Getting Started

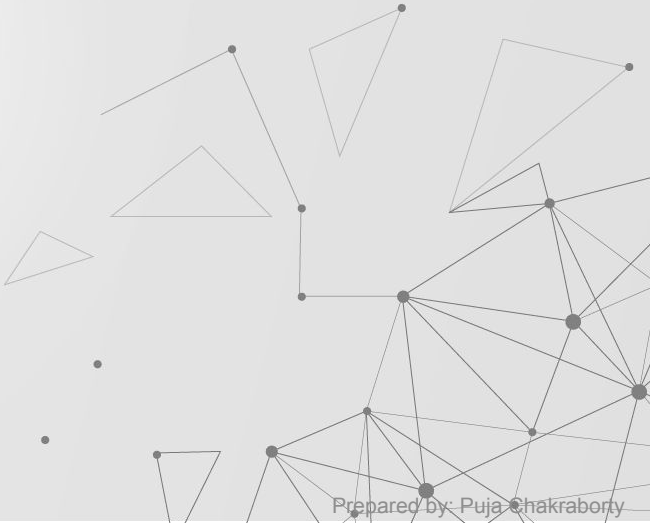
with

C

Prepared by: Puja Chakraborty

Programming Language: C

- High Level Programming Language
  - Simple
  - Fast
  - Machine Independent
  - Rich Library



Programming Language: C

- Developed by Dennis Ritchie

  In 1972

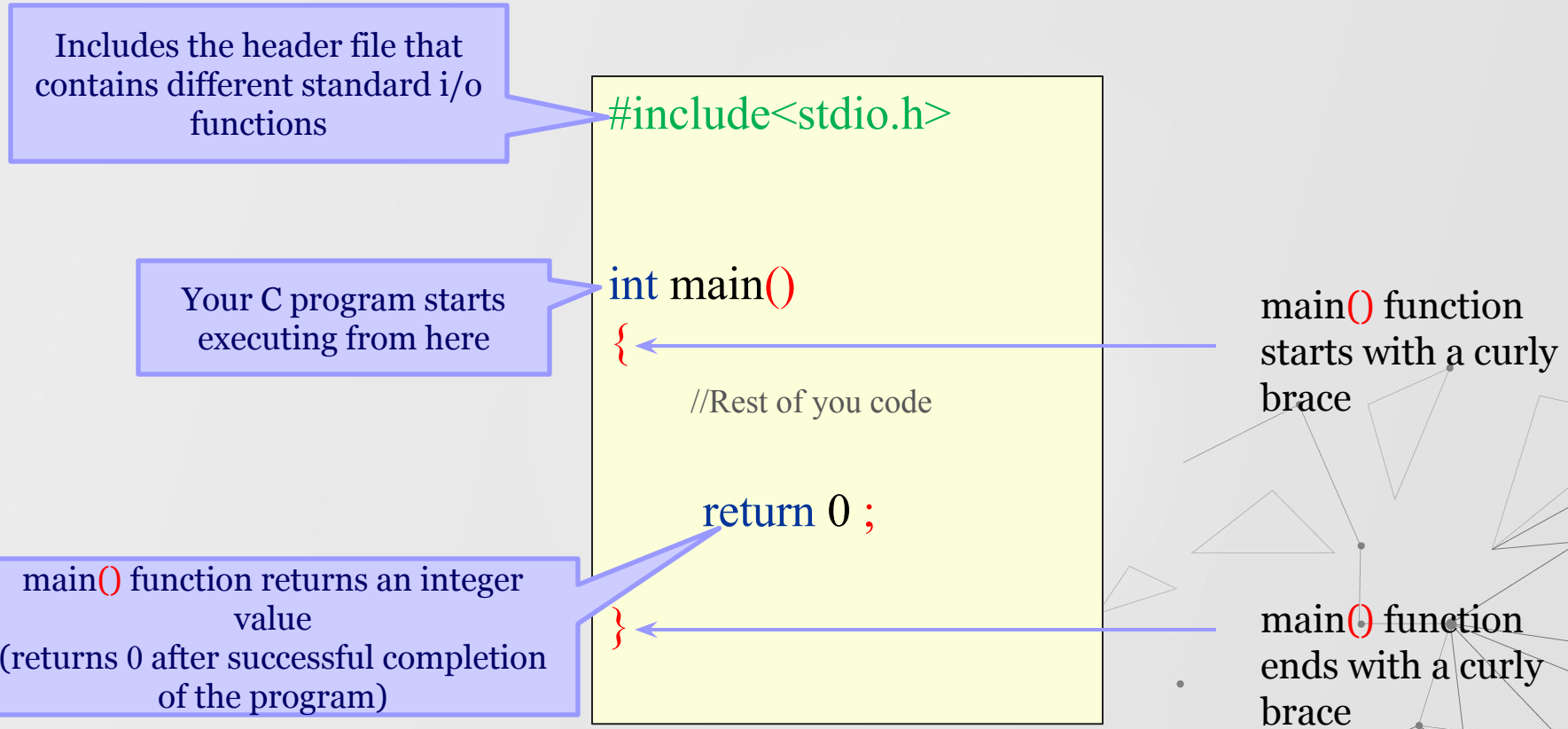
Basic Structure of a C program

```
#include<stdio.h>

int main()
{

        return 0;
}
```

