

ИДЗ №1

Воробьев Артём Александрович, БПИ214

Вариант №7

Отчет на 4

Условие задания: разработать программу, заменяющую все гласные буквы в заданной ASCII-строке их **ASCII кодами в шестнадцатичной системе счисления**. Код каждого символа задавать в формате «0xDD», где **D** — шестнадцатичная цифра от 0 до F.

Первое разделение:

```
int main(__attribute__((unused)) int argc, char *argv[]) {
    umask(0);

    char *t = argv[1];
    char *f = argv[2];
    int fd[2];
    char buffer[size];
    pipe(fd);

    int result = fork(); // разделяем процесс
    if (result < 0) {
        printf("Не получилось разделить\n");
        exit(-1);
    } else if (result > 0) { // Родитель
        if (!firstProcess(fd, t, buffer)) {
            exit(-1);
        }
        printf("Выходим из первого\n");
    } else { // Ребенок
        if (!secondProcess(fd, f, buffer)) {
            exit(-1);
        }
        printf("Выходим из второго\n");
    }
    return 0;
}
```

Второе разделение:

```
int secondProcess(int fd[], char *f, char buffer[]) {
    // Второй процесс
    if (close(fd[1]) < 0) {
        printf("Второй процесс не может записать\n");
        return 0;
    }

    start *createNewString = newString("");

    size_t a;
    while ((a = read(fd[0], buffer, size)) > 0) {
        for (int i = 0; i < a; ++i) {
            charEnd(createNewString, buffer[i]);
        }
    }

    start *string = change(createNewString);

    free(createNewString->string);
    free(createNewString);

    if (close(fd[0]) < 0) {
        printf("Второй процесс не может считать\n");
        return 0;
    }

    pipe(fd);

    int res = fork();
    if (res < 0) {
        printf("Не получилось разделить\n");
        return 0;
    } else if (res > 0) { // Подпроцесс второго
        if (!secondProcessChild(fd, string)) {
            return 0;
        }
    }
}
```

- первый процесс читает текстовые данные из заданного файла и через неименованный канал передает их второму процессу;
- второй процесс осуществляет обработку данных в соответствии с заданием и передает результат обработки через неименованный канал третьему процессу;
- третий процесс осуществляет вывод данных в заданный файл.

Ввод и вывод данных при работе с файлами осуществляется через системные вызовы `read` и `write`.

Для задания имен входного и выходного файлов используются аргументы командной строки.

