

4 балла

1. Разработать программу, которая «переворачивает» заданную позициями N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_4_points.c

4_points_test.zip

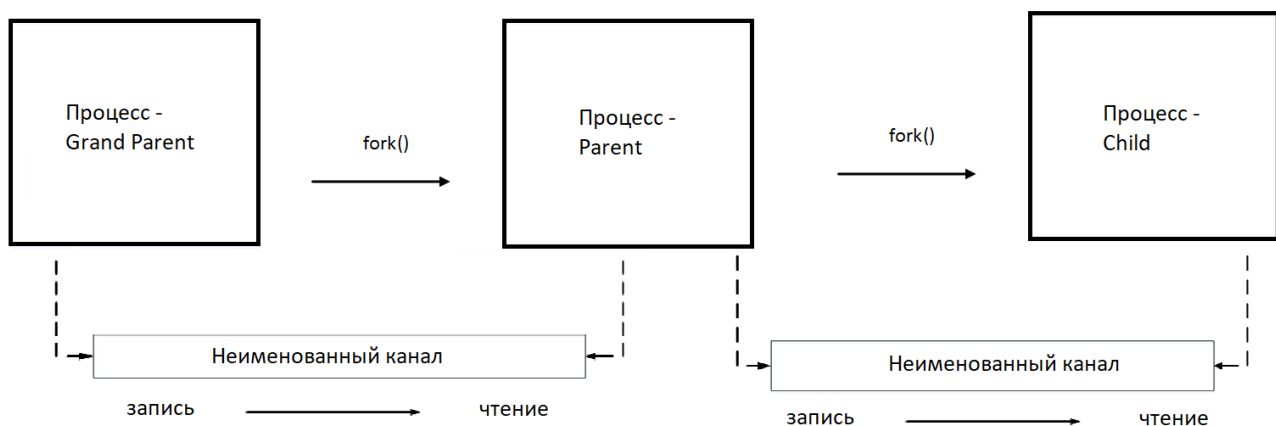
Ввод и вывод данных:

```
~/INT_1$ gcc program_4_poins.c
~/INT_1$ ./a.out text_test2.txt output.txt 0 100
```

Параметрами для исполняемого файла являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл
- 3) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 4) N_3 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:



5 баллов

1. Разработать программу, которая «переворачивает» заданную позициями N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_5_points.c

5_points_test.zip

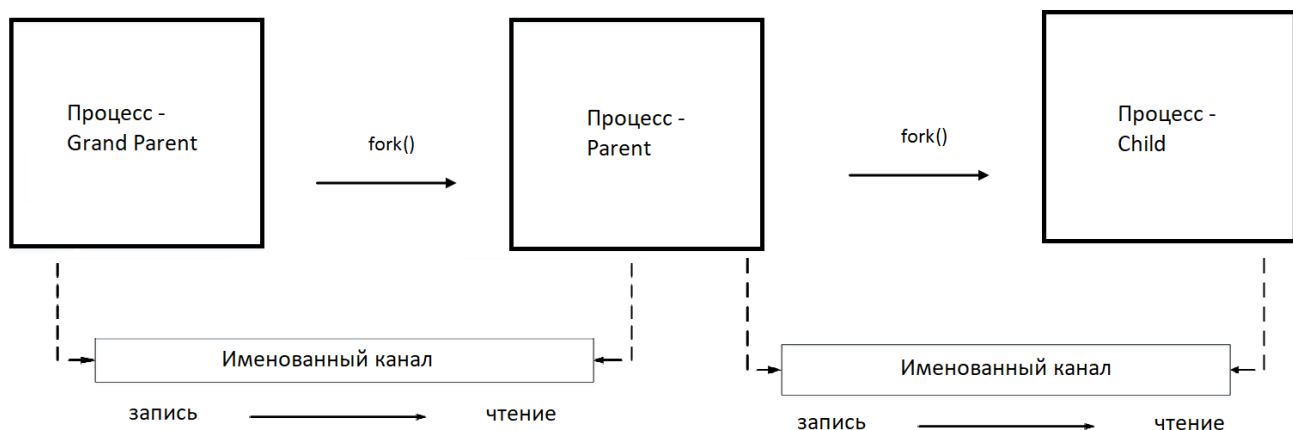
Ввод и вывод данных:

```
~/INT_1$ gcc program_5_poins.c  
~/INT_1$ ./a.out text test2.txt output.txt 0 100
```

Параметрами для исполняемого файла являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл
- 3) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 4) N_3 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:



