

Typedef and Union in C

Talk to a Teacher

<http://spoken-tutorial.org>

National Mission on Education through ICT

<http://sakshat.ac.in>

Ashwini R Patil

IIT Bombay

7 July 2014



Learning Objectives



Learning Objectives

We will learn about



Learning Objectives

We will learn about

- ▶ **typedef keyword**



Learning Objectives

We will learn about

- ▶ **typedef keyword**
- ▶ **union keyword**



System Requirements



System Requirements

- ▶ **Ubuntu Operating System version 11.10**



System Requirements

- ▶ **Ubuntu Operating System version 11.10**
- ▶ **gcc Compiler version 4.6.1 on Ubuntu**



Prerequisites



Prerequisites

You should be familiar with



Prerequisites

You should be familiar with

- ▶ C tutorials



Prerequisites

You should be familiar with

- ▶ C tutorials
- ▶ For relevant tutorials please visit our website: <http://spoken-tutorial.org>



Introduction to typedef



Introduction to typedef

- ▶ Used to give symbolic name to an existing type or user defined datatypes



Introduction to typedef

- ▶ Used to give symbolic name to an existing type or user defined datatypes
- ▶ It is a way to define alias to the commands



Introduction to typedef

- ▶ Used to give symbolic name to an existing type or user defined datatypes
- ▶ It is a way to define alias to the commands
- ▶ It helps to provide clarity to the code



Introduction to typedef

- ▶ Used to give symbolic name to an existing type or user defined datatypes
- ▶ It is a way to define alias to the commands
- ▶ It helps to provide clarity to the code
- ▶ It makes the code easier to understand and change



Talk to a Teacher

Syntax



Syntax

- ▶ **typedef existing_name alias_name**
- ▶ **ex:**
 - ▶ **typedef unsigned int uint;**



Introduction to Union



Introduction to Union

- ▶ **Union is a collection of different datatypes grouped together**



Introduction to Union

- ▶ **Union is a collection of different datatypes grouped together**
- ▶ **Union allocates one common storage space for all its members**



Introduction to Union

- ▶ **Union is a collection of different datatypes grouped together**
- ▶ **Union allocates one common storage space for all its members**
- ▶ **We can access only one member of union at a time**



Syntax1



Syntax1

```
▶ union union_name  
{  
    members;  
    members;  
}  
union_variable;
```



Syntax2



Syntax2

► **union union_name**

{

members;

members;

};

union union_name union_variable;



Difference



Difference

- ▶ **Union allocates a common storage space for all its members**



Difference

- ▶ **Union allocates a common storage space for all its members**
- ▶ **Structure allocates separate storage space for all its members**



Difference

- ▶ **Union allocates a common storage space for all its members**
- ▶ **Structure allocates separate storage space for all its members**
- ▶ **Union occupies lower memory space**



Difference

- ▶ **Union allocates a common storage space for all its members**
- ▶ **Structure allocates separate storage space for all its members**
- ▶ **Union occupies lower memory space**
- ▶ **Structure occupies higher memory space**



Union example



Union example

```
▶ union student
{
    int marks;
    char name[6];
    double average;
};
```



Union example



Union example

- ▶ **Memory allocation for union variable will be 8bytes**
- ▶ **As double data type will occupy the maximum memory space**



Structure example



Structure example

```
▶ struct student
{
    int marks;
    char name[6];
    double average;
};
```



Structure example



Structure example

- ▶ **Memory allocation for structure variable will be:**
- ▶ **$2\text{bytes} + 6\text{bytes} + 8\text{bytes} = 16\text{bytes}$**



Talk to a Teacher

Summary



Summary

We learnt,

- ▶ **typedef**
- ▶ **union**
- ▶ **Difference between union and structure**



Assignment

- ▶ Write a program, to display records of an employee
- ▶ Like name, address, salary
- ▶ Define a union named employee
- ▶ Give an alias name as emp using typedef



Talk to a Teacher

About the Spoken Tutorial Project

- ▶ Watch the video available at http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
- ▶ It summarises the Spoken Tutorial project
- ▶ If you do not have good bandwidth, you can download and watch it



Spoken Tutorial Workshops

The Spoken Tutorial Project Team

- ▶ Conducts workshops using spoken tutorials
- ▶ Gives certificates to those who pass an online test
- ▶ For more details, please write to contact@spoken-tutorial.org



Acknowledgements

- ▶ **Spoken Tutorial Project is a part of the Talk to a Teacher project**
- ▶ **It is supported by the National Mission on Education through ICT, MHRD, Government of India**
- ▶ **More information on this Mission is available at:**

<http://spoken-tutorial.org/NMEICT-Intro>

