

**POLITECHNIKA  
WROCŁAWSKA**



Wydział Informatyki i  
Telekomunikacji

Wydział: Informatyki i Telekomunikacji

Kierunek: Cyberbezpieczeństwo

Rok Akademicki: 2024/2025

Rok studiów, semestr: 3

Grupa: 6

Termin: środa, 15:15

### **Programowanie Systemowe**

Prowadzący:

mgr inż. Przemysław Świercz

Autor:

1. Jakub Regulski 283866

Data wykonania ćwiczenia:

19.11.2025

Data oddania sprawozdania:

24.11.2025

## Kod:

```
#include<stdio.h>
#include<string.h>
#include <sys/wait.h>
#include <unistd.h>

void server(int fd[2], int fd2[2]){
    char napis[256];
    close(fd[0]);
    close(fd2[1]);
    while (1){
        printf("Enter a string");
        scanf("%s", napis);
        write(fd[1], napis, strlen(napis) + 1);
        if(strcmp(napis, "EXIT") == 0) break;
        read(fd2[0], napis, 256);
        printf("Modified string: %s\n", napis);
    }
    wait(NULL);
}

void worker(int fd[2], int fd2[2]){
    char napis[256];
    close(fd[1]);
    close(fd2[0]);

    while (1){
        read(fd[0], napis, 256);
        if(strcmp(napis, "EXIT") == 0) break;
        napis[0] = 'X';
        write(fd2[1], napis, strlen(napis)+1);
    }
}

int main(){
    int fd[2];
    int fd2[2];
    pid_t pid;
    if(pipe(fd) == -1){
        printf("Error");
        return 1;
    }
    if(pipe(fd2) == -1){
        printf("Error");
        return 1;
    }
    sf_VM
    pid = fork();

    if(pid>0){
        server(fd,fd2);
    }
    if(pid==0){
        worker(fd, fd2);
    }
    return 0;
}
```

Wyjaśnienie:

Plik main.c zawiera 3 funkcje:

1. **Funkcja server** odczytuje wczytany tekst przez użytkownika, wysyła go przy pomocy pipe, później odczytuje przerobiony tekst wysłany przez drugiego pipe ` a i wyświetla go. Jeżeli wprowadzony przez użytkownika tekst to „EXIT” to funkcja kończy się.
2. **Funkcja worker** odczytuje tekst wysłany pipe ` m, Zamienia jego pierwszą literę na ‘X’ i przesyła przerobiony tekst drugim pipe ` m. Jeżeli odczytany z pipe ` a tekst to „EXIT” to funkcja kończy się.
3. **Funkcja main** tworzy proces potomny. Dla procesu rodzica wywołuje funkcję server a dla dziecka funkcję worker.