**Министерство цифрового развития, связи и массовых коммуникаций**

**Российской Федерации**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Кафедра Системного программирования

ЛАБОРАТОРНАЯ РАБОТА №4

по дисциплине

ОПЕРАЦИОННЫЕ СИСТЕМЫ

на тему:

«РЕАЛИЗАЦИЯ ОБМЕНА ДАННЫМИ МЕЖДУ ПРОЦЕССАМИ»

Выполнил:

студент \_\_\_\_Шамсутдинов Р.Ф.\_\_\_\_\_

(Ф.И.О.)

группа\_\_\_\_\_\_\_\_БВТ2201\_\_\_\_\_\_\_\_\_

Проверил:

\_\_\_\_\_\_\_Королькова Т. В.\_\_\_\_\_\_\_\_\_

(Ф.И.О., должность преподавателя)

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_30.04.2025\_\_\_\_\_\_\_\_\_\_

Москва 2025

**Введение**

**Цель работы:**

• изучение системных средств обмена данными между процессами в

ОС GNU/LINUX

• получение практических навыков использования механизмов

межпроцессного взаимодействия

**Задание:**

1. Изучите теоретическую часть лабораторной работы.

2. Написать программу, выполняющую с помощью ВСЕХ рассмотренных

выше системных средств обмена данными между процессами (разделяемая

память, сокеты, каналы) одну из задач (в соответствии с № по журналу),

приведенных в Таблице 1. При выполнении задачи необходимо создать как

минимум 2 ведомых процесса, выполняющих переданные ведущим процессом подзадачи и возвращающие результаты ведущему процессу. Финальную обработку результатов при необходимости осуществлять в ведущем процессе.

Задача из Таблицы 1.  
Реализовать обмен текстовыми сообщениями между несколькими

процессами. Обеспечить возможность отправки сообщения сразу

нескольким адресатам. Реализовать подтверждение приема сообщения адресатом или, в случае потери сообщения, повторную его передачу.

**Ход работы**

1. Реализация на каналах.

