МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Український державний університет науки і технологій**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №7**

**з дисципліни «Операційні системи»**

**на тему: «Вивчення механізму сигналів ОС UNIX»**

Виконав:

Студент гр. ПЗ2011

Кулик С.В.

Прийняв:

Андрющенко В.О.

Дніпро, 2023

**Тема:** Вивчення механізму сигналів ОС UNIX

**Мета:** Ознайомитися з механізмом сигналів UNIX (види сигналів, способи і засоби посилки сигналів). Отримати практичні навики програмування передачі і обробки сигналів. Отримати практичні навики використання сигналів.

**Короткі теоретичні відомості**

Сигнали є обмеженим засобом міжпроцессного обміну. Вони прекрасно підходять для повідомлень, але не можуть використовуватися для передачі інформації між процесами. Сигнали передаються без яких – то супутніх даних, тому вони зазвичай комбінуються з іншими способами обміну, наприклад, через файл.

Сигнали в системах сімейства UNIX можуть бути поділені на дві великі категорії: синхронні і асинхронні. Синхронні сигнали виникають при виконанні програмою певних операцій і таким чином, прив'язані до певної точки коди і певного потоку. Асинхронні сигнали виникають при подіях, зовнішніх по відношенню до процесу, і не мають такої прив'язки.

Довідковий матеріал:

Використовувані бібліоєтки:

- signal.h – заголовний файл для вказівки того, як програма обробляє сигнали під час її виконання.

- sys/type.h – декларація машинно-незалежних типів.

- sys/wait.h – для підключення системного виклику wait().

- sys/stat.h – містить структуру, що описує повернену функцією stat інформацію.

- unistd.h – містить символічні константи і структури, які ще не були описані в яких- небудь інших файлах, що включалися

- fcntl.h – описані запити і аргументи для системних викликів fcntl() і open().

iostream – потокове введення/вивід. Для використання функций з цієї бібліотеки необхідно вказати простір імен using namespace std;

**Постановка завдання**

Скласти програму, яка породжує декілька дочірніх процесів (схема fork+exec) і виконує завдання, вказане викладачем.

**Текст програми**

**Файл parent.cpp**

#include <signal.h>

#include <sys/types.h>

#include <sys/wait.h>

#include <unistd.h>

#include <fcntl.h>

#include <iostream>

using std::cout;

using std::endl;

const int STRING\_MAX\_SIZE = 256;

void ReadFile(int UNUSED) {

    char str[STRING\_MAX\_SIZE];

    int descriptor = open("./file.txt", O\_RDWR);

    if (descriptor != -1) {

        read(descriptor, str, sizeof(str));

        cout << "parent -> read: " << str << endl;

        close(descriptor);

    }

    else

        cout << "parent -> file not opened!" << endl;

}

int main() {

    cout << "parent -> process stared;" << endl;

    signal(SIGUSR1, ReadFile);

    int chpid = fork();

    if (chpid == 0){

        int error = execl("./child", "./child", NULL);

        cout << "parent -> child "

            << (error == -1 ? "was`t" : "")

            << " opened;";

    }

    else if (chpid > 0) {

        int statistic;

        waitpid(chpid, &statistic, 0);

    }

    else

        cout << "parent -> dublicating was't occured;";

    cout << "parent -> process ended;" << endl;

    return 0;

}

**Файл child.cpp**

#include <signal.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

#include <fcntl.h>

#include <iostream>

#include <string.h>

using std::cin;

using std::cout;

using std::endl;

int main() {

    cout << "cild -> process started;" << endl;

    const int STRING\_MAX\_SIZE = 256;

    char str[STRING\_MAX\_SIZE];

    int descriptor = creat("file.txt",0666);

    if (descriptor != -1) {

        cout << "child -> enter text: ";

        cin.getline(str, STRING\_MAX\_SIZE);

        write(descriptor, str, strlen(str) + 1);

        close(descriptor);

        kill(getppid(), SIGUSR1);

    }

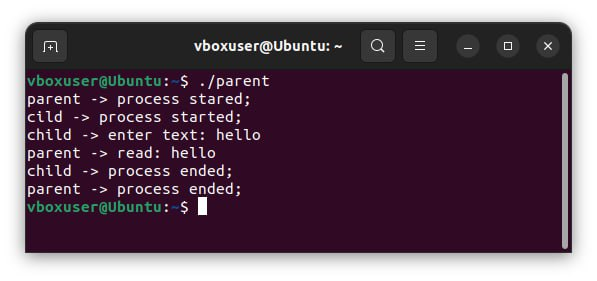
    else cout << "child -> file not created!" << endl;

    cout << "child -> process ended;" << endl;

    return 0;

}

**Результати програми та тестові прилади**



**Висновки**

У процесі виконання лабораторної роботи ознайомився з механізмом сигналів UNIX (види сигналів, способи і засоби посилки сигналів). Отримано практичні навики програмування передачі і обробки сигналів. Отримано практичні навики використання сигналів.