6 баллов

1. Разработать программу, которая «переворачивает» заданную позициями N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_5_points.c

5_points_test.zip

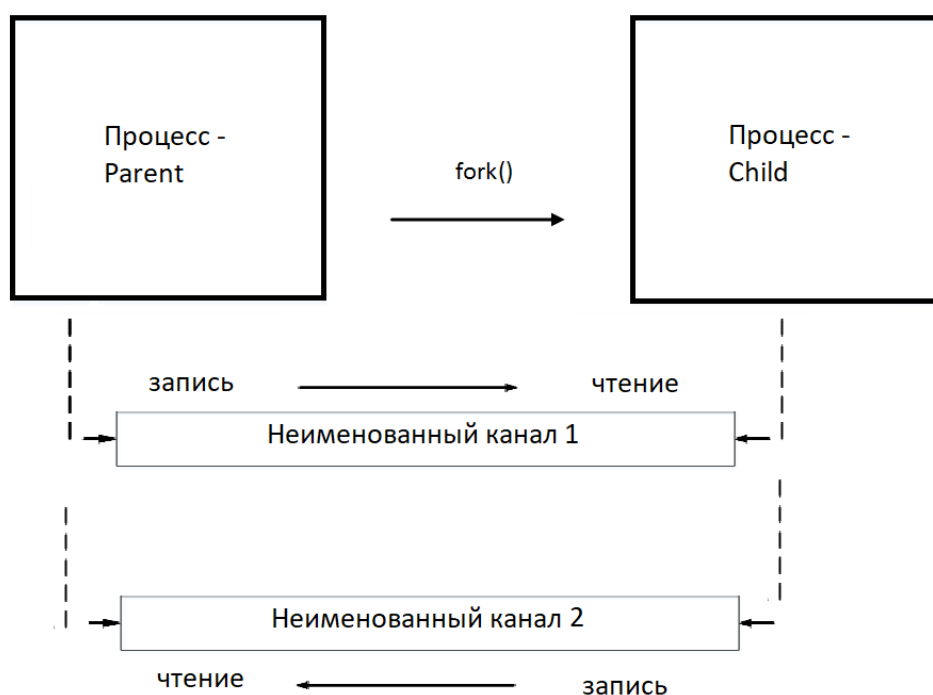
Ввод и вывод данных:

```
~/INT_1$ gcc program_6_poins.c  
~/INT_1$ ./a.out text_test2.txt output.txt 0 100
```

Параметрами для исполняемого файла являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл
- 3) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 4) N_3 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:



6 баллов

1. Разработать программу, которая «переворачивает» заданную позициями N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_6_points.c

6_points_test.zip

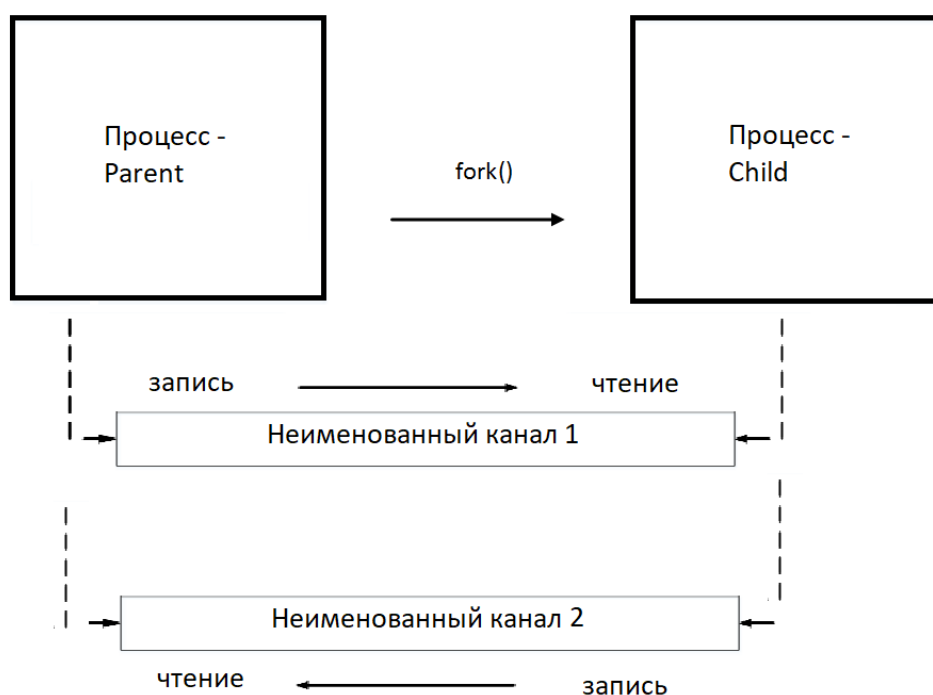
Ввод и вывод данных:

```
~/INT_1$ gcc program_6_poins.c
~/INT_1$ ./a.out text_test2.txt output.txt 0 100
```