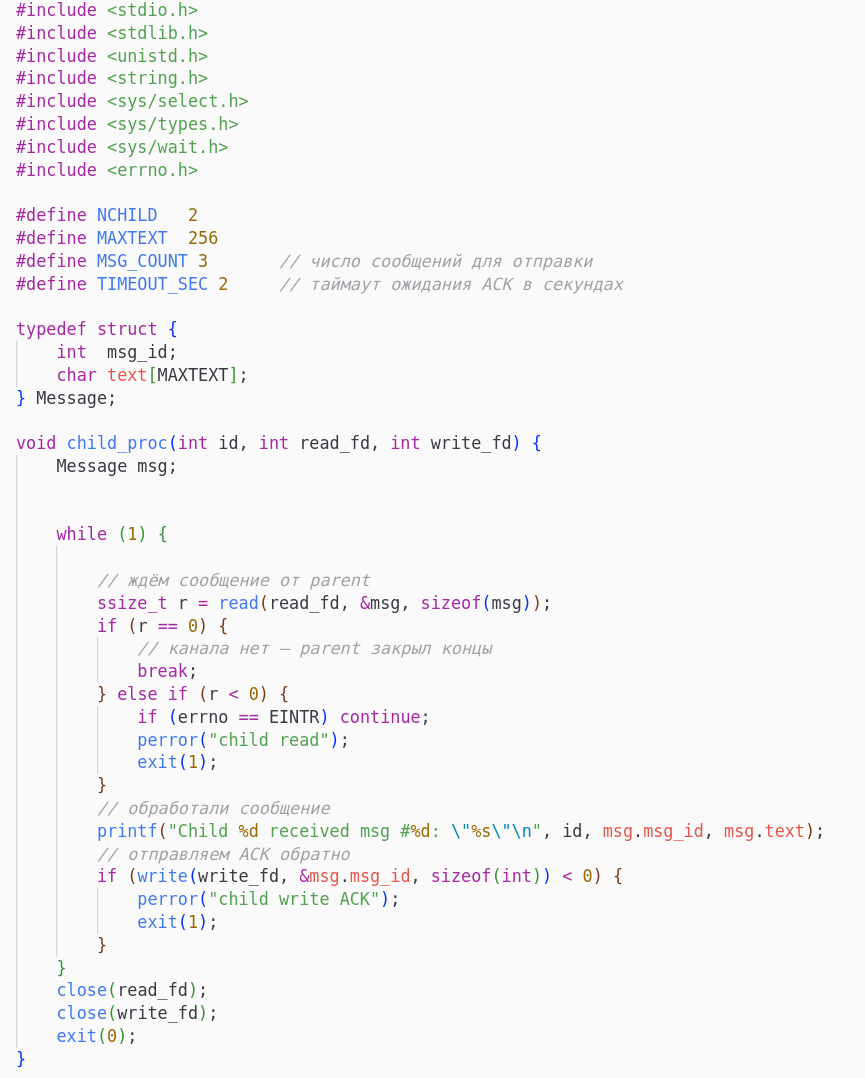


Рисунок 1. Библиотеки, константы, функция дочернего процесса



Рисунок 2. Функция мейн, создание каналов и дочерних процессов, закрытие ненужных концов, массив сообщений

