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-241

«23» марта 2023 г. \_\_\_\_\_\_\_\_\_\_\_\_ /Газиев Р.Р./

Проверил:

Ст. преп. Кафедры ВС

«5» мая 2023 г. \_\_\_\_\_\_\_\_\_\_\_\_ /Фульман В.О./

Оценка «\_\_\_\_\_\_\_\_\_\_\_\_\_»

Новосибирск 2023

ОГЛАВЛЕНИЕ

ЗАДАНИЕ.....................................................................................................2

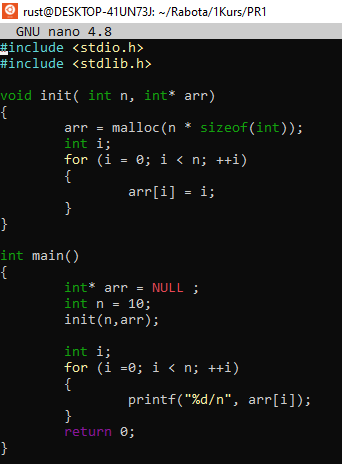
ВЫПОЛНЕНИЕ РАБОТЫ..............................................................................4

ИСПРАВЛЕННЫЕ ПРОГРАММЫ.................................................................8

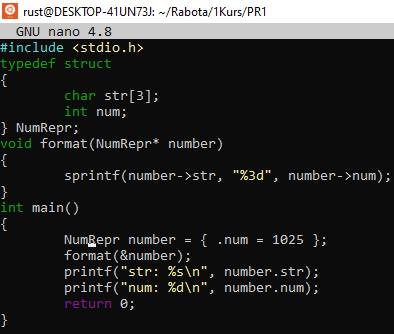
ЗАДАНИЕ

Нужно найти и исправить ошибки, используя GDB.

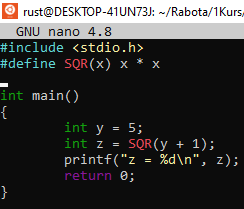
Задание 1



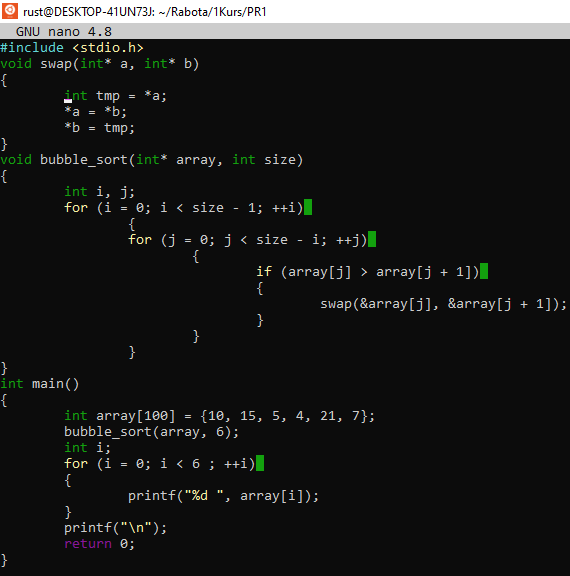
Задание 2



Задание 3



Задание 4

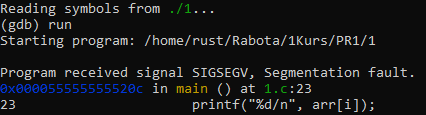


ВЫПОЛНЕНИЕ РАБОТЫ

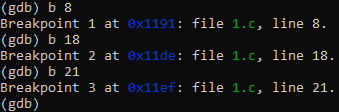
Задание 1

Дана программа, которая должна наполнить массив arr элементами, равными своему индексу. Скомпилируем её. После её компиляции мы видим, что программа скомпилировалась удачно.

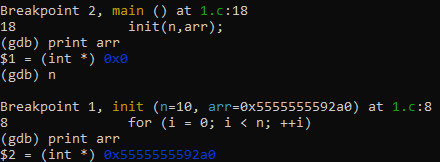
Запускаем программу в отладчике GDB. После запуска, отладчик обнаружил проблему в 23 строке:” Segmentation fault “.

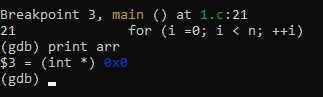


Поставим дополнительные точки остановки.

