Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Операционные системы и системное программирование

ОТЧЕТ
по лабораторной работе №2
на тему
“Понятие процессов”

Выполнил:
Студ. гр. 350501 Чехович Д.С.

Проверил:
ст.пр. Поденок Л.П.

МИНСК 2025

СОДЕРЖАНИЕ

1 ЗАДАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ.....3
1.1 Цель работы.....3
1.2 Исходные данные к работе.....3
2 ВЫПОЛНЕНИЕ РАБОТЫ.....3
2.1 Описание алгоритма выпыполения работы.....3
2.2 Описание основных функций.....4
3 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ.....5
4 ВЫВОД.....6

1 ЗАДАНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

1.1 Цель работы

Разработать две программы – parent (родительский процесс) и child (дочерний процесс).

Родительский процесс, запуская дочерний, создает для него сокращенную среду (окружение). Для этого пользователем создается файл env, содержащий небольшой набор имен переменных окружения, передаваемых при вызове execve().

1.2 Исходные данные к работе

Минимальный набор переменных в файле env должен включать SHELL, HOME, HOSTNAME, LOGNAME, LANG, TERM, USER, LC_COLLATE, PATH.

Перед запуском программы parent в ее окружении пользователем создается переменная CHILD_PATH с именем каталога, где находится программа child.

Родительский процесс (программа parent) после запуска получает переменные своего окружения и их значения, установленные оболочкой, сортирует в LC_COLLATE=C и выводит в stdout. Читает файл env и формирует среду для дочернего процесса в том виде, в котором она указывается в системном вызове execve(), используя значения для переменных из собственной среды. После этого входит в цикл обработки нажатий клавиатуры.

2 ВЫПОЛНЕНИЕ РАБОТЫ

2.1 Описание алгоритма выполнения работы

Алгоритм выполнения работы начинается с подготовки окружения. Создается текстовый файл env.txt, содержащий список переменных окружения, и устанавливаются переменные окружения CHILD_PATH (путь к исполняемому файлу дочерней программы) и ENV_PATH (путь к файлу env.txt). После этого осуществляется компиляция исходных кодов программ parent.c и child.c с использованием предоставленного Makefile, что приводит к созданию исполняемых файлов parent и child.

Далее запускается родительская программа, которая предоставляет текстовый интерфейс для взаимодействия с пользователем. Пользователь может инициировать создание дочерних процессов, которые выполняют программу child. Родительская программа создает дочерние процессы с использованием системного вызова fork, а затем передает управление дочерней программе через системный вызов execve.

Дочерняя программа открывает файл env.txt, читает из него имена переменных окружения и выводит их значения, используя функцию getenv.

После завершения работы дочернего процесса родительская программа продолжает выполнение, ожидая следующей команды от пользователя. Завершение работы программы происходит при вводе команды q, после чего все временные файлы удаляются с помощью команды make clean.

2.2 Описание основных функций

Функция fork()

Функция fork отвечает за создание нового процесса с помощью системного вызова fork. В родительском процессе она выводит информацию о текущем процессе и ожидает завершения дочернего процесса с помощью wait. В дочернем процессе она запускает программу child с помощью execve, передавая ей аргументы и переменные окружения. Эта функция обеспечивает взаимодействие между родительским и дочерним процессами, а также обработку ошибок при создании процесса.

Функция increment()

Функция increment генерирует уникальное имя для дочернего процесса в формате child_XX, где XX — номер процесса, увеличиваемый при каждом вызове. Она выделяет память для строки имени и заполняет её символами, соответствующими текущему значению счетчика. Если счетчик достигает 100, он сбрасывается на 0, чтобы избежать переполнения. Эта функция используется для идентификации дочерних процессов.

Функция print_env()

Функция print_env выводит все переменные окружения, переданные в массиве env. Она проходит по массиву и отображает каждую переменную в формате "ключ=значение". Эта функция полезна для диагностики и проверки доступных переменных окружения, которые могут быть использованы в программе. Она не изменяет переменные окружения, а только отображает их.

Функция main()

Функция main в программе child.c выполняет основную задачу — чтение файла env.txt и вывод значений переменных окружения, указанных в этом файле. Она открывает файл, считывает строки, сравнивает их для предотвращения повторений и использует функцию getenv для получения значений переменных окружения. После завершения работы с файлом она освобождает выделенную память и завершает выполнение программы. Эта функция взаимодействует с родительским процессом через аргументы командной строки.

3 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ

```
Hello! This is the parent process.
new process: '+','*' or '&'
'q' - exit
+
I am parent 19084
Child is 19087
In child program /home/dimin/term_4/OSSP/lab02/child child_00
/home/dimin/term_4/OSSP/lab02/env.txt
```

Рисунок 1 – пример работы символа “+”

```
Hello! This is the parent process.
new process: '+','*' or '&'
'q' - exit
&
I am parent 19087
Child is 19089
In child program /home/dimin/term_4/OSSP/lab02/child child_01
/home/dimin/term_4/OSSP/lab02/env.txt
```

Рисунок 2 – пример работы символа “&”

```
Hello! This is the parent process.
new process: '+','*' or '&'
'q' - exit
*
I am parent 19084
Child is 19090
In child program /home/dimin/term_4/OSSP/lab02/child child_01
/home/dimin/term_4/OSSP/lab02/env.txt
```

Рисунок 3 – пример работы символа “*”, после двух выходов (q)

4 ВЫВОД

В ходе изучения системных вызовов было рассмотрено создание процессов с помощью fork(), замена текущего процесса на новый с использованием execve(), а также получение идентификаторов процессов через getpid() и getppid(). Также была изучена функция getenv(), позволяющая извлекать значения переменных окружения, и рассмотрена связь системного вызова execve() с функцией main() языка C, что позволяет запускать новые программы с заданными аргументами и окружением.