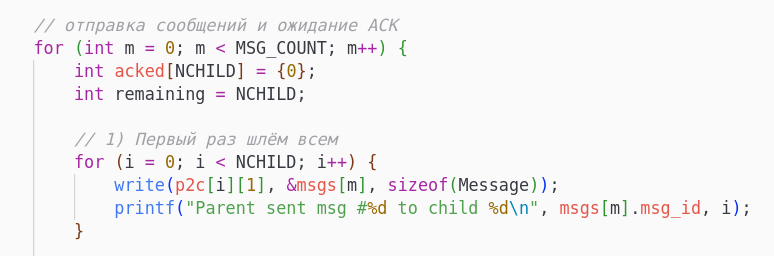


Рисунок 3. Отправка сообщений всем дочерним процессам

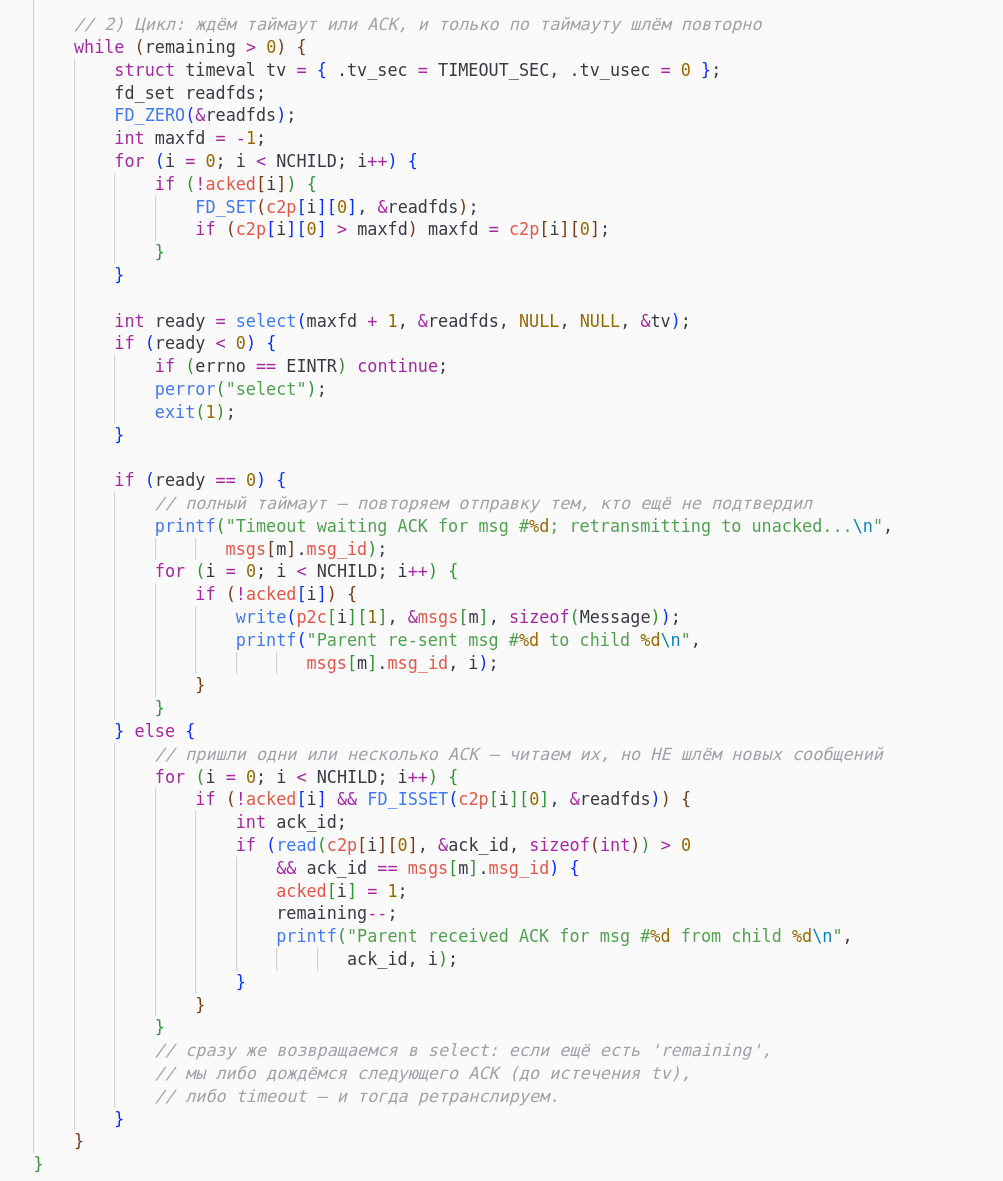


Рисунок 4. Ожидание подтверждений, повторная отправка



Рисунок 5. Закрытие каналов, дочерних процессов.

