

# Отчет по лабораторной работе № 10 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Фомин Иван Дмитриевич, № 22

Контакты email: [grenka388@gmail.com](mailto:grenka388@gmail.com),

Telegram: @Haliaven

Работа выполнена: «11» ноября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « » \_\_\_\_\_ 20\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Отладчик системы программирования ОС UNIX.
2. **Цель работы:** Научиться пользоваться отладчиком
3. **Задание:** Внесение и исправление ошибок различных типов
4. **Оборудование** (студента):

Процессор AMD Ryzen 7 4800HS with Radeon Graphics с ОП 8 Гб, SSD 512 Гб. Монитор 1920x1080

5. **Программное обеспечение** (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия 22.04 *jammy*

интерпретатор команд: *bash* версия 5.1.16(1)-release

Система программирования CLion версия 2021.1.3, редактор текстов *nano* версия 6.2

Утилиты операционной системы WinRar, Microsoft Word

Прикладные системы и программы Ubuntu wsl, CLion, Google Chrome

Местонахождение и имена файлов программ и данных на домашнем компьютере ~/home

## 6. Идея, метод, алгоритм

Генерируем различные виды ошибок, их описание есть в пункте 8

## 7. Сценарий выполнения работы

Входные данные	Выходные данные	Описание тестируемого случая
ничего	Разные ошибки	Получаем различные типы ошибок

## 8. Распечатка протокола

### ОШИБКА RE #1 – использование необъявленной переменной

```
int main() {
    int i;

    for (int j = 0; j < i * 10; ++j) {
        printf("%d", i);
    }

    return 0;
}
```

### РЕШЕНИЕ – инициализировать i=1;

```
int main() {
    int i;

    i = 1;
    for (int j = 0; j < i * 10; ++j) {
        printf("%d", i);
    }

    return 0;
}
```

### ОШИБКА RE #2 – деление на 0

```
#include <stdio.h>

float delZero(float i, float res) {
    if (i < 0) return res;
    return delZero(i - 1, res + (i / i - 1));
}

int main() {
    printf("%f", delZero(100, 1));

    return 0;
}
```

### РЕШЕНИЕ - заменить if (i < 0) на if (i <= 0)

```
float delZero(float i, float res) {
    if (i <= 0) return res;
    return delZero(i - 1, res + (i / i - 1));
}

int main() {
    printf("%f", delZero(100, 1));

    return 0;
}
```

### ОШИБКА SE #1 – деление массива на число

```
int main() {
    int arr[2] = {0};
    int num = 10;
    arr[0] = 100;

    printf("%d", arr/num);

    return 0;
}
```

### РЕШЕНИЕ – обратиться к числу по индексу

```
int main() {
    int arr[2] = {0};
    int num = 10;
    arr[0] = 100;

    printf("%d", arr[0]/num);

    return 0;
}
```

### ОШИБКА СЕ #2 – обращение к несуществующему полю

```
int main() {  
    int var = 120;  
    int err = var.field;  
  
    printf("%d", err);  
  
    return 0;  
}
```

### РЕШЕНИЕ – создать объект с таким полем

```
int main() {  
    struct obj {  
        int field;  
    } var;  
    var.field = 5;  
  
    int err = var.field;  
  
    printf("%d", err);  
  
    return 0;  
}
```

### ОШИБКА UB #1 – вызов функции, которая ничего не возвращает, хотя должна

```
int foo() {  
}  
  
int main() {  
    return foo();  
}
```

### РЕШЕНИЕ – сделать так, чтобы функция что-нибудь возвращала

```
int foo() {  
    return 5;  
}  
  
int main() {  
    return foo();  
}
```

### ОШИБКА UB #2 – обращение к несуществующему элементу

```
int main() {  
    int a[231];  
  
    printf("%d", a[-31]);  
  
    return 0;  
}
```

### РЕШЕНИЕ – поменять -31 на 31;

```
int main() {  
    int a[231];  
    a[31] = 80085;  
  
    printf("%d", a[31]);  
  
    return 0;  
}
```

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	дом	11.11.2022	11:30	Не знал как получить UB	Загуглил	Хоть и загуглил, понятнее не стало

### 10. Замечания автора

Неактуально, потому что информативность ошибок в Си нулевая. Просто код ошибки мне ничего не говорит. Фу какой язык.

### 11. Выводы

Работа вообще не понравилась, потому что самому придумывать ошибки – боль. Лучше наткаться на эти же ошибки в процессе работы, чтобы получить, так сказать, боевой опыт. В интернете нет вообще никакой информации о том, как вызвать эти ошибки, а сам Си разрешает делать почти что угодно, не бросая ошибок. Поэтому было неприятно заниматься выдумыванием проблем.

*Закончить надо на хорошей ноте, поэтому анекдот:*

Дагестанские врачи боролись за жизнь пациента

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента \_\_\_\_\_