Шабалкин Виталий Витальевич

БПИ-214

Вариант №33

Задача: сформировать массив B на основе элементов массива A, полученных как разность соседних элементов.

Дополнение с функцией принимающей несколько аргументов.

Код на С:

```
#include <stdio.h>


void create_a(int n, int arr[]){
printf("Enter array A\n");
for(int i = 0; i < n; i++){
printf("a[%d]: ", i);
scanf("%d", &arr[i]);
}
}


int main()
{
int ek = 1;
int n;
while(ek == 1){
printf("Enter n(2-100): ");
scanf("%d", &n);
if(n >= 2 && n <= 100){
ek = 0;
break;
} else {
printf("Incorrect input, try again\n");
}
}
int a [100];
int b [100];
create_a(n,a);
printf("Generated array B:\n");
for(int j = 0; j < n - 1; j++){
b[j] = a[j] - a[j + 1];
printf("b[%d]: %d\n", j, b[j]);
}
return 0;
}
```

Код GAS:

```
                .file "idz_na_c_1_wfunction.c"
.text
.section .rodata
.LC0:
.string   "Enter array A"
.LC1:
.string   "a[%d]: "
.LC2:
.string   "%d"
.text
.globl    create_a
.type     create_a, @function
create_a:                                         #Добавленная функция
endbr64
pushq     %rbp
movq      %rsp, %rbp
subq      $32, %rsp
movl      %edi, -20(%rbp)
```

```
movq    %rsi, -32(%rbp)
leaq    .LC0(%rip), %rax
movq    %rax, %rdi
call    puts@PLT
movl    $0, -4(%rbp)
jmp     .L2
.L3:
movl    -4(%rbp), %eax
movl    %eax, %esi
leaq    .LC1(%rip), %rax
movq    %rax, %rdi
movl    $0, %eax
call    printf@PLT
movl    -4(%rbp), %eax
cltq
leaq    0(,%rax,4), %rdx
movq    -32(%rbp), %rax
addq    %rdx, %rax
movq    %rax, %rsi
leaq    .LC2(%rip), %rax
movq    %rax, %rdi
movl    $0, %eax
call    __isoc99_scanf@PLT
addl    $1, -4(%rbp)
.L2:
movl    -4(%rbp), %eax
cmpl    -20(%rbp), %eax
jl      .L3
nop
nop
leave
```

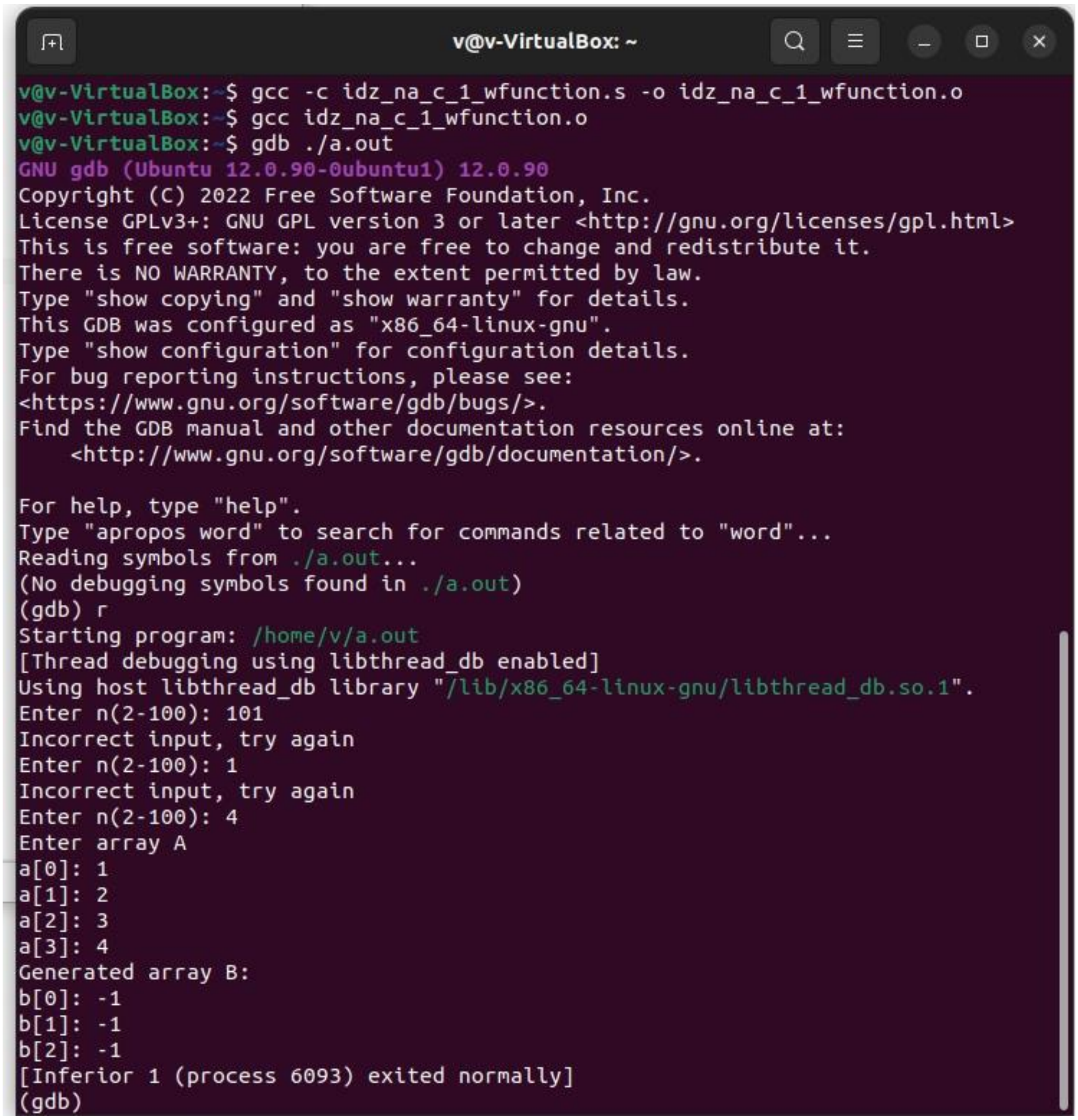
```
ret
.size     create_a, .-create_a
.section.rodata
.LC3:
.string   "Enter n(2-100): "
.LC4:
.string   "Incorrect input, try again"
.LC5:
.string   "Generated array B:"
.LC6:
.string   "b[%d]: %d\n"
.text
.globl    main
.type     main, @function
main:
endbr64
pushq     %rbp
movq      %rsp, %rbp
subq      $832, %rsp
movq      %fs:40, %rax
movq      %rax, -8(%rbp)
xorl      %eax, %eax
movl      $1, -820(%rbp)
jmp       .L5
.L8:
leaq      .LC3(%rip), %rax
movq      %rax, %rdi
movl      $0, %eax
call      printf@PLT
leaq      -828(%rbp), %rax
movq      %rax, %rsi
```

```
leaq    .LC2(%rip), %rax
movq    %rax, %rdi
movl    $0, %eax
call    __isoc99_scanf@PLT
movl    -828(%rbp), %eax
cmpl    $1, %eax
jle     .L6
movl    -828(%rbp), %eax
cmpl    $100, %eax
jg      .L6
movl    $0, -820(%rbp)
jmp     .L7
.L6:
leaq    .LC4(%rip), %rax
movq    %rax, %rdi
call    puts@PLT
.L5:
cmpl    $1, -820(%rbp)
je      .L8
.L7:
movl    -828(%rbp), %eax            #Передача аргументов в функцию
leaq    -816(%rbp), %rdx              #
movq    %rdx, %rsi                    #
movl    %eax, %edi                    #
call    create_a                     #Вызов функции
leaq    .LC5(%rip), %rax
movq    %rax, %rdi
call    puts@PLT
movl    $0, -824(%rbp)
jmp     .L9
.L10:
```

```
movl    -824(%rbp), %eax
cltq
movl    -816(%rbp,%rax,4), %edx
movl    -824(%rbp), %eax
addl    $1, %eax
cltq
movl    -816(%rbp,%rax,4), %eax
subl    %eax, %edx
movl    -824(%rbp), %eax
cltq
movl    %edx, -416(%rbp,%rax,4)
movl    -824(%rbp), %eax
cltq
movl    -416(%rbp,%rax,4), %edx
movl    -824(%rbp), %eax
movl    %eax, %esi
leaq    .LC6(%rip), %rax
movq    %rax, %rdi
movl    $0, %eax
call    printf@PLT
addl    $1, -824(%rbp)
.L9:
movl    -828(%rbp), %eax
subl    $1, %eax
cmpl    %eax, -824(%rbp)
jl      .L10
movl    $0, %eax
movq    -8(%rbp), %rdx
subq    %fs:40, %rdx
je      .L12
call    __stack_chk_fail@PLT
```

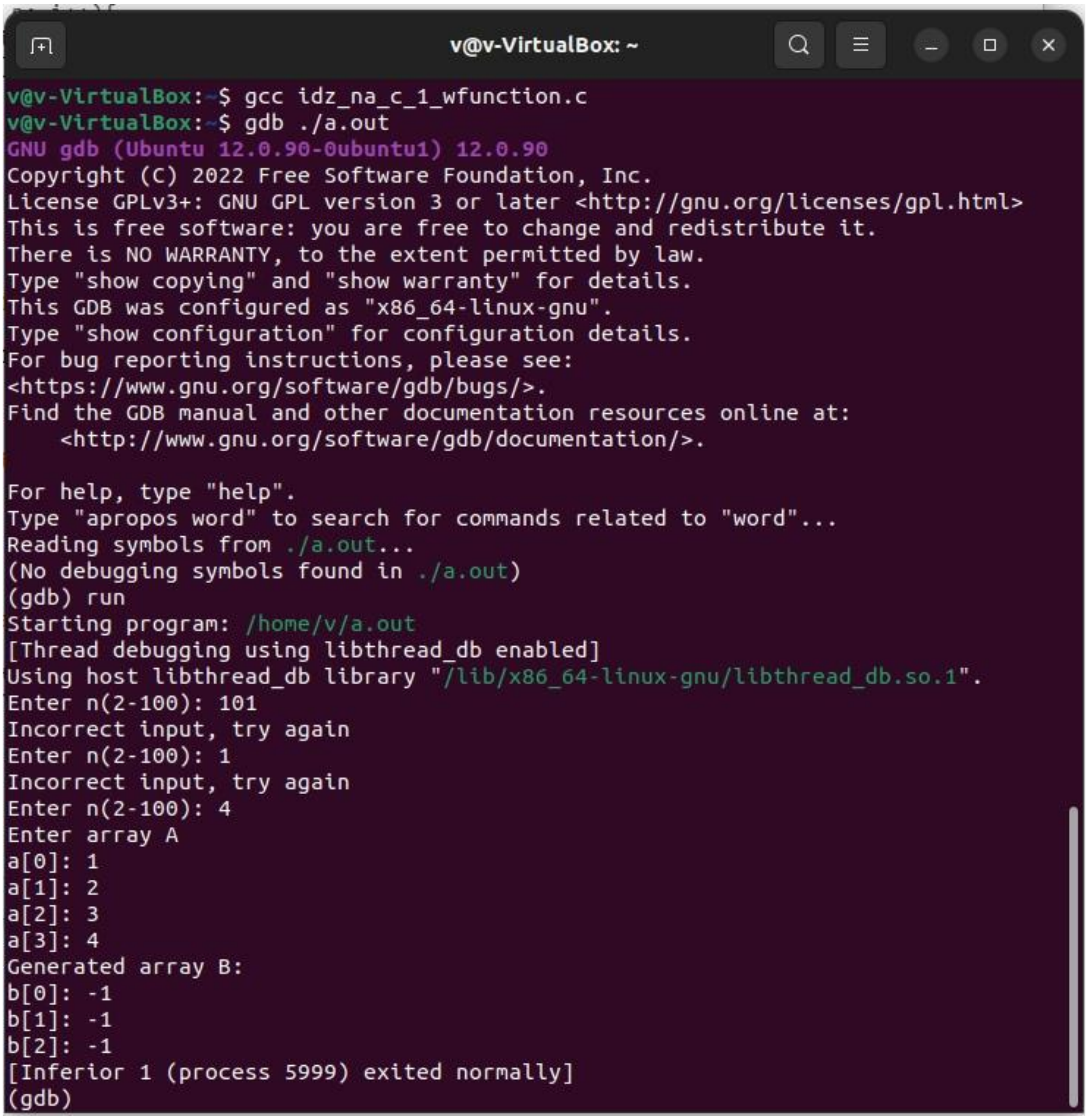
```
.L12:
leave
ret
.size	main, .-main
.ident	"GCC: (Ubuntu 11.2.0-19ubuntu1) 11.2.0"
.section.note.GNU-stack,"",@progbits
.section.note.gnu.property,"a"
.align 8
.long	1f - 0f
.long	4f - 1f
.long	5
0:
.string	"GNU"
1:
.align 8
.long	0xc0000002
.long	3f - 2f
2:
.long	0x3
3:
.align 8
4:
```

Сравнительные результаты:



```
v@v-VirtualBox:~$ gcc -c idz_na_c_1_wfunction.s -o idz_na_c_1_wfunction.o
v@v-VirtualBox:~$ gcc idz_na_c_1_wfunction.o
v@v-VirtualBox:~$ gdb ./a.out
GNU gdb (Ubuntu 12.0.90-0ubuntu1) 12.0.90
Copyright (C) 2022 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/>.
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 ./a.out...
(No debugging symbols found in ./a.out)
(gdb) r
Starting program: /home/v/a.out
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Enter n(2-100): 101
Incorrect input, try again
Enter n(2-100): 1
Incorrect input, try again
Enter n(2-100): 4
Enter array A
a[0]: 1
a[1]: 2
a[2]: 3
a[3]: 4
Generated array B:
b[0]: -1
b[1]: -1
b[2]: -1
[Inferior 1 (process 6093) exited normally]
(gdb)
```



```
v@v-VirtualBox:~$ gcc idz_na_c_1_wfunction.c
v@v-VirtualBox:~$ gdb ./a.out
GNU gdb (Ubuntu 12.0.90-0ubuntu1) 12.0.90
Copyright (C) 2022 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/>.
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 ./a.out...
(No debugging symbols found in ./a.out)
(gdb) run
Starting program: /home/v/a.out
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
Enter n(2-100): 101
Incorrect input, try again
Enter n(2-100): 1
Incorrect input, try again
Enter n(2-100): 4
Enter array A
a[0]: 1
a[1]: 2
a[2]: 3
a[3]: 4
Generated array B:
b[0]: -1
b[1]: -1
b[2]: -1
[Inferior 1 (process 5999) exited normally]
(gdb)
```