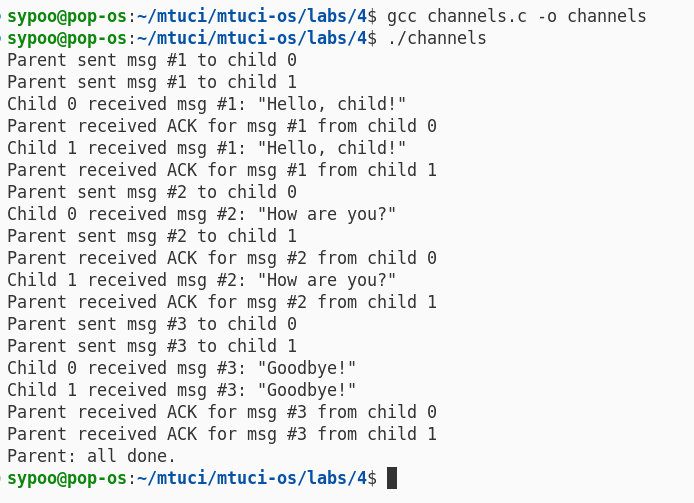


Рисунок 6. Сборка и вывод программы

1. Реализация с общей памятью

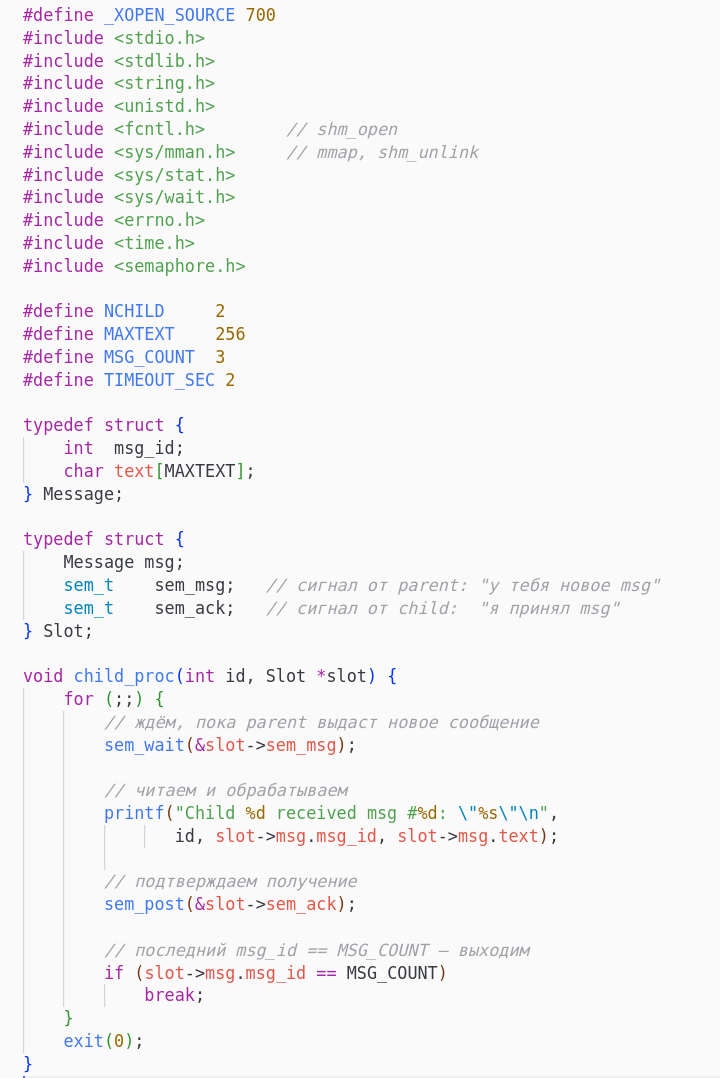


Рисунок 7. Библиотеки, константы, функция дочернего процесса

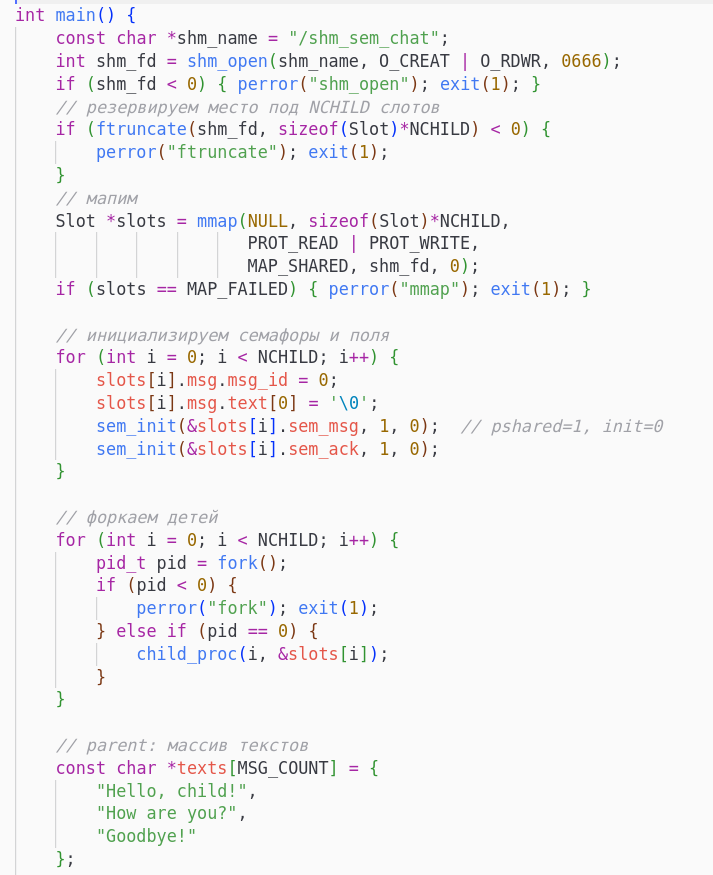


Рисунок 8. Создание общей памяти, семафоры, создание дочерних процессов



Рисунок 9. Отправка сообщений и ожидание подтверждений, повторная отправка.

