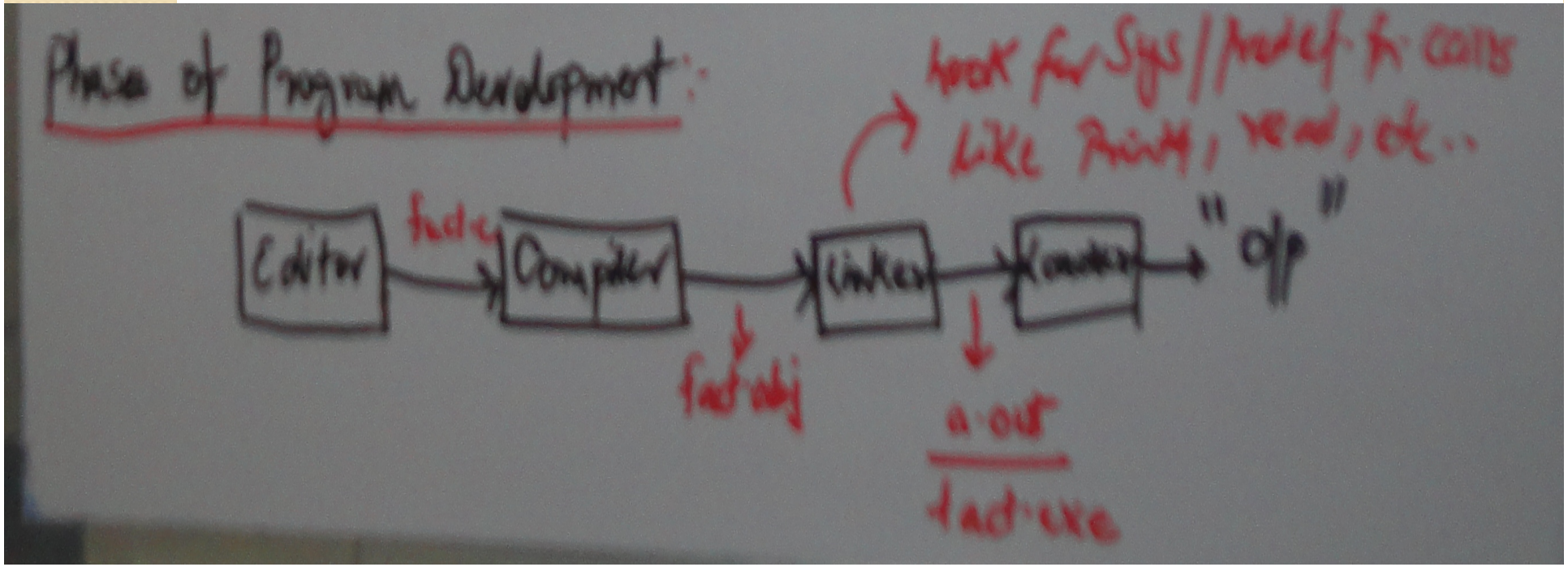
OPERATING SYSTEMS – Major Features

OS Functionalities (technical) – Course Modules

- **Process Concept** – Process Management – System Calls fork(), etc. Processing; Multiprogramming; Multi tasking; Multiprocessing
- **Multithreading** – Threaded Execution of Processes – POSIX library, pthreads, thread scheduling ,etc.
- **Inter Process Communication** (IPC) – exchange of data across processes, pipe() ; SHM, etc.
- **Process / Thread Synchronization –** semaphores, monitors, **…**
- **Memory Management –** paging, segmentation, virtual memory, etc.
- **Device Management** – IO Read, Write system calls etc.

PROCESS CONCEPT - Intro

- **PROCESS NOTION**
- Program in execution- Programs RUNS!!! – understand the terminology !
- Active Program (Process) v/s Binaries-Exe's on HDD (passive!)
- Program here is not Source Code!
- **Phases of PROGRAM Development (BINARY)**



Binary Development Phases

- **Revisit of the Classical Hello World Example!**

```
#include <stdio.h>
int main() {
printf ("Hello World \n");
return 0; }      /* Assume Stored in first.c */
```

- Editors in Linux – gedit, emacs, vim, etc.
- these are again programs later become processes
- Each block in the phases is a program - process
- Header File Inclusion Why ? – System v/s User Defined Headers
- What do They Contain – Function Declaration or Function Definition ; int add (int , int) ; v/s
- int add (int x, int y) { return x+y;}
- Benefit of Having Declaration and Then Definition
- Where is Printf Defined ? - Role of Linker Justification

A Visual Understanding of the phases