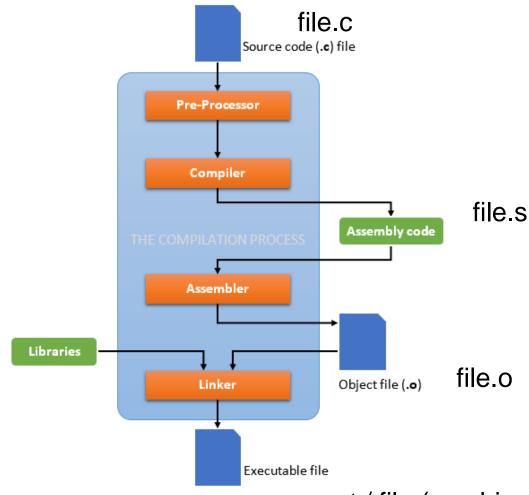
Disassembly

תרגול 7 ניתוח קוד

Reminder - Compilation Process



a.out / file (machine code)

m

How to - Disassembly of code

- Compilation of code:
 - □ gcc code.c
 - □ We get the file: a.out
- Disassembly:
 - □ objdump -d a.out
 - We get an assembly-like code that represents the c code appeared in file code.c
 - □ objdump –t a.out
 - □ This will print out the symbol table of the file. The symbol table includes the names of all functions and global variables in the file, the names of all the functions being called by the file, and their addresses.
 - □ strings a.out will print all strings in the code



Basic:

- Many times we work with an executive file and we are interested in the code that behind it.
- We can use the disassembly option or the debugger option in order to analyze the executive file and understand what is does.
- Sometimes we want to use both options.
- Disassembly enable us to get an assembly-like file that represent the activity of the executive file.



Important aspects

- In disassembly we only get the code of the functions in the files and functions that were used by the files.
- We don't get the code of the system's functions (printf, scanf...).
- We don't get the values of global constants or strings.
- Many times there are optimizations or nops added by the compiler – what make is harder to understand. For example,
 - nop
 - xchg %cx, %cx



An example

- While using disassember there are many global general functions added (init, start) usually we don't care about them.
- Show Tirgul6d_disass.txt



```
0000000000400610 <main>:
  400610:
                55
                                                %rbp
                                        push
                48 89 e5
  400611:
                                                %rsp,%rbp
                                        mov
                b8 00 00 00 00
                                                $0x0,%eax
  400614:
                                        mov
  400619:
                e8 9f ff ff ff
                                        callq
                                                4005bd <hello>
                bf 04 00 00 00
                                                $0x4,%edi
  40061e:
                                        mov
                e8 a5 ff ff ff
                                                4005cd <even>
  400623:
                                        callq
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                                %rbp
                                        pop
  40062e:
                c3
                                        retq
  40062f:
                90
                                        nop
```



```
0000000000400610 <main>:
  400610:
                                                %rbp
                55
                                        push
                48 89 e5
  400611:
                                                %rsp,%rbp
                                        mov
                b8 00 00 00 00
                                                $0x0,%eax
  400614:
                                        mov
  400619:
                e8 9f ff ff ff
                                        callq
                                                4005bd <hello>
                bf 04 00 00 00
                                                $0x4,%edi
  40061e:
                                        mov
                e8 a5 ff ff ff
                                                4005cd <even>
  400623:
                                        callq
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                                %rbp
                                        pop
  40062e:
                c3
                                        retq
  40062f:
                90
                                        nop
```



```
0000000000400610 <main>:
  400610:
                55
                                                %rbp
                                        push
                48 89 e5
  400611:
                                                %rsp,%rbp
                                        mov
                b8 00 00 00 00
                                                $0x0,%eax
  400614:
                                        mov
                e8 9f ff ff ff
  400619:
                                        callq
                                                4005bd <hello>
                bf 04 00 00 00
                                                $0x4,%edi
  40061e:
                                        mov
                e8 a5 ff ff ff
                                                4005cd <even>
  400623:
                                        callq
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                                %rbp
                                        pop
  40062e:
                c3
                                        retq
  40062f:
                90
                                        nop
```



```
0000000000400610 <main>:
  400610:
                55
                                                %rbp
                                        push
                48 89 e5
  400611:
                                                %rsp,%rbp
                                        mov
                b8 00 00 00 00
                                                $0x0,%eax
  400614:
                                        mov
                e8 9f ff ff ff
  400619:
                                        callq
                                                4005bd <hello>
               bf 04 00 00 00
  40061e:
                                                $0x4,%edi
                                        mov
                e8 a5 ff ff ff
                                                4005cd <even>
  400623:
                                        callq
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                                %rbp
                                        pop
  40062e:
                c3
                                        retq
  40062f:
                90
                                        nop
```



```
0000000000400610 <main>:
  400610:
                55
                                                %rbp
                                         push
                48 89 e5
                                                %rsp,%rbp
  400611:
                                        mov
                b8 00 00 00 00
                                                $0x0, %eax
  400614:
                                        mov
  400619:
                e8 9f ff ff ff
                                        callq
                                                4005bd <hello>
                bf 04 00 00 00
                                                $0x4,%edi
  40061e:
                                        mov
                e8 a5 ff ff ff
                                        callq
                                                4005cd <even>
  400623:
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                                %rbp
                                         pop
  40062e:
                c3
                                         retq
  40062f:
                90
                                         nop
```



