

Phys4051: C Lecture 1

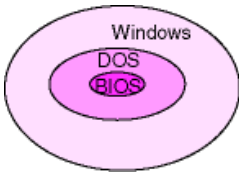
Intro

Overview

- Software: Operating Systems
- Programs
- Programming Languages
- C Syntax

2

Operating System (OS)

- Purpose:
 - Software & Hardware Interface
- Example:
- Other Operating Systems:
 - Unix (Linux), MacOS
 - OS2, VMS

3

OS: Process & Multitasking

- Process (Program)
 - Threads
- Timeslicing (WindowsNT)
 - OS Interrupts Threads
- Preemptive
 - Thread Voluntarily Relinquishes Control
 - Selfish Threads

4

Files & Common File Extensions

- Executable
 - Command Files (*.com)
 - Batch Files (*.bat)
 - COMPILED Programs (*.exe)
- Non-Executable
 - Data Files (*.dat, *.bin)
 - Text Files (*.txt)
 - Images (*.jpg, *.bmp, *.gif)
 - Program Source Code (*.c, *.cpp, *.jpp)

5

Creating an Executable File

- Compiled Code
 - (ASCII) Source code is compiled into a standalone executable file.
 - Example: C
 - LabWindows Environment
- Interpreted Code
 - An interpreter (i.e., a program) compiles the code (usually by line by line) at run time.
 - Example: BASIC
 - Java Virtual Machine (VM)

6

Computer Languages

- Low Level / High Level
 - Fortran, C, C++
 - Assembly Language
 - Machine Language
- OOP (Objet Oriented Programming) / Non OOP
 - C / C++

7

High Level Languages

	Used in Physics	OOP	WWW	
C				
C++				
Java				
Basic				
Fortran				

8

C, C++, Java

- C is a subset of C++
- Java uses C syntax but in addition is OOP

9

C Language

- Keywords:
 - Example: main, for, while, if, else..
- Variables (Properties)
 - Example: integers, floating point...
- Operators
 - Example: +, -, *, /, <, &, |, ~, !...
- Functions (Methods)
 - ANSI C vs. User Defined Functions

10

First C Example

- Program:

```
#include <stdio.h>
main()
{
    printf("Hello");
}
```
- Output:
Hello

11

C Syntax

- C is **CASE SeNsITivE!**
 - Example:

```
int volt = 1;
int Volt = 2;
int VolT = 3;
int VOLT = 4;
int volt = 5, VoLt = 6;
```

12

C Syntax: Functions

- The C language uses a function based syntax.
- A C program consists of various functions that may call other functions.
- Every C program must contain one function called *main()*. This is where the program starts!

13

C Syntax: Most Simple Function

- Function Declaration:

```
_____;
```
- Function Header:

```
_____
```
- Statement(s):

```
{  
_____  
}
```

14

C Syntax: Functions

- Function Declaration:

```
#include <stdio.h>
```
- Function Header:

```
main()
```
- Statement(s):

```
{  
printf("Hello");  
}
```

15



C Functions

- A C function can have any number of arguments (or none) but it can return at most only one value.
- Built in (ANSI C) functions vs. user defined ones

16



Function Declarations

- Declarations Consist of:
 - Function Return Type
 - Function Name (First letter Uppercase)
 - Arguments (variable type and name)
- Sample Function Declarations:

```
float Calc( int x, float y);  
void ClearScreen( void );
```

17



Function Body: Header & Statements

- Function Header
 - Identical to function declaration (except for semicolon!)
- Statements
 - All statements end in semicolon!

18



Function Example: KeV

Declaration `float KeV(float V);`

Header `float KeV(float V)`
`{`

Statements `float ke;`
`ke = 1.6E-16 * V;`
`return(ke);`
`}`

19



Function Example: KeV (2)

Declaration `float KeV(float V);`

Function Call `main()`
`{`
`float fke;`
`float fv = 123.0;`
`ke = KeV(fv);`
`}`

Function Body `float KeV(float V)`
`{`
`float ke; ...etc, rest identical to`
`previous example`
`}`

20



Function Example: KeV

Declaration

Header `main()`
`{`

Statements `float ke;`
`float V = 100.0;`
`ke = 1.6E-16 * V;`
`}`

21



Include Files

- The file "stdio.h" contains the function declaration for the "printf()" function.

Program:

```
#include<stdio.h>
main()
{
printf("Hello");
}
```

22