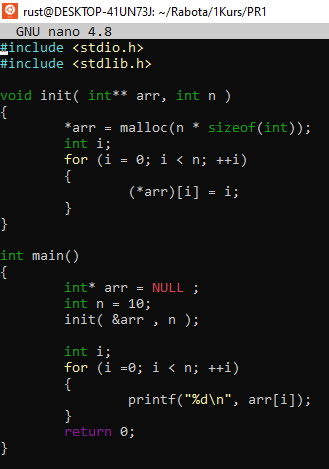


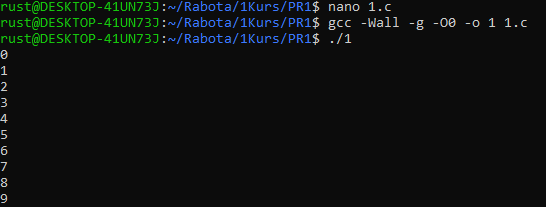
Выводя arr, после каждой остановки, видно, что arr в init и arr в main имеют разные значения, до и после выполнения функции.





Это значит, что arr в init работает только в этой функции. Для исправления требуется передать в функцию указатель на указатель.

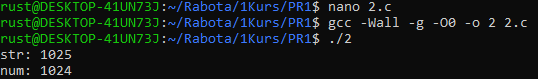




Задание 2

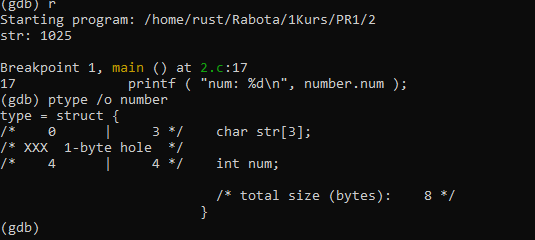
Нам дана программа, которая должна записывать строковое представление числа 1025.

После компиляции ошибок нет. Но выходные данные отличаются.



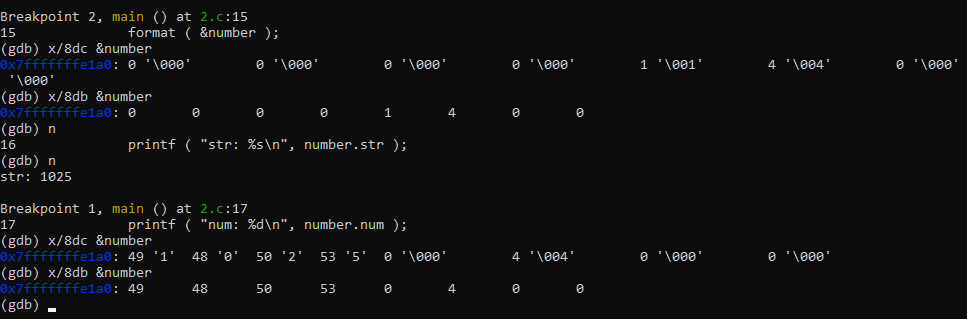
Используя отладчик узнаем, что хранилось в полях структуры перед их изменением.

Поставив точку остановки на 17 строке узнаём что всего файл занимает в памяти 8 байт, 3 из которых уходят на массив str, 4 на тип int и 1 на выравнивающий байт.

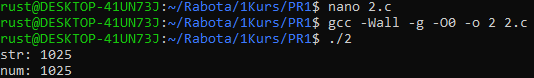


Установим точку установки до изменений полей структуры и используем команды “x/8db”, “x/8dc”, где “x” - команда, отображающая содержимое памяти по заданному адресу в указанном формате, “8” - длина значений, “d” и “c” - формат значений, “b” - модификатор, означает вывод в байтах, “&number” - адресное выражение.

После чего используем те же команды после вызова функции “format”.

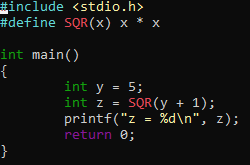


Видим, что первые 5 байт структуры изменили значение, следовательно массив str имеет пересечение с num. В результате получаем 1024 потому, что завершающий байт был записан в старший байт num. Для исправления этой ошибки нужно увеличить размер массива srt.

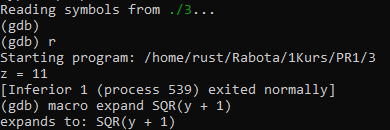


Задание 3

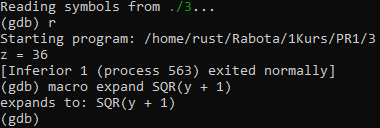
Дана программа, которая рассчитана на вывод выражения (y + 1) умноженное на саму себя.



На выходе получаем 11, но ожидали 36. При помощи отладчика открываем макрос SQR.

