Федеральное государственное бюджетное образовательное учреждение высшего образования «Сибирский государственный университет телекоммуникаций и информатики»

## (СибГУТИ)

Кафедра вычислительных систем

**ОТЧЕТ**

по практической работе 1

## по дисциплине «**Программирование**»

Выполнил:

студент гр. ИС-242

«8» марта 2023 г. \_\_\_\_\_\_\_\_\_\_\_\_ /Воркунов К.С./

Проверил:

Старший Преподаватель Кафедры

ВС

«10» марта 2023 г. \_\_\_\_\_\_\_\_\_\_\_\_ /Фульман В.О./

Оценка «\_\_\_\_\_\_\_\_\_\_\_\_\_»

Новосибирск 2023

# **ОГЛАВЛЕНИЕ**

[ОГЛАВЛЕНИЕ 2](#_Toc15171)

[ЗАДАНИЕ 3](#_Toc15172)

[ПРИЛОЖЕНИЕ 8](#_Toc15173)

# **ЗАДАНИЕ**

В лабораторной работе представлены задачи с кодом. Необходимо с помощью отладчика локализовать и исправить ошибки.

Задание 1:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | **#include <stdio.h>**  #include <stdlib.h>    **void** **init**(**int**\* arr, **int** n)  {  arr = malloc(n \* **sizeof**(**int**)); **int** i;  **for** (i = **0**; i < n; ++i)  {  arr[i] = i;  }  } **int** **main**()  { **int**\* arr = NULL; **int** n = **10**; init(arr, n); **int** i;  **for** (i = **0**; i < n; ++i)  {  printf("%d**\n**", arr[i]);  } **return** **0**;  } |

Задание 2:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | **#include <stdio.h>**  **typedef** **struct**  {  **char** str[**3**]; **int** num;  } NumberRepr;  **void** **format**(NumberRepr\* number)  {  sprintf(number->str, "%3d", number->num); } **int** **main**()  {  NumberRepr number = { .num = **1025** }; format(&number);  printf("str: %s**\n**", number.str); printf("num: %d**\n**", number.num); **return** **0**;  } |

Задание 3:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | **#include <stdio.h>**    #define SQR(x) x \* x  **int** **main**()  {  **int** y = **5**; **int** z = SQR(y + **1**); printf("z = %d**\n**", z); **return** **0**;  } |

Задание 4:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | **#include <stdio.h>**    **void** **swap**(**int**\* a, **int**\* b)  { **int** tmp = \*a;  \*a = \*b;  \*b = tmp;  }  **void** **bubble\_sort**(**int**\* array, **int** size)  {  **int** i, j;  **for** (i = **0**; i < size - **1**; ++i) { **for** (j = **0**; j < size - i; ++j) { **if** (array[j] > array[j + **1**]) { swap(&array[j], &array[j + **1**]);  }  }  }  } **int** **main**()  {  **int** array[**100**] = {**10**, **15**, **5**, **4**, **21**, **7**}; bubble\_sort(array, **6**); **int** i;  **for** (i = **0**; i < **6** ; ++i) { printf("%d ", array[i]);  }  printf("**\n**"); **return** **0**;  } |

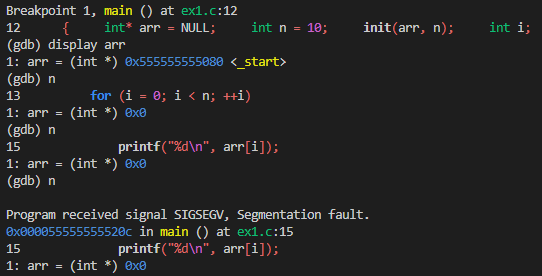
**ВЫПОЛНЕНИЕ РАБОТЫ**

Задание 1:

После компилирования и запуска программы получаем ошибку сегментирования.

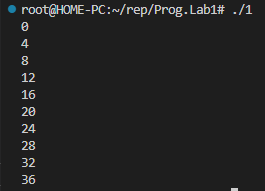


Посмотрим адреса переменных в функциях main и init c помощью отладчика gdb.



В функции init выделяется память под массив и инициализируются его элементы. Как можно увидеть на скриншоте выше, после выполнения функции init, в переменной arr в функции main остается указатель на нулевую память, что приводит к ошибке. Это происходит из-за того, что в локальной переменной arr в функции init сохранен указатель на выделенную память, но указатель не на возвращается в функцию main. Для исправления ошибки нужно возвращать указатель на выделенную память.

Посмотрим на результат работы программы после исправления ошибки.

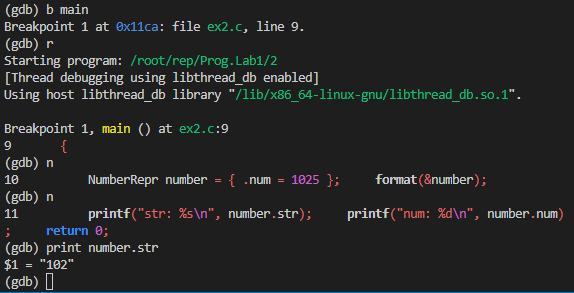


Задание 2:

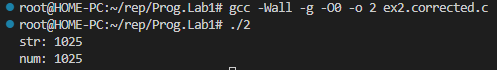
После выполнения программы элемент num структуры number меняет свое значение на 1024, хотя напрямую мы не меняли значение элемента num.