hello:

```
00000000004005bd <hello>:
 4005bd:
               55
                                      push
                                             %rbp
              48 89 e5
 4005be:
                                             %rsp,%rbp
                                      mov
 4005c1:
               bf b4 06 40 00
                                             $0x4006b4,%edi
                                      mov
          e8 c5 fe ff ff
 4005c6:
                                             400490 <puts@plt>
                                      callq
 4005cb:
               5d
                                             %rbp
                                      pop
 4005cc:
               c3
                                      retq
```

- Address 0x4006b4 does not appear in the disassembly code we can see.
 - What does that tell us?
 - How can we find out what is its value?



hello:

```
00000000004005bd <hello>:
 4005bd:
               55
                                      push
                                             %rbp
              48 89 e5
 4005be:
                                             %rsp,%rbp
                                      mov
 4005c1:
               bf b4 06 40 00
                                             $0x4006b4,%edi
                                      mov
          e8 c5 fe ff ff
 4005c6:
                                      callq
                                             400490 <puts@plt:
 4005cb:
               5d
                                             %rbp
                                      pop
 4005cc:
               c3
                                      retq
```

Function "puts" is a simplified version of the printf() function. It doesn't have all printf formats and it always put the newline character in the end of its strings.



```
0000000000400610 <main>:
  400610:
                55
                                               %rbp
                                        push
                48 89 e5
  400611:
                                               %rsp,%rbp
                                        mov
                b8 00 00 00 00
                                               $0x0,%eax
  400614:
                                        mov
  400619:
                e8 9f ff ff ff
                                        callq
                                                4005bd <hello>
                bf 04 00 00 00
                                                $0x4,%edi
  40061e:
                                        mov
                e8 a5 ff ff ff
                                        callq
                                                4005cd <even>
  400623:
                b8 00 00 00 00
  400628:
                                                $0x0,%eax
                                        mov
  40062d:
                5d
                                               %rbp
                                        pop
  40062e:
                c3
                                        retq
  40062f:
                90
                                        nop
```



00000000004005cd <even>: even: 4005cd: 55 push %rbp 4005ce: 48 89 e5 %rsp,%rbp mov 48 83 ec 20 4005d1: sub \$0x20,%rsp 4005d5: 89 7d ec %edi,-0x14(%rbp) mov 4005d8: c7 45 fc 00 00 00 00 \$0x0,-0x4(%rbp)movl eb 1d 4005fe <even+0x31> 4005df: jmp 8b 45 fc -0x4(%rbp),%eax 4005e1: mov 8b 55 ec -0x14(%rbp),%edx 4005e4: mov 4005e7: 01 d0 add %edx,%eax 4005e9: 89 c6 %eax,%esi mov bf c1 06 40 00 4005eb: mov \$0x4006c1,%edi 4005f0: b8 00 00 00 00 \$0x0,%eax mov e8 a6 fe ff ff 4005f5: calla 4004a0 <printf@plt> 4005fa: 83 45 fc 02 addl \$0x2,-0x4(%rbp)4005fe: 83 7d fc 09 \$0x9,-0x4(%rbp)cmpl 400602: 7e dd jle 4005e1 <even+0x14> 0a 00 00 00 \$0xa,%edi 400604: mov e8 72 fe ff ff 400609: callq 400480 <putchar@plt> 40060e: **c9** leaveg

retq

What kind of a loop is it?

