

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

Студент группы М80-109Б-22 Потапов Егор Дмитриевич, №14

Контакты @mathculture

Работа выполнена: «10» 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:** Введение и исправление ошибок различных типов

4. **Оборудование** (студента):

Процессор *Intel Core i5-8265U @ 8x 3.9GHz* с ОП 7851 Мб, НМД 1024 Гб. Монитор 1920x1080

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

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *18.10 cosmic*
интерпретатор команд: *bash* версия *4.4.19*.

Система программирования -- версия --, редактор текстов *emacs* версия 25.2.2

Утилиты операционной системы --

Прикладные системы и программы --

Местонахождение и имена файлов программ и данных на домашнем компьютере --

6. **Идея, метод, алгоритм.**

Придумываем ошибки и сами их исправляем

7. **Сценарий выполнения работы.**

Входные данные	Выходные данные	Описание тестируемого случая
ничего	ничего	ничего

8. Распечатка протокола.

SE ошибка. Функции подаётся на вход 1 аргумент, а нужно – 2. Исправление – подать функции на вход 2 аргумента.

<pre>1 #include<stdio.h> 2 int sum(int a, int b){ 3 return a+b; 4 } 5 6 int main(){ 7 printf("%d", sum(5)); 8 return 0; 9 } 10</pre>	<pre>1 #include<stdio.h> 2 int sum(int a, int b){ 3 return a+b; 4 } 5 6 int main(){ 7 printf("%d", sum(5, 3)); 8 return 0; 9 } 10</pre>
--	---

SE ошибка. Не объявлен тип переменной c. Решение – int c

<pre>1 #include<stdio.h> 2 #include<math.h> 3 4 int main(){ 5 int a=2; 6 c=powl(4, a); 7 printf("%d", c); 8 return 0; 9 } 10</pre>	<pre>1 #include<stdio.h> 2 #include<math.h> 3 4 int main(){ 5 int a=2; 6 int c=powl(4, a); 7 printf("%d", c); 8 return 0; 9 } 10</pre>
--	--

RE ошибка. b может быть нулём, упало во время выполнения. Решение - проверка, что b!=0.

<pre>1 #include<stdio.h> 2 int del(a, b){ 3 return a/b; 4 } 5 6 int main(){ 7 printf("%d", del(16, 0)); 8 return 0; 9 } 10</pre>	<pre>1 #include<stdio.h> 2 int del(int a, int b){ 3 if(b!=0) 4 return a/b; 5 else 6 return 0; 7 } 8 9 int main(){ 10 printf("%d", del(16, 0)); 11 return 0; 12 } 13</pre>
--	---

RE ошибка. Нет спецификатора %d. Решение – добавить его.

<pre>1 #include<stdio.h> 2 int main(){ 3 int j=10; 4 for(int i=1;i<j;i++){ 5 printf(i); 6 } 7 return 0; 8 } 9</pre>	<pre>1 #include<stdio.h> 2 int main(){ 3 int j=10; 4 for(int i=1;i<j;i++){ 5 printf("%d\n", i); 6 } 7 return 0; 8 } 9</pre>
--	--

UB ошибка. в Си нет двойных неравенств. (делал в другом IDE по техническим причинам). Решение – разбить на несколько неравенств

```
20 }
21 int main(){
22     int i = 10;
23     int j = -8;
24     int l = 10;
25     if (5 <= i <= 15 && -15 <= j <= -5)
26         printf("Точка находится в границах на 0 шаге: i=%d j=%d l=%d", i, j, l);
27     return 0;
}
```

Консоль отладки Microsoft Visual Studio

C:\Users\simple\Desktop\УЧЕБА\ЛАБЫ ИО\лаба8\Project2\x64\Debug\Project2.exe (процесс 9296) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно: _

```
if ((i>=5 && i<=15) && (j>=-15 && j<=-5))
    printf("The point is in the right boundaries: i=%d j=%d l=%d", i, j, l);
```

UB ошибка. Складывал число и список. Решение - обращаться к элементу списка по индексу и складывать.

<pre>1 #include<stdio.h> 2 3 int main(){ 4 int a[3]={1, 2, 3}; 5 int b=5; 6 int c=a+b; 7 printf("%d", c); 8 return 0; 9 }</pre>	<pre>1 #include<stdio.h> 2 3 int main(){ 4 int a[3]={1, 2, 3}; 5 int b=5; 6 int c=a[1]+b; 7 printf("%d", c); 8 return 0; 9 } 10</pre>
---	---

9. Дневник отладки.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
0	дом	10.11.2022	04:20	У меня ничего не произошло	Зачилился	

10. Замечания автора по существу работы

Замечаний нет

11. Выводы

Нарочно придумывать ошибки – самая сложная задача их всех. Вот как бы ты не хотел, специально их придумать трудно, да ещё и при условии, что Си всеядный и компилирует любую ерунду.

Подпись студента _____