Это происходит из-за того, что мы пытаемся записать в массив char str[3], состоящий из трех элементов, четырехзначное число и символ конца строки. На скриншоте видно, что четвертый символ сохранился в выравнивающий байт, а признак конца строки в первый байт переменной num. Для устранения проблемы предлагаю выделить 5 байт для массива str.



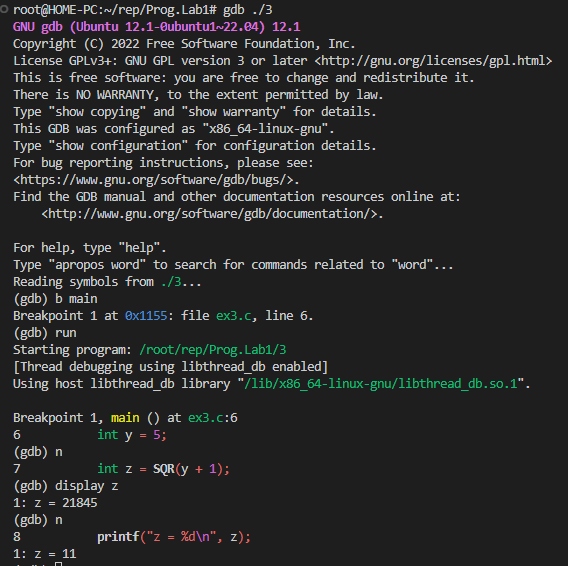
После внесенных изменений получаем такой результат:



Задание 3:

В ходе выполнения программы происходит ошибка в вычислении квадрата.



Посмотрим во что будет развернут макрос SQR. Для этого необходимо скомпилировать программу с ключом –g3. 

Ошибка происходит из-за неправильного порядка выполнения арифметических операций. Для решения предлагаю расставить скобки, чтобы порядок стал правильным.

Посмотрим на результат работы программы после исправления ошибки.



gcc

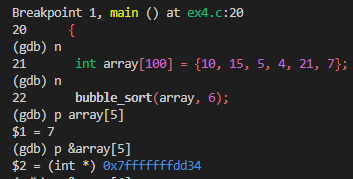
Задача 4:

После выполнения программы мы видим неправильно отсортированный массив, а так же элементы, которых не было в исходном массиве.

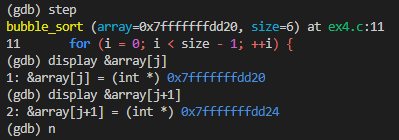


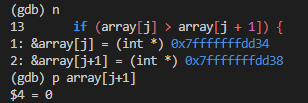
Попробуем проследить за процессом сортировки и найти ошибку.

Узнаем адрес последнего элемента.



Проходим по сортировке и видим, что сортировка сравнивает следующий элемент, после инициализированного элемента массива, который равен нулю.





Что бы этого избежать, мы добавим -1 в цикле к size



Ошибка происходит из-за того, что совершается лишняя итерация внутреннего цикла пузырьковой сортировки. Для решения необходимо добавить -1 в условии работы цикла.

После исправления ошибки программа правильно сортирует массив.



# **ПРИЛОЖЕНИЕ**

Исходный код с комментариями;

**ex1.c**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 25  26 | #include <stdio.h> #include <stdlib.h>    **int**\* **init**(**int**\* arr, **int** n)  {  arr = malloc(n \* **sizeof**(**int**)); **int** i; **for** (i = **0**; i < n; ++i)  { arr[i] = i;  } **return** arr;  } **int** **main**()  {  **int**\* arr = NULL; **int** n = **10**; arr = init(arr, n); **int** i; **for** (i = **0**; i < n; ++i)  {  printf("%d**\n**", arr[i]);  } **return** **0**;  } |

**ex2.c**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 20  21 | #include <stdio.h>  **typedef** **struct**  {  **char** str[**5**]; **int** num;  } NumberRepr;    **void** **format**(NumberRepr\* number)  { sprintf(number->str, "%3d", number->num);  } **int** **main**()  {  NumberRepr number = { .num = **1025** }; format(&number); printf("str: %s**\n**", number.str); printf("num: %d**\n**", number.num); **return** **0**;  } |

**ex3.c**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | #include <stdio.h>    #define SQR(x) (x) \* (x)  **int** **main**()  {  **int** y = **5**; **int** z = SQR(y + **1**); printf("z = %d**\n**", z); **return** **0**; |

**ex4.c**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 #include <stdio.h>  2   1. **void** **swap**(**int**\* a, **int**\* b) 2. { 3. **int** tmp = \*a; 4. \*a = \*b; 5. \*b = tmp; 6. }   9   1. **void** **bubble\_sort**(**int**\* array, **int** size) 2. { 3. **int** i, j; 4. **for** (i = **0**; i < size - **1**; ++i) { 5. **for** (j = **0**; j < size - i - **1**; ++j) { 6. **if** (array[j] > array[j + **1**]) { 7. swap(&array[j], &array[j + **1**]); 8. } 9. } 10. } 11. }   21   1. **int** **main**() 2. { 3. **int** array[**100**] = {**10**, **15**, **5**, **4**, **21**, **7**}; 4. bubble\_sort(array, **6**); 5. **int** i; 6. **for** (i = **0**; i < **6** ; ++i) {  |  |  |  | | --- | --- | --- | | printf( | "%d " | , array[i]); | | } | | |   28  29   1. printf("**\n**"); 2. **return** **0**; 3. } |