c3

40060f:





```
push
       %rbp
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq 4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
       $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveq
retq
```

rsp Old rbp





```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
       $0xa,%edi
mov
callq
      400480 <putchar@plt:
leaveg
retq
```

rsp Old rbp rbp





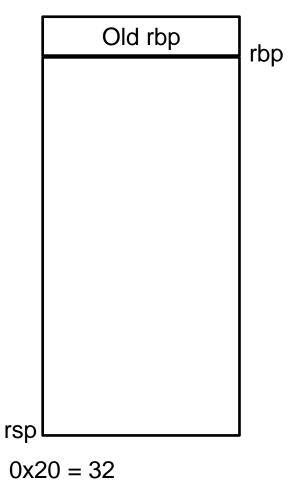
```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
       $0xa,%edi
mov
callq
      400480 <putchar@plt:
leaveg
retq
```

rsp Old rbp rbp

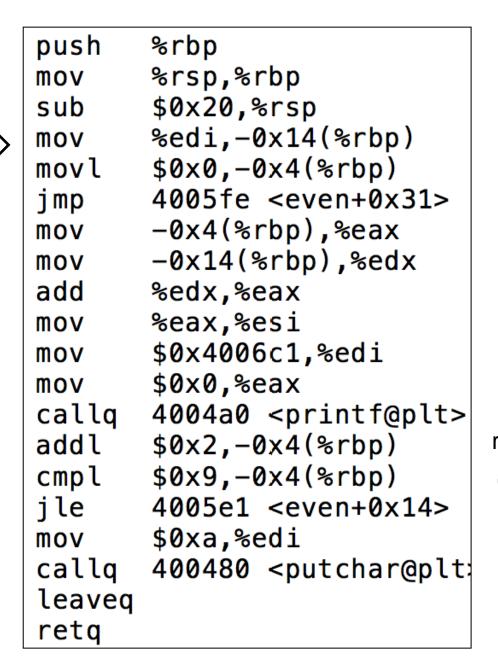


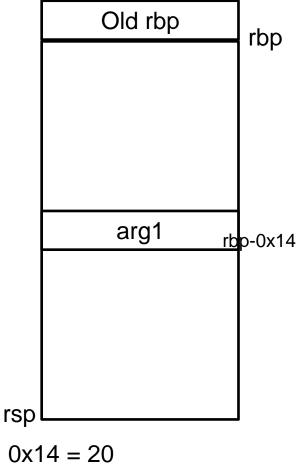


```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
      $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveg
retq
```