Basic Structure of a C program



printf() function

- A standard output function
- In built library function, defined in stdio.h (header file)

Syntax of printf() function

```
printf("The text you want to display", argument list);
```

optional

printf() function

```
#include<stdio.h>

int main()
{
        printf("Welcome to C Programming");
        return 0;
}
```

Output of the Program:

Welcome to C Programming

Tokens in C

- Token is
  - a keyword ,
  - an identifier ,
  - a constant,
  - a string literal or
  - Symbol.

printf(“Welcome to C Programming”);

How many tokens are present in this statement?

Tokens in C

Statement: printf(“Welcome to C Programming”);

Tokens in this statement:

printf
(
“Welcome to C Programming”
)
;

Total 5 tokens are present in this statements.

Semicolons in C

- In a C program, the semicolon is a statement terminator
- Each individual statement **must** be ended with a semicolon

```
#include<stdio.h>

int main()
{
        printf("Welcome to C Programming");
        return 0;
}
```

End of statements

Keywords in C

- A keyword is a **reserved word**.
- A keyword can not be used as a variable name, constant name, etc

32 keywords in C

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

Identifiers in C

 An identifier is a name used to identify a variable, function, or any other user-defined item

❖ Rules of defining identifiers:

 An identifier can contain alphabets, digits, and underscore.

 An identifier name can start with the alphabet, and underscore only. It can't start with a digit.

No whitespace is allowed within the identifier.

 An identifier name must not be any reserved word or keyword, e.g. int, float, etc.