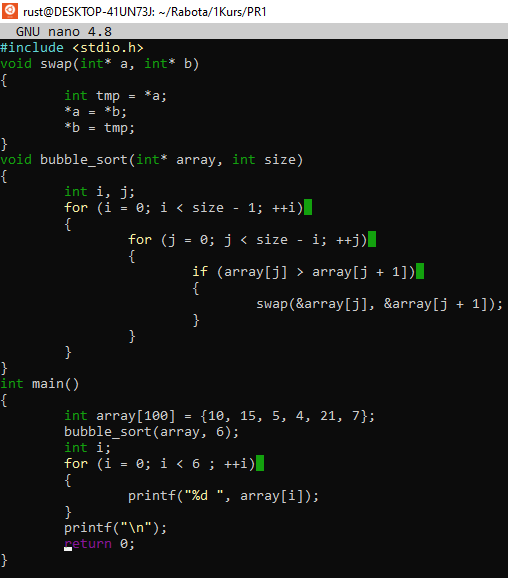


Раскрыв макрос не понятно в чем проблема, но вспомнив правила математики можем заметить, что действия выполняются не в том порядке. Это легко решить поставив скобки по бокам выражения “ y+1 “. После чего результат стал правильным.



Задание 4

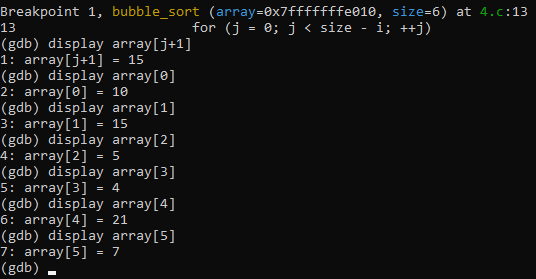
Дана программа, которая должна отсортировать массив из 6 элементов.



Но программа не работает так как надо и выдает неправильно отсортированный массив.

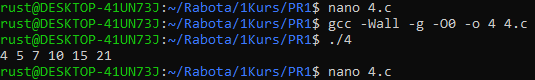


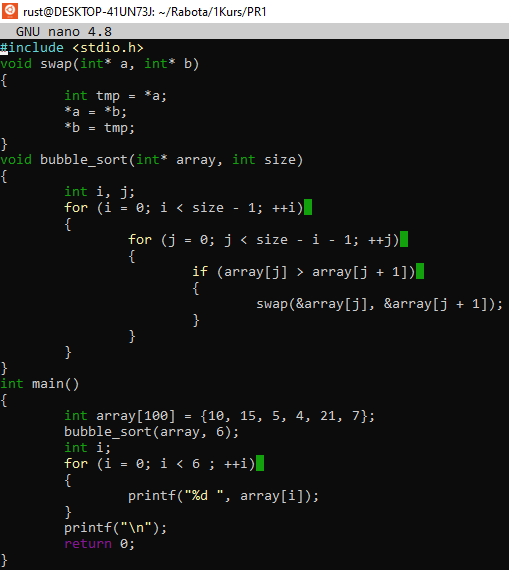
Запускаем программу через отладчик и смотрим работу цикла в функции bubble\_sort.



Используя команду отладчика display следим за работой цикла. Замечаем ошибку в работе:

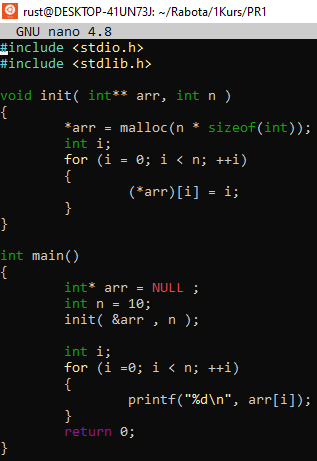
Последний элемент сравнивается со следующим. Для исправления нужно в строке “ size -I” нужно дописать “ -1 “.



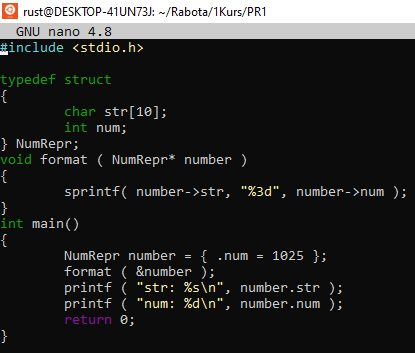


ИСПРАВЛЕННЫЕ ПРОГРАММЫ

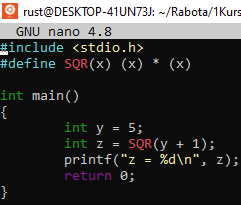
Задание 1



Задание 2



Задание 3



Задание 4