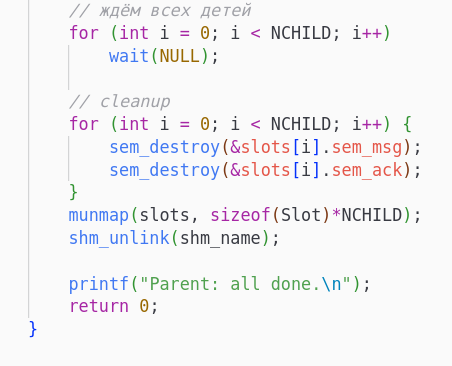


Рисунок 10. Ожидание завершения дочерних процессов, очистка общей памяти.

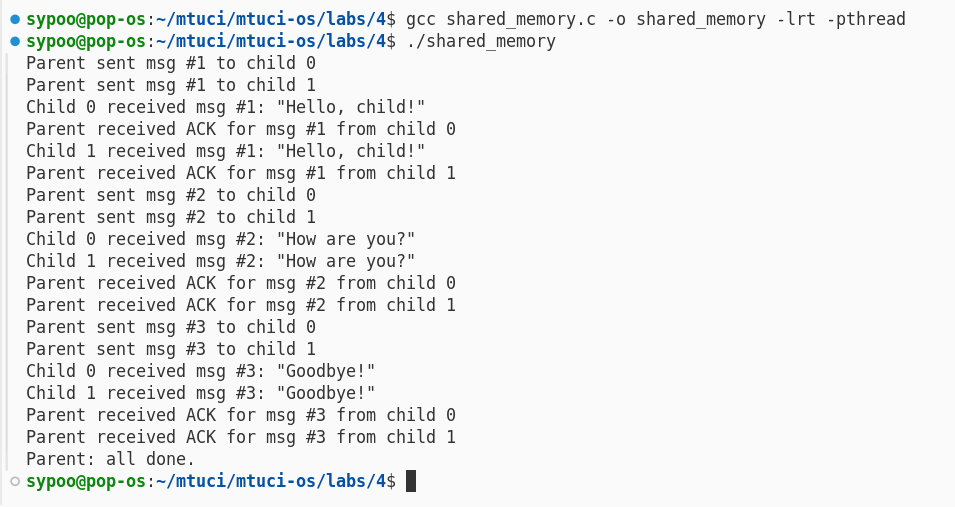


Рисунок 11. Сборка и вывод программы

1. Реализация на сокетах.



