

Report 4

Main with integer

I begun with creating a programm with int data type

```

1  #include <stdio.h>
2
3  int main(int argc, char const *argv[])
4  {
5      int a;
6      int b;
7      int c;
8      int d;
9      int e;
10     int f;
11
12     int expr_1 = (((a * b - c) / d) + e * f);
13     int expr_2 = a - b + c * d;
14     int expr_3 = e * f;
15     int expr_4 = (c * a - d) / (b + e * f - a);
16     int result = expr_1 * expr_2 - expr_3 + expr_4;
17     return 0;
18 }
19
20
21
22 /* expr_1: d + 0
23    expr_2: no restrictions
24    expr_3: no restrictions
25    expr_4: e * f + a - b; b + 0 and a + 0 and e * f + 0
26
27    a = 3
28    b = -6
29    c = 0
30    d = 8
31    e = 10
32    f = 2
33
34    (((a * b - c) / d) + e * f) = (((3 * (-6) - 0) / 8) + 10 * 2) = 18
35    a - b + c * d = 3 + 0 + 0 * 8 = 3
36    e * f = 10 * 2 = 20
37    (c * a - d) / (b + e * f - a) = (0 * 3 - 8) / (-6 + 10 * 2 - 3) = -8 / 11 = 0
38    expr_1 * expr_2 - expr_3 + expr_4 = 18 * 3 - 20 + 0 = 142
39 */
40
41

```

```

$ gcc -g -O0 main.c -o expr
$ gdb ./expr
GNU gdb (Debian 13.2-1+b2) 13.2
Copyright (C) 2023 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or
later http://gnu.org/licenses/gpl.html This is free software: you are free to change and
redistribute it. There is NO WARRANTY, to the extent permitted by law. Type "show
copying" and "show warranty" for details. This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details. For bug reporting instructions, please
see: https://www.gnu.org/software/gdb/bugs/. --Type for more, q to quit, c to continue
without paging--c Find the GDB manual and other documentation resources online at:
http://www.gnu.org/software/gdb/documentation/.

(gdb) b 7
Breakpoint 1 at 0x1134: file main.c, line 13.

(gdb) r
Starting program: /usr/bin/expr
18
(gdb) p result
$6 = 142
(gdb)

```

Text of debugging

```
(malex-kali@MA)-[~/lab04]
```

```
$ gcc -g -O0 main.c -o expr
```

```
(malex-kali@MA)-[~/lab04]
```

```
$ gdb ./expr
GNU gdb (Debian 13.2-1+b2) 13.2
```

Copyright (C) 2023 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html> This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law. Type "show copying" and "show warranty" for details. This GDB was configured as "x86_64-linux-gnu". Type "show configuration" for configuration details. For bug reporting instructions, please see: <https://www.gnu.org/software/gdb/bugs/>. --Type for more, q to quit, c to continue without paging--c Find the GDB manual and other documentation resources online at: <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".

Type "apropos word" to search for commands related to "word"...

Reading symbols from ./expr...

```
(gdb) b 7
```

```
Breakpoint 1 at 0x1134: file main.c, line 13.
```

```
(gdb) p a
No symbol "a" in current context.
(gdb) r
Starting program: /home/malex-kali/lab04/expr
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, main (argc=1, argv=0x7ffffffdf38) at main.c:13
13 int expr_1 = (((a * b - c) / d) + e * f);
(gdb) whatis a
type = int
(gdb) p a
$1 = 32767
(gdb) p b
$2 = -134227280 (gdb) p c
$3 = 0
(gdb) p d
$4 = 0
(gdb) p e
$5 = 32767
(gdb) p f
$6 = -134321088 (gdb) set var a=3
(gdb) set var b=-6
(gdb) set var c=0
(gdb) set var d=8
(gdb) set var e=10
(gdb) set var f=2
(gdb) p a
$7 = 3
(gdb) p c
$8 = 0
(gdb) next
14 int expr_2 = a - b + c * d; (gdb) p expr_1
$9 = 18 (gdb) next 15 int expr_3 = e * f;
(gdb) p expr_2
$10 = 9 (gdb) next
16 int expr_4 = (c * a - d)/(b + e * f - a);
(gdb) p expr_3
$11 = 20 (gdb) next
17 int result = expr_1 * expr_2 - expr_3 + expr_4;
(gdb) p expr_4
$12 = 0 (gdb) n
18 return 0;
(gdb) p result
$13 = 142
(gdb) q
A debugging session is active.
```

Inferior 1 [process 4116] will be killed.

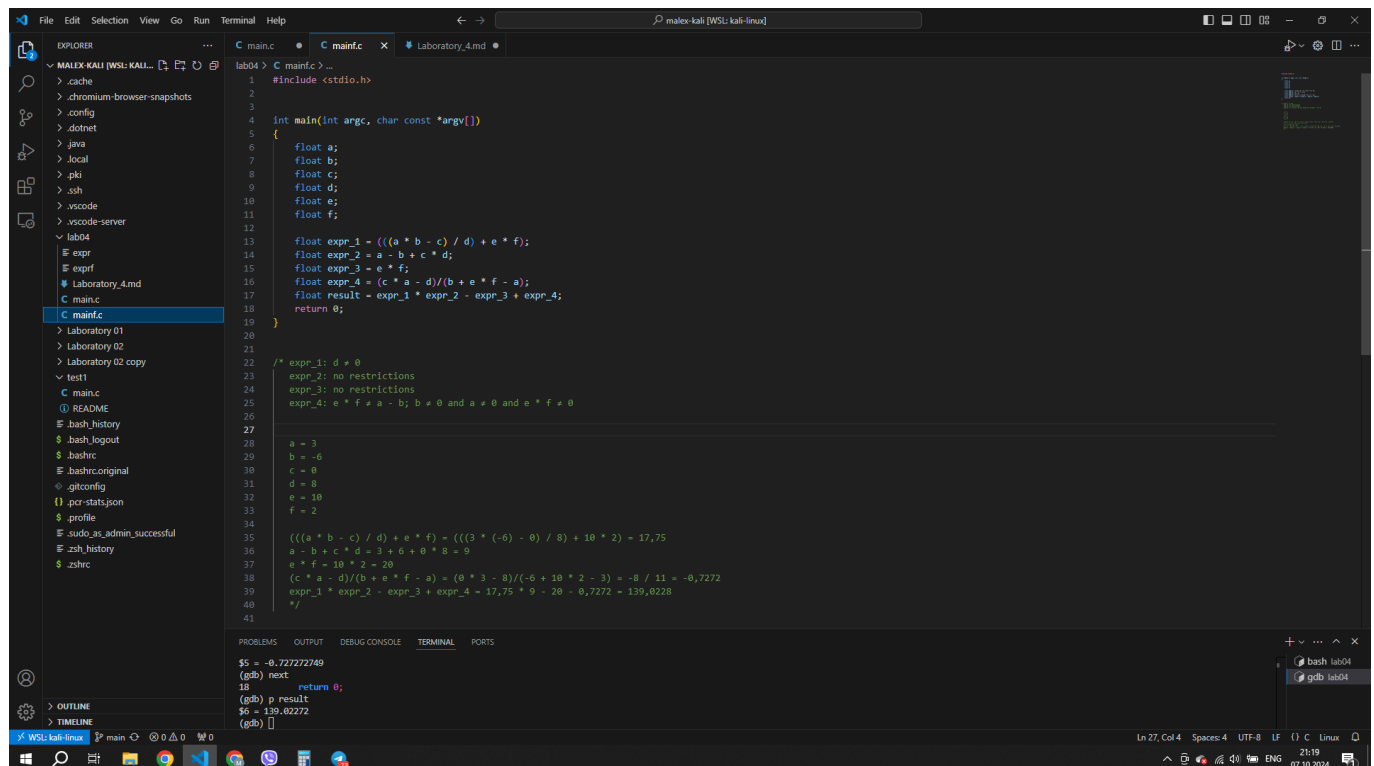
Quit anyway? (y or n) y

└─(malex-kali@MA)-[~/lab04]

└─\$

Main with float

Then I created a programm with float data type



```

1  #include <stdio.h>
2
3
4  int main(int argc, char const *argv[])
5  {
6      float a;
7      float b;
8      float c;
9      float d;
10     float e;
11     float f;
12
13     float expr_1 = (((a * b - c) / d) + e * f);
14     float expr_2 = a - b + c * d;
15     float expr_3 = e * f;
16     float expr_4 = (c * a - d) / (b + e * f - a);
17     float result = expr_1 * expr_2 - expr_3 + expr_4;
18     return 0;
19 }
20
21
22 /* expr_1: d != 0
23    expr_2: no restrictions
24    expr_3: no restrictions
25    expr_4: e * f * a - b; b != 0 and a != 0 and e * f != 0
26
27    a = 3
28    b = -6
29    c = 0
30    d = 8
31    e = 10
32    f = 2
33
34    (((a * b - c) / d) + e * f) = (((3 * (-6) - 0) / 8) + 10 * 2) = 17,75
35    a - b + c * d = 3 + 6 + 0 * 8 = 9
36    e * f = 10 * 2 = 20
37    (c * a - d) / (b + e * f - a) = (0 * 3 - 8) / (-6 + 10 * 2 - 3) = -8 / 11 = -0,7272
38    expr_1 * expr_2 - expr_3 + expr_4 = 17,75 * 9 - 20 - 0,7272 = 139,0228
39
40 */
41

```

```

$5 = -0.727272749
(gdb) next
18      return 0;
(gdb) p result
$6 = 139.02272
(gdb)

```

Text of debugging

└─(malex-kali@MA)-[~/lab04]

└─\$ cd /home/malex-kali/lab04

└─(malex-kali@MA)-[~/lab04]

└─\$ gcc -g -O0 mainf.c -o exprf

└─(malex-kali@MA)-[~/lab04]

└─\$ gdb ./exprf

GNU gdb (Debian 13.2-1+b2) 13.2

Copyright (C) 2023 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

Type "show copying" and "show warranty" for details.

This GDB was configured as "x86_64-linux-gnu".

Type "show configuration" for configuration details.

For bug reporting instructions, please see:

<https://www.gnu.org/software/gdb/bugs/>.

--Type for more, q to quit, c to continue without paging--c

Find the GDB manual and other documentation resources online at:

<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".

Type "apropos word" to search for commands related to "word"...

Reading symbols from ./exprf...

(gdb) b main

Breakpoint 1 at 0x1134: file mainf.c, line 13.

(gdb) r

Starting program: /home/malex-kali/lab04/exprf

[Thread debugging using libthread_db enabled]

Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, main (argc=1, argv=0x7ffffffdf38) at mainf.c:13

13 float expr_1 = (((a * b - c) / d) + e * f);

(gdb) ptype a

type = float

(gdb) set var a=3

(gdb) set var b=-6

(gdb) set var c=0

(gdb) set var d=8

(gdb) set var e=10

(gdb) Quit

(gdb) set var f=2

(gdb) p a

\$1 = 3

(gdb) next

14 float expr_2 = a - b + c * d;

(gdb) p expr_1

\$2 = 17.75

(gdb) next

15 float expr_3 = e * f;

(gdb) p expr_2

\$3 = 9

(gdb) next

16 float expr_4 = (c * a - d)/(b + e * f - a);

(gdb) p expr_3

\$4 = 20

(gdb) next

17 float result = expr_1 * expr_2 - expr_3 + expr_4;

(gdb) p expr_4

\$5 = -0.727272749

(gdb) next

18 return 0;

(gdb) p result

```
$6 = 139.02272
```

```
(gdb) q
```

```
A debugging session is active.
```

```
Inferior 1 [process 10490] will be killed.
```

```
Quit anyway? (y or n) y
```

```
└─(malex-kali@MA)-[~/lab04]
```

```
└─$
```