Identifiers in C

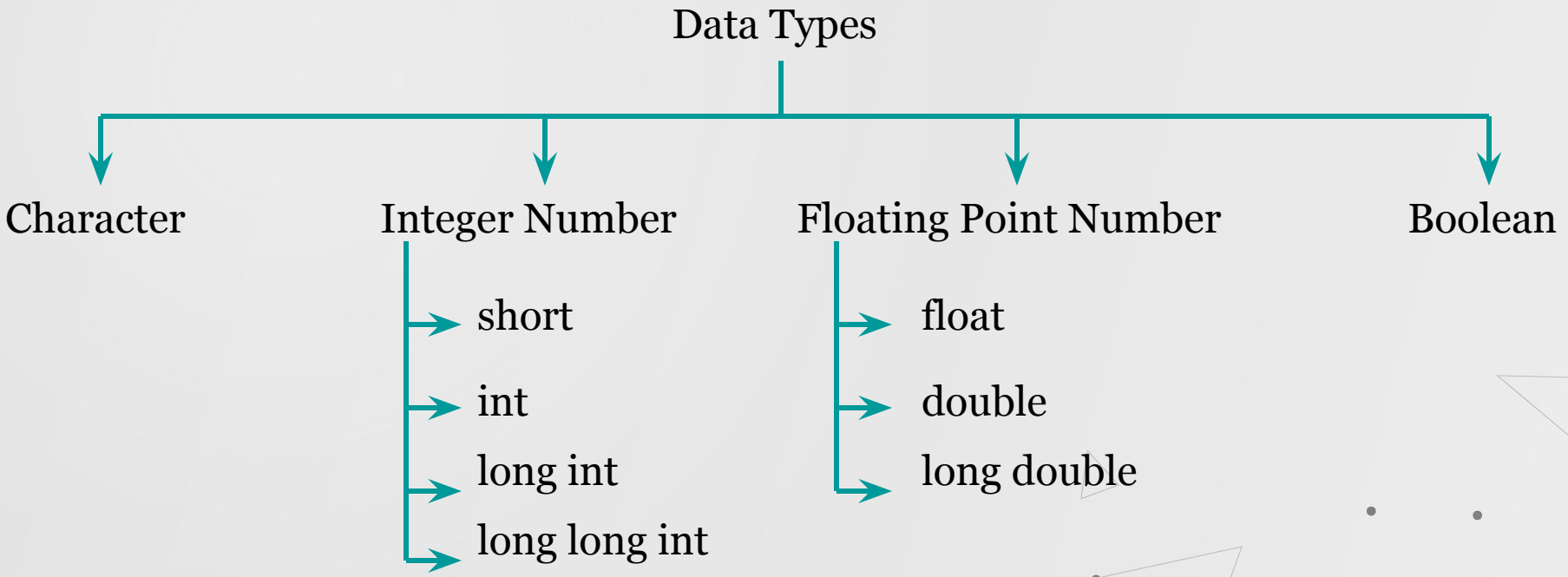
Some valid identifier

abc
xyz_12
_a1
myVariable
myVariable2
myIdentifier_3

Some invalid identifier

10abc
double
_a 13
my Variable
my%Variable
my@Identifier_3

Basic Data Types in C



Basic Data Types in C

Data Type	Size (Byte)	Range
short int	2	-32,768 **to** 32,767
unsigned short int	2	0 **to** 65,535
int	4	-2,147,483,648 **to** 2,147,483,647
unsigned int	4	0 **to** 4,294,967,295
long int	4	-2,147,483,648 **to** 2,147,483,647
unsigned long int	4	0 **to** 4,294,967,295
long long int	8	-(2^63) **to** (2^63)-1
unsigned long long int	8	0 **to** 18,446,744,073,709,551,615

Basic Data Types in C

Data Type	Size (Byte)	Range
signed char	1	-128 **to** 127
unsigned char	1	0 **to** 255
float	4	1.2E-38 **to** 3.4E+38
double	8	2.3E-308 **to** 1.7E+308
long double	12	3.4E-4932 **to** 1.1E+4932
bool	1	

Variables in C

- A **variable** is a name of the memory location.
- It is used to store data.
- Its value can be changed, and it can be reused many times.
- It is used to store data.

Syntax to declare a variable

```
Data_type variable_Name ;
```

For exapmle:

```
int x ;
float f1 ;
double f2 ;
```

Operators in C

An operator is simply a symbol that is used to perform operations.

❖Types of operators:

- Arithmetic Operator (+ , -, *, /, %)
- Relational Operator (>, <, >=, <=, == , !=)
- Logical Operator (||, && , !)
- Assignment Operator (=, +=, -=, /=)
- Bitwise Operator , Shift Operator, Conditional Operator etc.

Operators in C (Precedence and Associativity)

❖Precedence:

If more than one operators are involved in an expression, C language has a predefined rule of priority for the operators.

This rule of priority of operators is called operator precedence.

❖Associativity:

If two operators of same precedence (priority) is present in an expression, Associativity of operators indicate the order in which they execute.

Operators in C (Precedence and Associativity)

High

Precedence

Low

Category	Operator	Associativity
Postfix	() [] -> . ++ - -	Left to right
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise OR	|	Left to right
Logical AND	&&	Left to right
Logical OR	||	Left to right
Assignment	= += -= *= /= %=	Right to left

Format Specifier in C

The Format specifier is a string used in the formatted input and output functions.

Data Type	Format Specifier
short	%hd
int	%d or, %i
long int	%ld
long long int	%lld
unsigned short	%hu
unsigned int	%u
unsigned long long int	%llu

Data Type	Format Specifier
float	%f
double	%lf
long double	%Lf
char	%c
char sequence (string)	%s
Hexadecimal integer	%x
Octal Integer	%o

Escape Sequence in C

An escape sequence is a sequence of characters that doesn't represent itself when used inside string literal or character

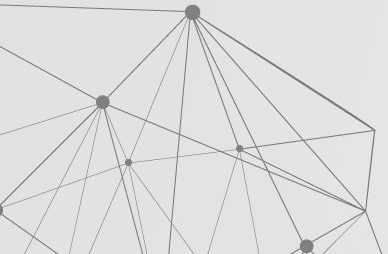
Sequence	Meaning
\b	Backspace
\f	Form Feed
\n	Newline
\t	Horizontal Tab
\v	Vertical Tab
\\	Backslash
\’	Single Quote
\”	Double Quote
\?	Question Mark

scanf() function

The **scanf()** function is used for input. It reads the input data from the console.

Syntax of scanf() function

```
scanf("Format String", argument list);
```



scanf() function

Syntax of scanf() function

```
scanf("Format String", argument list);
```

Taking input (one integer):

```
scanf("%d",  &a);
```

Taking input (two float):

```
scanf("%f %f",  &x, &y);
```

Taking input (one integer and one char):

```
scanf("%d %c",  &n, &c);
```

Thank You

Instructor Information:

Puja Chakraborty
Lecturer
Department of Computer Science and Engineering
Premier University
Chattogram, Bangladesh
Email: puja.csecu@gmail.com
Contact: +880-1863-927559