Оценка работоспособности программы shell-sort:
При вводе данных: 1 2 3 4 5. Вывод: Output:  0 1 2 3 4 N.  Следовательно можно сделать вывод, что где-то произошел выход за границу массива.

Проверим корректность программы на следующем наборе данных: 2 1 5 -12 3 -9.  На экран выведется: Output: -12 -9 0 1 2 3 N. Ошибки нет.

Проверим также следующий набор данных: 0 1 2 3 4. Вывод:  0 0 1 2 3 N. Опять ошибка!

Обнаружено некорректное поведение, поэтому проведем отладку и коррекцию программы.

```
Breakpoint 1, shell_sort (a=0x5555555592a0, size=6) at shell-sort.c:8
8       int h = 1;
(gdb) step
11          h = h * 3 + 1;
(gdb) next
12      } while (h <= size);
(gdb) next
11          h = h * 3 + 1;
(gdb) next
12      } while (h <= size);
(gdb) next
15          h /= 3;
(gdb) next


Breakpoint 1, main (argc=6, argv=0x7fffffffe0d8) at shell-sort.c:32
32      a = (int *)malloc((argc - 1) * sizeof(int));
(gdb) display a[0]
(gdb) display a[1]
(gdb) next
33      for (i = 0; i < argc - 1; i++) {
1: a[0] = 0
2: a[1] = 0
(gdb) display a[2]
3: a[2] = 0
(gdb) next
34          a[i] = atoi(argv[i + 1]);
1: a[0] = 0
2: a[1] = 0
3: a[2] = 0
(gdb) display a[3]
4: a[3] = 0
(gdb) display a[4]
5: a[4] = 0
(gdb) display a[5]
6: a[5] = 0
(gdb) next
33      for (i = 0; i < argc - 1; i++) {
1: a[0] = 0
2: a[1] = 0
3: a[2] = 0
4: a[3] = 0
5: a[4] = 0
6: a[5] = 0
(gdb) next
34          a[i] = atoi(argv[i + 1]);
1: a[0] = 0
2: a[1] = 0
3: a[2] = 0
```

```
4: a[3] = 0
5: a[4] = 0
6: a[5] = 0
(gdb) next
33      for (i = 0; i < argc - 1; i++) {
1: a[0] = 0
2: a[1] = 1
3: a[2] = 0
4: a[3] = 0
5: a[4] = 0
6: a[5] = 0
(gdb) next
34        a[i] = atoi(argv[i + 1]);
1: a[0] = 0
2: a[1] = 1
3: a[2] = 0
4: a[3] = 0
5: a[4] = 0
6: a[5] = 0
(gdb) next
33      for (i = 0; i < argc - 1; i++) {
1: a[0] = 0
2: a[1] = 1
3: a[2] = 2
4: a[3] = 0
5: a[4] = 0
6: a[5] = 0
(gdb) next
.
.
.
38      for (i = 0; i < argc - 1; i++) {
1: a[0] = 0
2: a[1] = 0
3: a[2] = 1
4: a[3] = 2
5: a[4] = 3
6: a[5] = 4
```

В массиве находится лишнее значение. Это можно исправить, если в 36 строке в метод shell_sort передавать (a, argc-1);

Исправим ошибку и вновь введем значения: 0 1 2 3 4. Вывод: Output: 0 1 2 3 4. Все работает верно!