Рисунок 12. Библиотеки, константы



Рисунок 13. Функция дочернего процесса

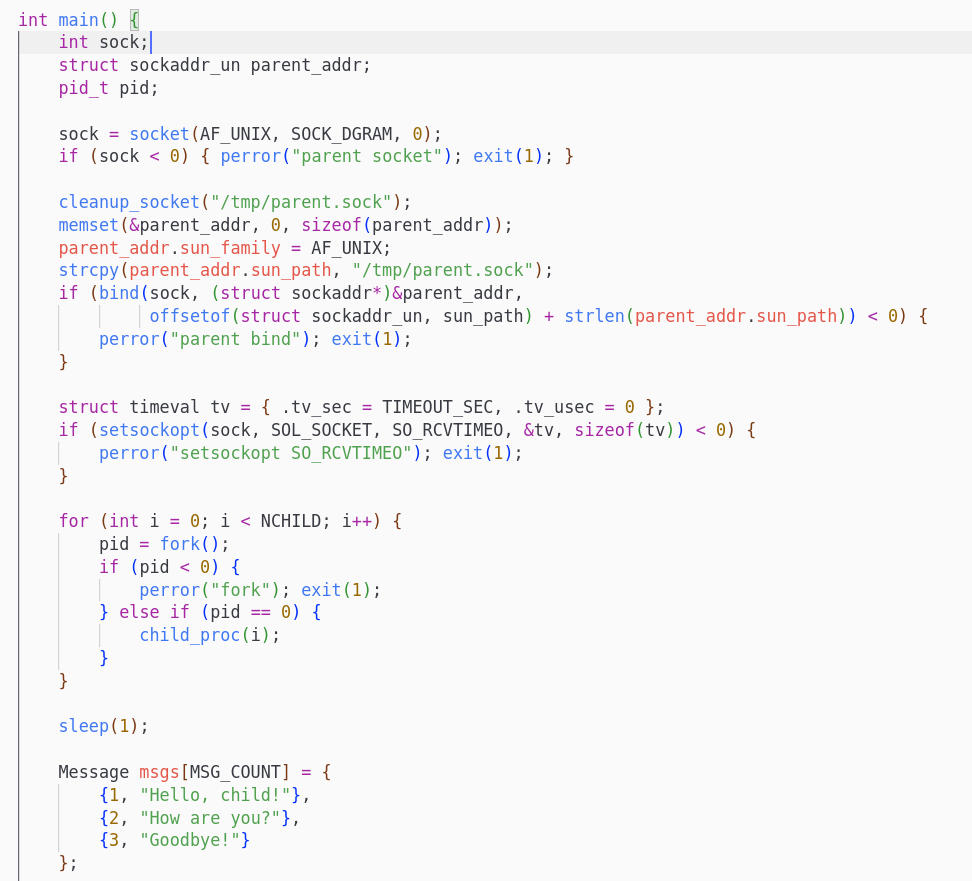


Рисунок 14. Создание сокетов и дочерних процессов.

