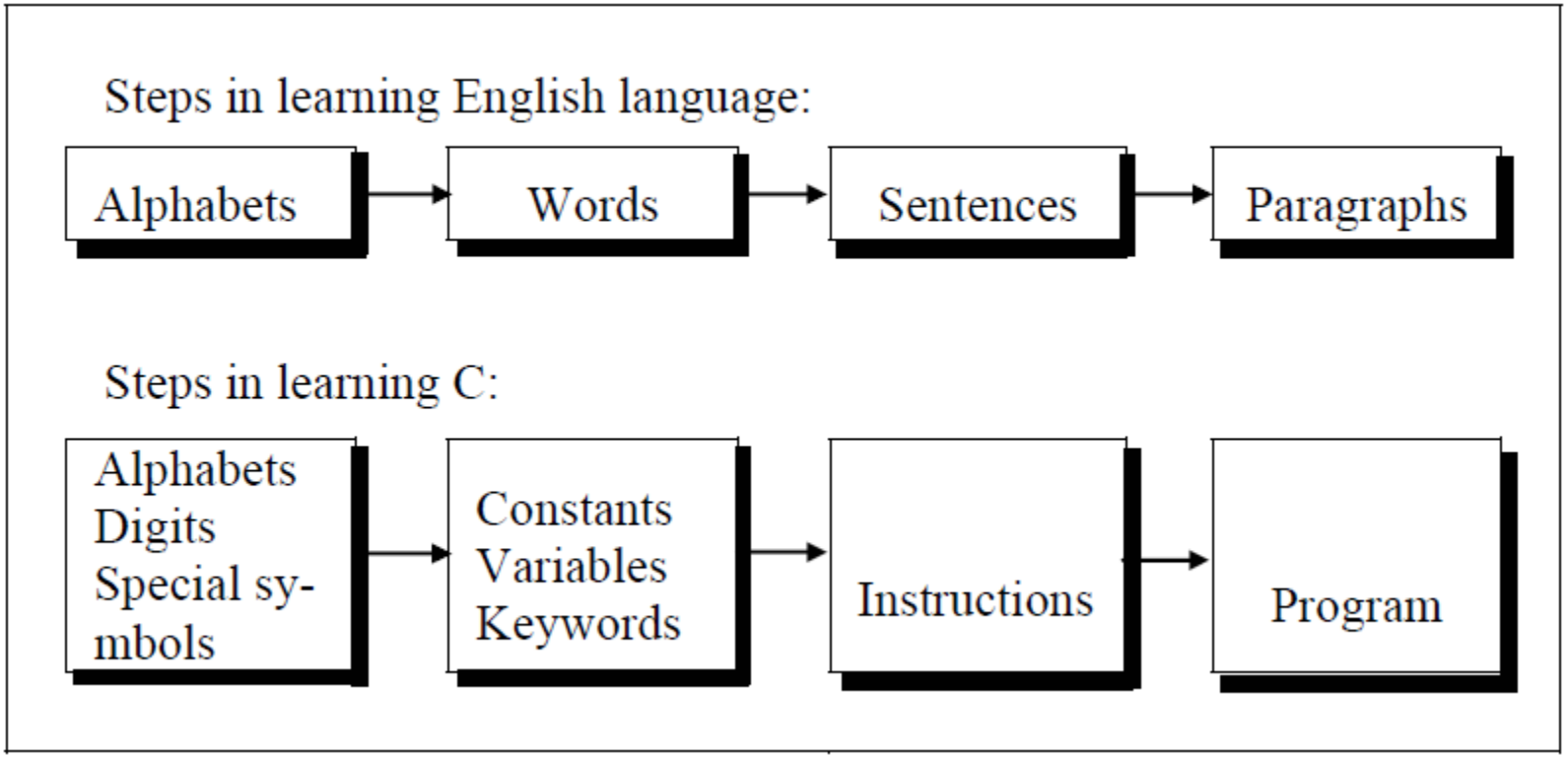
What is C

- C is a programming language developed at AT & T's Bell Laboratories of USA in 1972.
- It was designed and written by Dennis Ritchie.
- Reliable, simple & easy to use

Why bother to learn C today

- For learning C++, C# or Java one still needs a good hold over the language elements of C and the basic programming skills.
- Major parts of OS written in C.
- Consumer devices getting smarter day by day.
- Many popular gaming frameworks have been built using C language.
- C provides several language elements that make this interaction feasible with hardware devices.

