**KARNATAK LAW SOCIETY’S**

**GOGTE INSTITUTE OF TECHNOLOGY**

**UDYAMBAG, BELAGAVI – 590008**

**(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)**

**(Approved By AICTE, New Delhi)**

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

**COURSE PROJECT**

**OBJECT ORIENTED MODELING AND DESIGN**

**Submitted by**

Rachana Kampli 2GI18IS032

Laxmi Nyamagoud 2GI18IS020

Hemanth I T 2GI18IS015

Guided By: Prof. Pushpa Tamase

**CERTIFICATE**



This is to certify that **Ms. Rachana Kampli, Ms. Laxmi Nyamagoud, Mr. Hemanth I T,** of **Fifth Semester** bearing **USN: 2GI18IS032, 2GI18IS020, 2GI18IS015** has satisfactorily completed the course in Course activity of Unix System Programming. It can be considered as a bonafide work carried out for partial fulfilment of the academic requirement of 5th Semester B.E.(Information Science & Engineering) prescribed by KLS Gogte Institute of Technology, Belagavi during the academic year 2020- 2021.

The report has been approved as it satisfies the academic requirements prescribed for the said degree.

Signature of the Faculty Member Signature of the HOD

Date:/12/2020

**Course project report and ppt content**

* Title
* Problem statement for that the project
* Objectives of Defined Problem statement
* Design / Algorithm/Flowchart/Methodology
* Implementation details/Function/Procedures/Classes and Objects (Language/Tools)
* Working model of the final solution
* Report and Oral Presentation skill

**Marks allocation:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Batch No. : | | | | | |
| 1. | Project Title: | Marks  Range | USN | | | |
|  |  |  |  |
| 2. | Problem statement (PO2) | 0-1 |  |  |  |  |
| 3. | Objectives of Defined Problem statement  (PO1,PO2) | 0-2 |  |  |  |  |
| 4. | Design / Algorithm/Flowchart/Methodology  (PO3) | 0-3 |  |  |  |  |
| 5. | Implementation  details/Function/Procedures/Classes and Objects (Language/Tools) (PO1,PO3,PO4,PO5) | 0-4 |  |  |  |  |
| 6. | Working model of the final solution  (PO3,PO12) | 0-5 |  |  |  |  |
| 7. | Report and Oral presentation skill (PO9,PO10) | 0-5 |  |  |  |  |
|  | Total | 20 |  |  |  |  |

**\* 20 marks is converted to 10 marks for CGPA calculation**

* **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
* **Problem Analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and Engineering sciences.
* **Design/Development of solutions:**Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
* **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
* **Modern tool usage :**Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
* **The engineer and society:**Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
* **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
* **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
* **Individual and team work:** Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.
* **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
* **Project management and finance:** Demonstrate knowledge and understanding of the engineering management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
* **Life-long learning:** Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

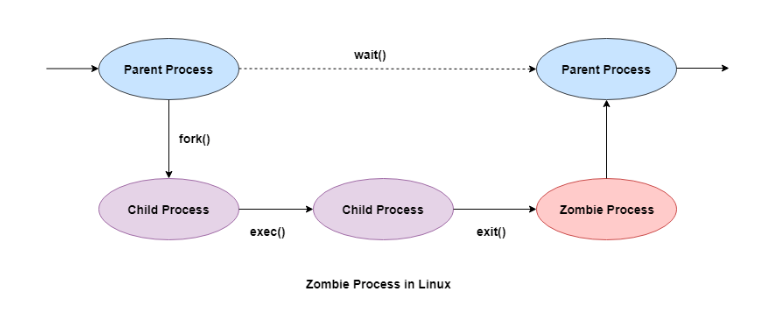
Title: ZOMBIE PROCESS

Abstract:

Zombie processes that are specific to Unix and Unix-like systems. Usually, they are not too dangerous, but in some cases, zombie processes can cause resource leakage and can be a sign of a bug in a program or an operating system.

Definition:

A zombie process is a process whose execution is completed but it still has an entry in the process table. Zombie processes usually occur for child processes, as the parent process still needs to read its child’s exit status.



Salient points of Zombie Processes :

* The exit status of the zombie process can be read by the parent process using the wait() system call. After that, the zombie process is removed from the system. Then the process ID and the process table entry of the zombie process can be reused.
* If the parent process does not use the wait() system call, the zombie process is left in the process table. This creates a resource leak.
* The zombie processes can be removed from the system by sending the SIGCHLD signal to the parent, using the kill command.

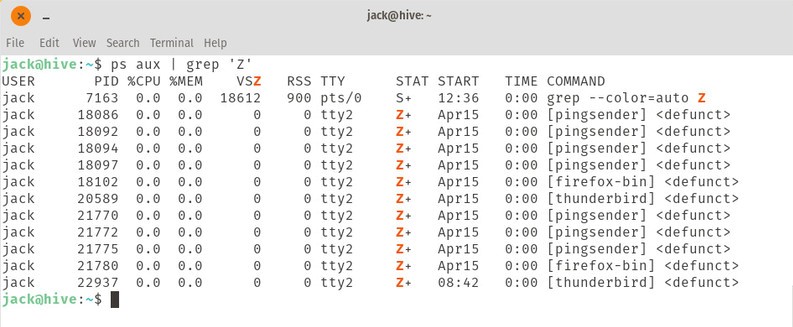
Why Zombie Process Created:

When a process is created in UNIX using fork() system call, the address space of the Parent process is replicated. If the parent process calls wait() system call, then the execution of parent is suspended until the child is terminated. This state of the child process is known as the Zombie state.

How to Spot a Zombie Process:

The first thing to do is to find the zombie process. Pipe the output of the ps command through grep to list out any process whose STAT is Z (for zombie). Open a terminal window and issue the following command:

**ps aux | grep 'Z’**

****

How to Kill Zombie Process:

Instead of decapitating, burning, or completely dismembering the zombie, we want to use the kill command to end those zombies. The output of the first command will include the PID of all zombied processes, so to kill one of those, you issue the command:

**kill PID**

Why do we need to prevent the creation of Zombie process?

There is one process table per system. The size of the process table is finite. If too many zombie processes are generated, then the process table will be full. That is, the system will not be able to generate any new process, then the system will come to a standstill. Hence, we need to prevent the creation of zombie processes.

// A C program to demonstrate working of

// fork()/wait() and Zombie processes

#include<stdio.h>

#include<unistd.h>

#include<sys/wait.h>

#include<sys/types.h>

int main()

{

int i;

int pid = fork();

if (pid==0)

{

for (i=0; i<20; i++)

printf("I am Child\n");

}

else

{

wait(NULL);

printf("I am Parent\n");

while(1);

}

}

Conclusion:

Finding zombie processes and killing their parents seems easy, but in some cases that might not be the best solution. The kill -9 command (aka kill –SIGKILL ) terminates a program immediately. It might work for some simple processes, however, most processes need to clean up temporary files and wrap up properly before being terminated. As a result of kill -9 there is a risk of unexpected problems, that are difficult to debug. That is why we highly recommend you only use kill -9 if you don’t see any other ways to solve this problem.