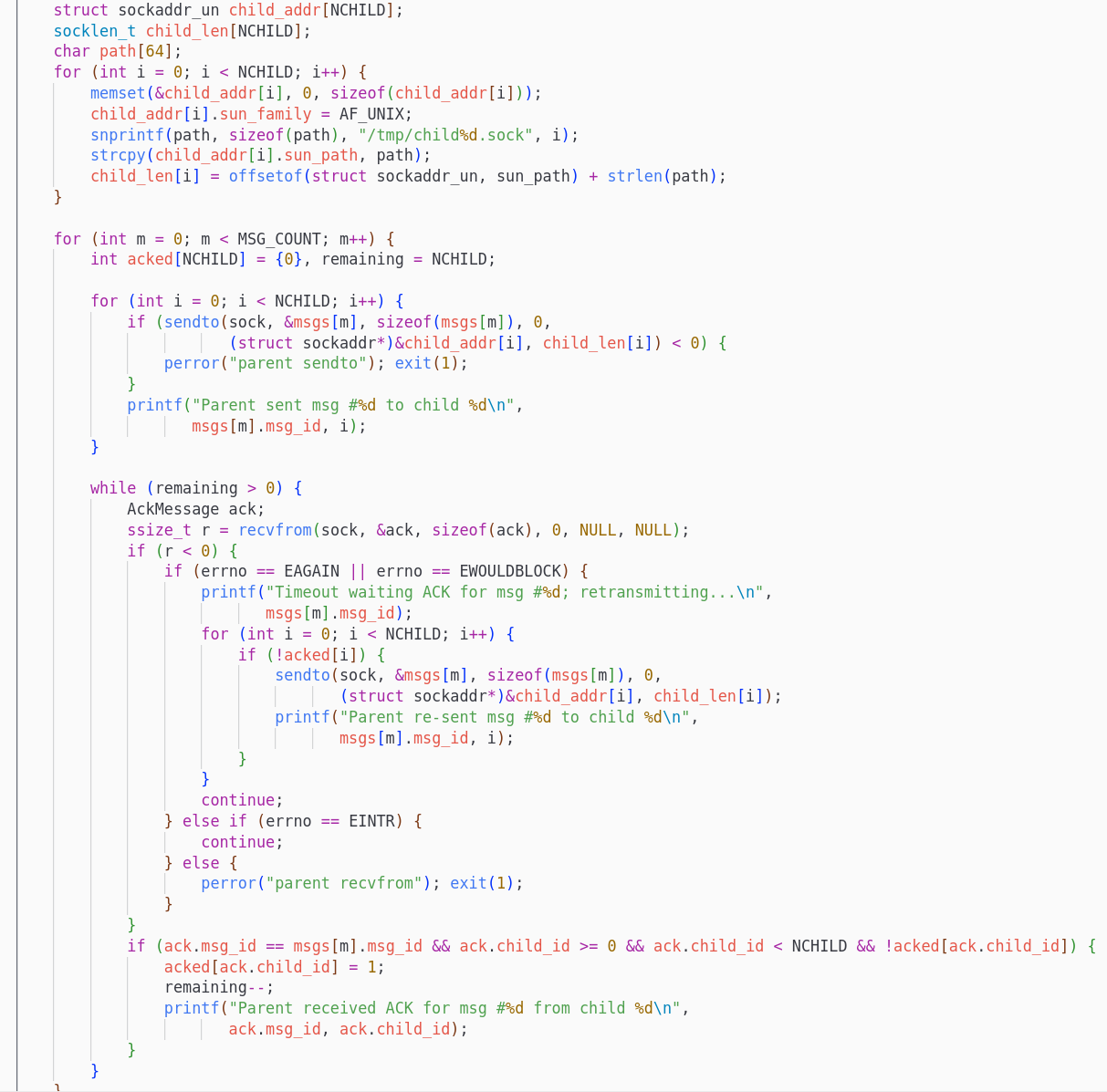


Рисунок 15. Отправка сообщений.



Рисунок 16. Завершение программы, очистка.

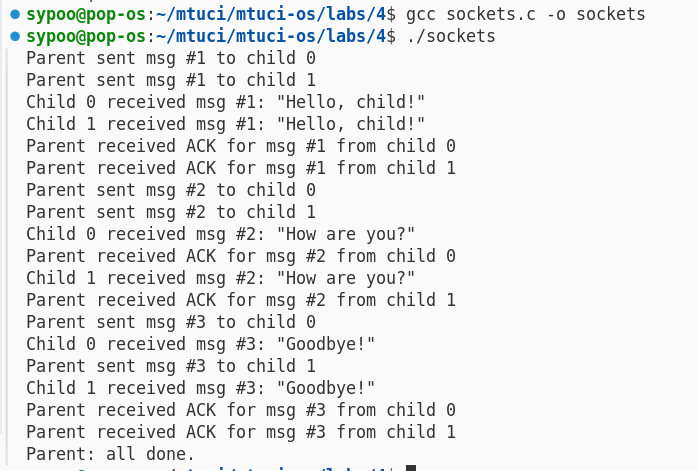


Рисунок 17. Сборка и вывод программы

**Заключение**

**Вывод:** В ходе лабораторной работы были успешно реализованы три механизма межпроцессного взаимодействия (IPC) — сокеты, разделяемая память с семафорами и каналы с select(). Каждый метод показал свою эффективность: сокеты обеспечили гибкость, разделяемая память — высокую скорость, а каналы — простоту реализации. Наибольшую надежность продемонстрировал подход с сокетами благодаря встроенной поддержке таймаутов и переотправки, что делает его оптимальным для задач, требующих гарантированной доставки сообщений.