



C Programming

Trainer: Ketan G Kore



About Trainer

- Mr. Ketan G Kore
 - M.Sc.Computer Science
- Training
 - PreCAT batches
 - Email: ketan.kore@sunbeaminfo.com
 - Mobile: 8805822402



History

- C language was developed by Dennis Ritchie in 1972 at AT & T Bell Labs on PDP-11 machine.
- It was developed while porting UNIX from PDP-7 to PDP-11.
- Many features of C are inspired from B (Ken Thompson) and BCPL (Martin Richards).
- Initial release of C is referred as K & R C.

Standardization

- C was standardized by ANSI in 1989. This is referred as C89.
- Standardization ensures C code to remain portable.
- C standard is revised multiple times to add new features in the language.
 - C89 – First ANSI standard
 - C90 – ANSI standard adopted by ISO
 - C99 – Added few C++ features like bool, inline, etc.
 - C11 – Added multi-threading feature.
 - C17 – Few technical corrections.



Hello World

- Source Code

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

- Commands

- cmd> gcc hello.c
- cmd> ./a.exe



Compilation & Execution of C program



Hello World

- printf() – library function
- `stdio.h` – header file
- `main()` – entry point function
 - `void main() { ... }`
 - `int main() { ... }`
 - `int main(void) { ... }`
 - `int main(int argc, char *argv[]) { ... }`
 - `int main(int argc, char *argv[], char *envp[]) { ... }`
- return 0 – exit status



Toolchain & IDE

- Toolchain is set of tools to convert high level language program to machine level code.
 - Preprocessor
 - Compiler
 - Assembler
 - Linker
 - Debugger
 - Utilities
- Popular compiler (toolchains)
 - GCC
 - Visual Studio
- IDE – Integrated development environment
 - Visual Studio
 - Eclipse
 - VS Code (+ gcc)
 - Turbo C
 - Anjuta, KDevelop, Codeblocks, Dev C++, etc.



Introduction

- High-level
- Compiled
- Procedural
- Block-Structured (control structures).
- Typed
- Library Functions



Strengths

- Low level memory access (pointers, data structures)
- Effective memory access (bitwise operators, bit-fields, unions)
- Can access OS features (functions/commands)
- Extensive library functions (math, strings, file IO, ...)
- Compilers for different platforms & architectures
- Highly Readable (macros, enum, functions, ...)



Applications

- System programming
 - OS development
 - Device drivers
 - System utilities
- Embedded programming
 - ARM, AVR, PIC, etc.
 - IoT development
- Language development
 - Compiler development
- Achievements (tiobe.com)
 - In top-2 languages in last 40 years.
 - Language of year: 2019, 2017, 2008.



Tokens

- C program is made up of functions.
- Function is made up of statements.
- Statement contain multiple tokens.
 - Keywords
 - Data Types
 - Identifiers
 - Variables
 - Constants
 - Operators



Keywords

- Keywords are predefined words used in program, which have special meanings to the compiler.
- They are reserved words, so cannot be used as identifier.
- K & R C has 27 keywords. C89 added 5 keywords. C99 added 5 new keywords.

C89 = 32

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



Identifiers

- Identifiers give names to variables, functions, defined types and pre-processor macros.
- Rules of Identifiers:
 - Should start with alphabet or with _ (underscore)
 - Can include alphabets, _ (underscore), digits
 - Case sensitive
- Examples:
 - Var_1 //Valid
 - 1_var // Not Valid
 - _var //valid
 - Var-1 // invalid
 - Basic Salary //invalid



Data Types, Variables & Constants

- C allows computations to be performed on various types of data.
 - Numerical: Whole numbers, Real numbers
 - Character: Single character, Strings
- Fixed data values are said to be constants.
 - 12, -45, 0, 2.3, 76.9, 1.23456e+2, 'A', "Sunbeam", etc.
- Data is hold in memory locations identified by names called as variables.
 - Variable must be declared before its use in the program.
 - As per need, variable have some data type.
- Simple C data types are: int, double, char.
 - Data type represents amount of space assigned to the variable.
 - It also defines internal storage of the data.



printf()

- Arbitrary strings and variable values can be printed using printf() function.
 - int - %d
 - double - %lf
 - char - %c
 - ...
- Examples:
 - printf("Hello PreCAT @ Sunbeam");
 - printf("%d", roll_number);
 - printf("%d %lf %c", number, basic_salary, letter);
 - printf("Book price is %lf", price);
- Escape sequences
 - \n, \r, \t, \b, \a, \\, %, \", \'



Data Types

- Data type defines storage space and format of variable.
- Primitive types
 - int
 - short
 - long
 - char
 - float
 - double
- Integer types can be signed/unsigned
- Derived types
 - Array
 - Pointer
 - Function
- Type qualifiers
 - const and volatile
- printf() format specifiers
 - %d, %u, %o, %x
 - %hd, %hu
 - %ld, %lu
 - %c
 - %f, %e
 - %lf
- User defined types
 - struct
 - union
 - enum
- void type – represent no value.



Data Types, Variables & Constants

- Variable examples
 - `int number = 10;`
 - `double basic_salary = 20000.0;`
 - `char letter = 'A';`
 - `int roll_number;`
 - `roll_number = 20;`
 - `double price = 200.0;`
 - `price = 300.0;`
- Constant examples
 - `-23, 1L, 34U, 3UL, 0x41, 0101,`
 - `1.234f, 1.234567e+2, ...`
 - `'A', '\101', '\x41'`
 - `"SunBeam", "A\101\x41"`
- Each variable is assigned some memory location.
- Size of data type of given variable or constant is found by `sizeof()` operator.



Data Types

- Data type defines storage space and format of variable.
- Primitive types
 - int
 - short
 - long
 - char
 - float
 - double
- Integer types can be signed/unsigned
- Derived types
 - Array
 - Pointer
 - Function
- Type qualifiers
 - const and volatile
- printf() format specifiers
 - %d, %u, %o, %x
 - %hd, %hu
 - %ld, %lu
 - %c
 - %f, %e
 - %lf
- User defined types
 - struct
 - union
 - enum
- void type – represent no value.



Data Types

C Basic Data Types	32-bit CPU		64-bit CPU	
	Size (bytes)	Range	Size (bytes)	Range
char	1	-128 to 127	1	-128 to 127
short	2	-32,768 to 32,767	2	-32,768 to 32,767
int	4	-2,147,483,648 to 2,147,483,647	4	-2,147,483,648 to 2,147,483,647
long	4	-2,147,483,648 to 2,147,483,647	8	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
long long	8	9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	8	9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
float	4	3.4E +/- 38	4	3.4E +/- 38
double	8	1.7E +/- 308	8	1.7E +/- 308





Thank you!