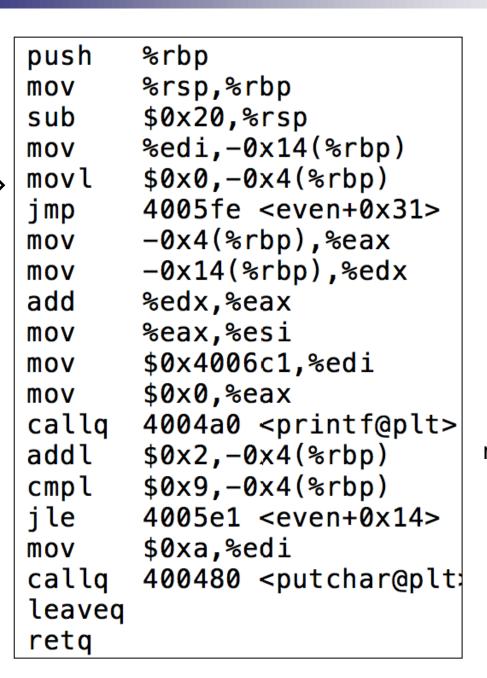


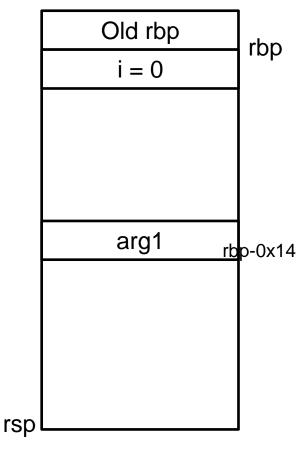




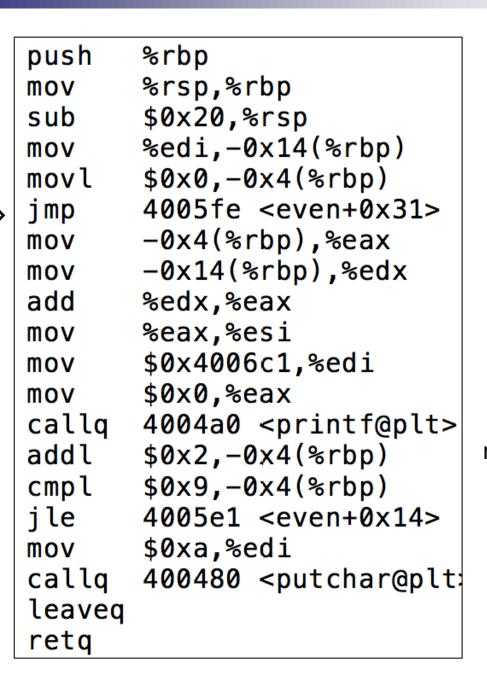


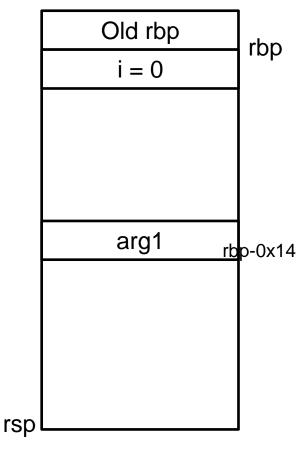






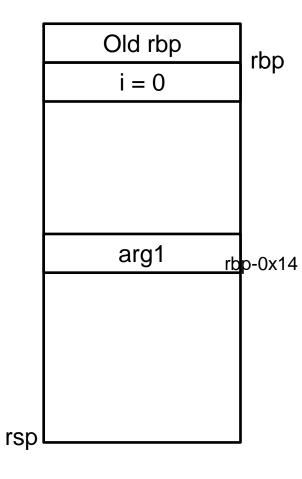






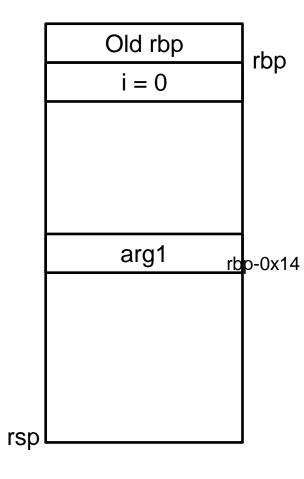


push	%rbp
mov	%rsp,%rbp
sub	\$0x20,%rsp
mov	%edi,-0x14(%rbp)
movl	\$0x0,-0x4(%rbp)
jmp	4005fe <even+0x31></even+0x31>
mov	-0x4(%rbp),%eax
mov	-0x14(%rbp),%edx
add	%edx,%eax
mov	%eax,%esi
mov	\$0x4006c1,%edi
mov	\$0x0,%eax
callq	4004a0 <printf@plt></printf@plt>
addl	\$0x2 , -0x4(%rbp)
cmpl	\$0x9 , -0x4(%rbp)
jle	4005e1 <even+0x14></even+0x14>
mov	\$0xa,%edi
callq	400480 <putchar@plt:< td=""></putchar@plt:<>
leaveq	
retq	
	mov sub mov movl jmp mov add mov mov callq addl cmpl jle mov callq leaveq





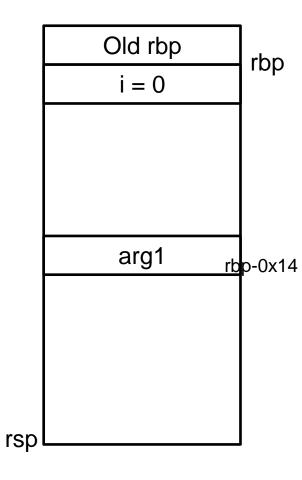
push	%rbp
mov	%rsp,%rbp
sub	\$0x20,%rsp
mov	%edi,-0x14(%rbp)
movl	\$0x0,-0x4(%rbp)
jmp	4005fe <even+0x31></even+0x31>
mov	-0x4(%rbp),%eax
mov	-0x14(%rbp),%edx
add	%edx,%eax
mov	%eax,%esi
mov	\$0x4006c1,%edi
mov	\$0x0,%eax
callq	4004a0 <printf@plt></printf@plt>
addl	\$0x2,-0x4(%rbp)
cmpl	\$0x9,-0x4(%rbp)
jle	4005e1 <even+0x14></even+0x14>
mov	\$0xa,%edi
callq	400480 <putchar@plt:< td=""></putchar@plt:<>
leaveq	
retq	





$eax \leftarrow i$

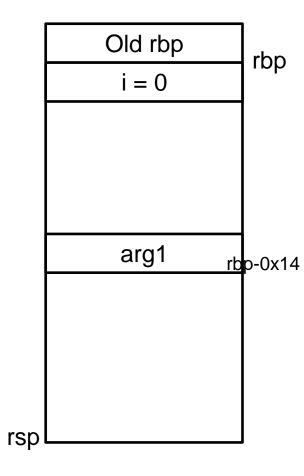
%rbp push %rsp,%rbp mov \$0x20,%rsp sub %edi,-0x14(%rbp) mov \$0x0,-0x4(%rbp)movl 4005fe <even+0x31> jmp -0x4(%rbp),%eax mov -0x14(%rbp),%edx mov %edx,%eax add %eax,%esi mov \$0x4006c1,%edi mov \$0x0,%eax mov callq 4004a0 <printf@plt> \$0x2,-0x4(%rbp)addl \$0x9,-0x4(%rbp) cmpl ile 4005e1 <even+0x14> \$0xa,%edi mov callq 400480 <putchar@plt: leaveg retq





