Отладка с помощью gdb

Отладчик

Отладчик представляет собой программу, которая запускает другую программу в полностью контролируемом окружении.

Это позволяет выполнять программу шаг за шагом, наблюдая содержимое переменных, памяти и даже регистров процессора после выполнения каждого оператора.

Подготовка программы

Чтобы отладчик мог показывать имена переменных и функций, выполнить сопоставление исходного кода программы и машинного кода и т.п., в исполняемый файл нужно добавить отладочную информацию:

gcc -std=c99 -Wall -Werror **-g** main.c -o app.exe

Сеанс отладки GDB

Запуск отладчика

```
$ gdb ./app.exe
GNU gdb (GDB) 11.2
...
(gdb)
```

Отладчик загружает исполняемый файл программы и ждет инструкций

Сеанс отладки GDB

- list
- run
- break и точки останова

  Достигнув *точки останова*, отладчик прерывает выполнение программы, предоставляя возможность изучить состояние программы в этой точке.
- next
- step
- print
- continue

Использование команд GDB

- Каждая команда – строка текста, которая начинается с ключевого слова. Остальная часть строки содержит аргументы команды.
- Ключевое слово можно усечь до части, однозначно идентифицирующей команду.
- Ввод пустой строки приведет к повтору предыдущей команды.
- GDB поддерживает автоматическое завершение имен команд, файлов, переменных, функций (<Tab>).

Вывод справки о командах GDB

```
(gdb) help
List of classes of commands:

aliases -- User-defined aliases of other commands.
breakpoints -- Making program stop at certain points.
data -- Examining data.
...
Type "help" followed by a class name for a list of commands in that class.
Type "help all" for the list of all commands.
Type "help" followed by command name for full documentation.
Type "apropos word" to search for commands related to "word".
Type "apropos -v word" for full documentation of commands related to "word".
Command name abbreviations are allowed if unambiguous.
```

Установка и вывод точек останова

- break [имя_файла:]номер_строки
- break имя_функции
- break

- info breakpoints

- tbreak (вр*е*менная точка останова)

Удаление, отключение и игнорирование точек останова

- delete [номер|диапазон]
- disable [номер|диапазон]
- enable [номер|диапазон]
- ignore номер количество_итераций

Условные точки останова

- break позиция if условие

  Позиция – номер строки или имя функции.

- condition номер [условие]

  Без условия эта команда используется для удаления условия.

Анализ стека

- bt
- frame [номер]

- Большинство команд для изучения стека работают с *текущим кадром стека*.
- Когда отладчик останавливает программу в точке останова, текущий кадр стека – это кадр соответствующей функции, т.е. кадр с номеров 0 в списке трассировки.

Вывод аргументов и локальных переменных

- info frame
- info locals
- info args

Вывод данных

- print [/формат] [выражение]

```
(gdb) print r
$1 = 1
(gdb) print r=7
$2 = 7
(gdb) set variable r=1.5
...
15          printf("%f\n", area);
(gdb) print circular_area($2)
$6 = 153.93804002640002
```

Вывод содержимого блоков памяти

- x [/nfu] [адрес]

- n – сколько единиц памяти должно быть выведено
- f – спецификатор формата
- u – размер выводимой единицы памяти

Точки наблюдения

Точки наблюдения подобны точкам останова, они позволяют получать уведомления о доступах для чтения и записи к переменным.

- watch выражение

  Отладчик останавливает программу при изменении выражения.

- rwatch выражение

  Отладчик останавливает программу, когда она читает значение любого объекта, используемого в выражении.

- awatch выражение

  Отладчик останавливает программу, когда она читает или изменяет значение любого объекта, используемого в выражении.