C - Fundamentals

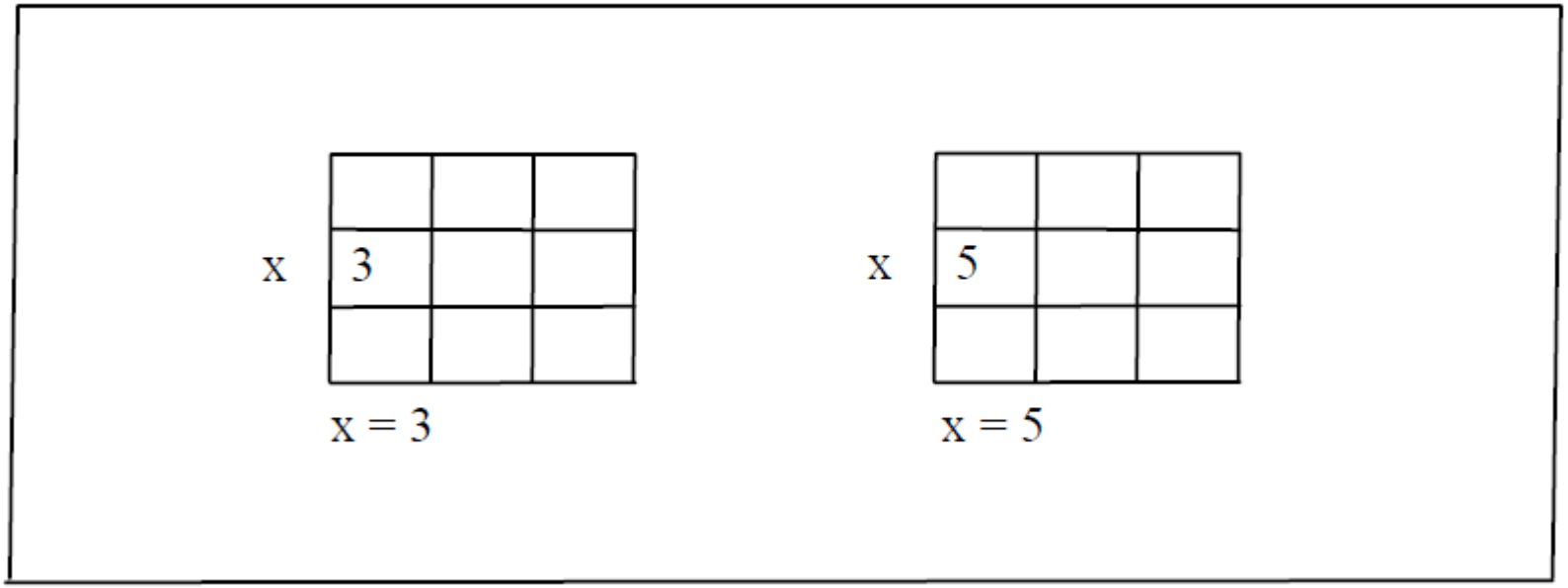


C Character Set

Alphabets	A, B, ….., Y, Z a, b, ……, y, z
Digits	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Special symbols	~ ‘ ! @ # % ^ & * ( ) _ - + = | \ { } [ ] : ; " ' < > , . ? /

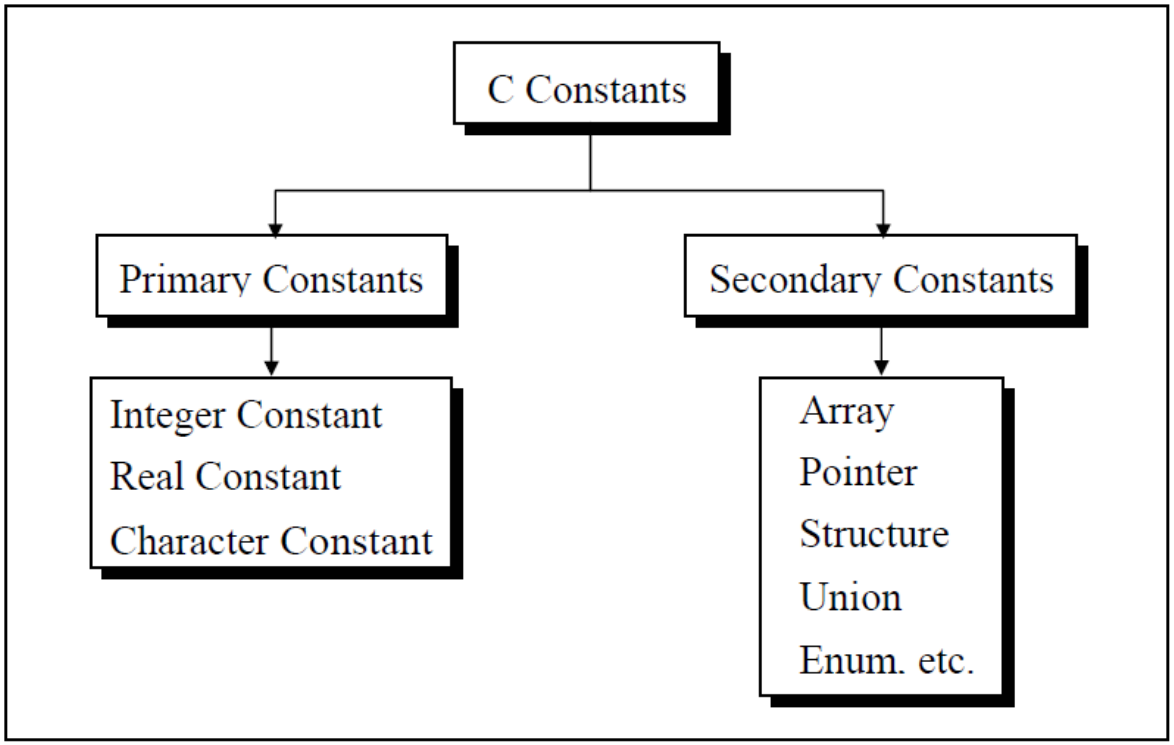
Constants, Variables and Keywords

- The alphabets, numbers and special symbols when properly combined form constants, variables and keywords.



Types of C Constants

- C constants can be divided into two major categories:
  - Primary Constants
  - Secondary Constants



Rules for Constructing Integer Constants

a. An integer constant must have at least one digit.
b. It must not have a decimal point.
c. It can be either positive or negative.
d. If no sign precedes an integer constant it is assumed to be positive.
e. No commas or blanks are allowed within an integer constant.
f. The allowable range for integer constants is -2147483648 to +2147483647.

- Ex.: 426
  +782
  -8000
  -7605

Rules for Constructing Real Constants

- A real constant must have at least one digit.
- It must have a decimal point.
- It could be either positive or negative.
- Default sign is positive.
- No commas or blanks are allowed within a real constant

- Ex.: +325.34
  426.0
  -32.76
  -48.5792

Rules for constructing real constants expressed in exponential form

- The mantissa part and the exponential part should be separated by a letter e or E.
- The mantissa part may have a positive or negative sign.
- Default sign of mantissa part is positive.
- The exponent must have at least one digit, which must be a positive or negative integer. Default sign is positive.
- Range of real constants expressed in exponential form is -3.4e38 to 3.4e38.

- Ex.: +3.2e-5
  4.1e8
  -0.2e+3
  -3.2e-5

Rules for Constructing Character Constants

- A character constant is a single alphabet, a single digit or a single special symbol enclosed within single inverted commas.
- Both the inverted commas should point to the left. For example, ’A’ is a valid character constant whereas ‘A’ is not.
- Ex.:

'A'
'I'
'5'
'='

Exercise 1. Question

- Which of the following are invalid C constants and why?

1. ‘3.15’
2. 35,500
3. 3.25e2
4. 2e-3
5. ‘eLearning’
6. “show”
7. ‘Quest’
8. 2^3
9. 4 6 5 3

Exercise 1. Solution

- Which of the following are invalid C constants and why?

1. ‘3.15’
2. 35,500
3. 3.25e2
4. 2e-3
5. ‘eLearning’
6. “show”
7. ‘Quest’
8. 2^3
9. 4 6 5 3

Rules for Constructing Variable Names

- A variable name is any combination of 1 to 31 alphabets, digits or underscores. Some compilers allow variable names whose length could be up to 247 characters. Still, it would be safer to stick to the rule of 31 characters. Do not create unnecessarily long variable names as it adds to your typing effort.
- The first character in the variable name must be an alphabet or underscore.
- No commas or blanks are allowed within a variable name.
- No special symbol other than an underscore (as in **gross_sal**) can be used in a variable name.
- Ex.:

    Si_int

    m_hra

    pop_e_89

Principle,

Roi,

noy

Rules for Constructing Variable Names

- Examples of type declaration statements:
- Ex.:

```
int si, m_hra ;
float bassal ;
char code ;
```

C Keywords

- words whose meaning has already been explained to the C compiler.
- **cannot** be used as variable names .
- The keywords are also called ‘Reserved words’.
- 32 keywords

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

Exercise 2: Question

Which of the following are invalid variable names and Why?

- B’day
- int
- $hello
- #HASH
- Dot.
- Number
- totalArea
- _main()
- Temp_in_Deg
- Total%
- 1st
- Stack-queue
- Variable name
- %name%
- salary

Exercise 2: Solution

Which of the following are invalid variable names and Why?

- B’day
- int
- $hello
- #HASH
- Dot.
- Number
- totalArea
- _main()
- Temp_in_Deg
- Total%
- 1st
- Stack-queue
- Variable name
- %name%
- salary