Параметрами для исполняемого файла являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл
- 3) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 4) N_3 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:



7 баллов

1. Разработать программу, которая «переворачивает» заданную позициями N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_7_points.c

7_points_test.zip

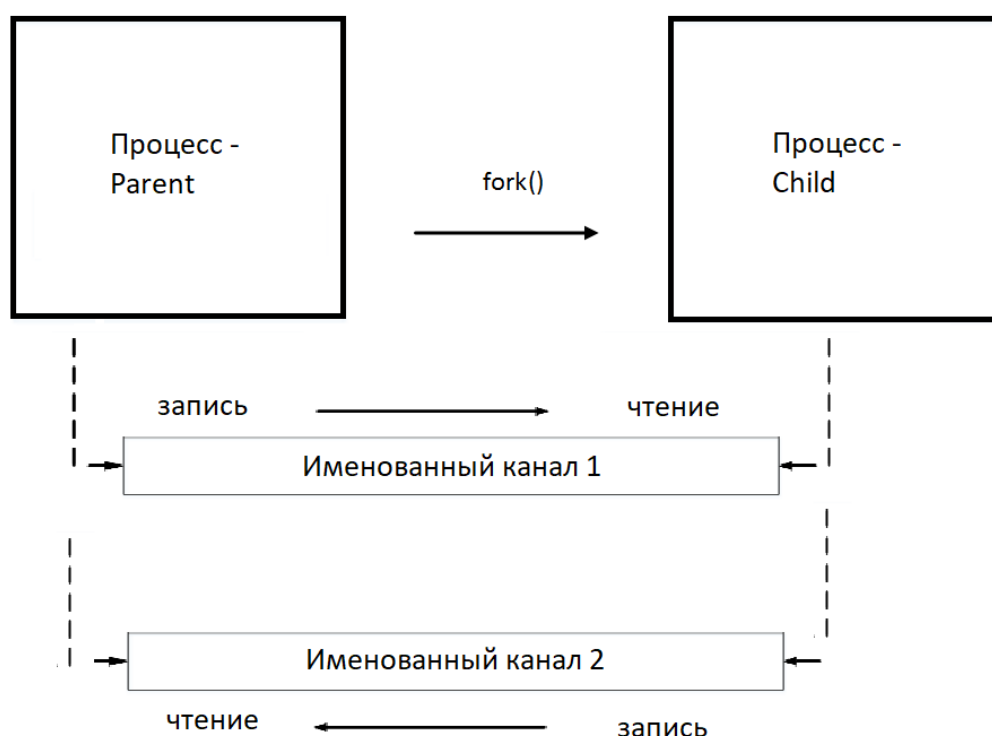
Ввод и вывод данных:

```
~/INT_1$ gcc program_7_poins.c  
~/INT_1$ ./a.out text test2.txt output.txt 0 100
```

Параметрами для исполняемого файла являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл
- 3) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 4) N_3 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:



8 баллов

1. Разработать программу, которая «переворачивает» заданную позициюми N_1 – N_2 часть ASCII-строки символов (N_1 , N_2 вводятся как параметры).

Файлы и каталоги:

program_8_poins_process_1.c

program_8_poins_process_2.c

8_points_test.zip

Ввод и вывод данных:

```
~/INT_1$ gcc program_8_poins_process_1.c -o p1.out  
~/INT_1$ gcc program_8_poins_process_2.c -o p2.out  
~/INT_1$ ./p1.out text_test2.txt output.txt
```

```
~/INT_1$ ./p2.out 0 100
```

Параметрами для исполняемого файла p1 являются:

- 1) Входной файл с текстом из ASCII символов (коды в диапазоне от 0-127)
- 2) Выходной файл

Параметрами для исполняемого файла p2 являются:

- 1) N_1 – начальная позиция переворота строки (включая и символ на позиции N_1)
- 2) N_2 – конечная позиция переворота строки (включая и символ на позиции N_2)

Общая схема решаемой задачи:

