<https://banvien.udemy.com/course/linux-system-programming-f/learn/lecture/21169010#overview>

A white background with black text

Description automatically generated

A diagram of a program

Description automatically generated

Mỗi process có 1 process ID : PID

A screenshot of a computer

Description automatically generated

A computer code with text

Description automatically generated with medium confidence

PPID: parent process ID : 1 parent (creator) process chứa các process con. Dùng lệnh pid\_t **getppid**(void); để lấy danh sách các ppid. Khi chúng ta execute 1 process nhiều lần, thì shell dùng để execute process đó ko đổi -> PPID không đổi nhưng PID sẽ thay đổi (mỗi lần terminate sẽ hủy PID đó trả về cho system)

A screenshot of a computer

Description automatically generated

A diagram of a memory

Description automatically generated

A process memory layout

int add(int a, int b) {

int sum = a + b;

return sum;

}

int main() {

int x = 5;

int y = 10;

int result = add(x, y);

return 0;

}

Khi bắt đầu hàm main:

Stack: | ... |

| ... |

| x=5 |

| y=10 |

Khi gọi hàm add(x, y):

Stack: | ... |

| ... |

| x=5 |

| y=10 |

| a(copy of x) |

| b(copy of y) |

| return address |

Khi bắt đầu thực thi hàm add:

Stack: | ... |

| ... |

| x=5 |

| y=10 |

| a(copy of x) |

| b(copy of y) |

| return address |

| sum=a+b |

Khi kết thúc hàm add:

Stack: | ... |

| ... |

| x=5 |

| y=10 |

| result=return value of add |

Khi kết thúc hàm main:

Stack: | ... |

| ... |