$\underline{eax \leftarrow i} \\ \underline{edx \leftarrow arg1}$

```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
       4005e1 <even+0x14>
ile
      $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveg
retq
```

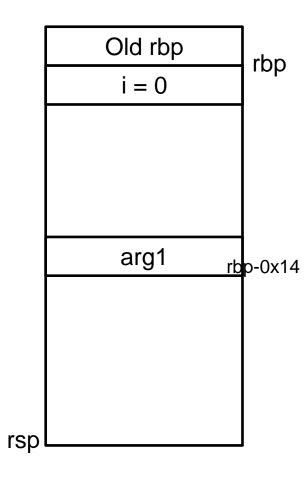




```
\frac{eax \leftarrow i}{edx \leftarrow arg1} \\
\underline{eax \leftarrow arg1 + i}
```



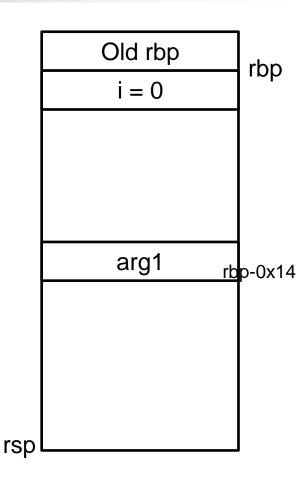
```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
      $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveg
retq
```





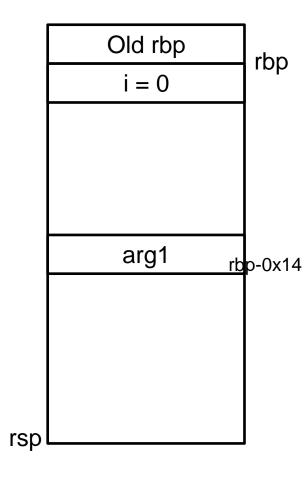
```
eax \leftarrow i
edx \leftarrow arg1
eax \leftarrow arg1 + i
esi \leftarrow arg1 + i
```

```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
       $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveg
```



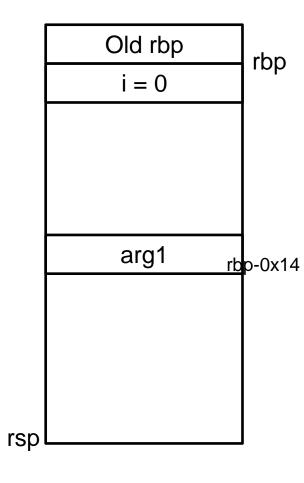


```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
       $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
       $0xa,%edi
mov
      400480 <putchar@plt:
callq
leaveg
```





```
%rbp
push
       %rsp,%rbp
mov
       $0x20,%rsp
sub
       %edi,-0x14(%rbp)
mov
       $0x0,-0x4(%rbp)
movl
       4005fe <even+0x31>
jmp
       -0x4(%rbp),%eax
mov
       -0x14(%rbp),%edx
mov
       %edx,%eax
add
       %eax,%esi
mov
       $0x4006c1,%edi
mov
       $0x0,%eax
mov
callq
      4004a0 <printf@plt>
       $0x2,-0x4(%rbp)
addl
      $0x9,-0x4(%rbp)
cmpl
ile
       4005e1 <even+0x14>
     $0xa,%edi
mov
callq 400480 <putchar@plt:
leaveq
```

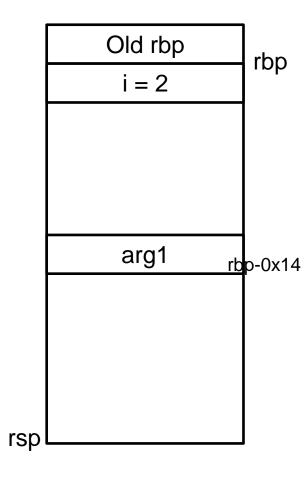




```
push
                       %rbp
                                                        Old rbp
                                                                    rbp
                       %rsp,%rbp
              mov
                                                         i = 0
eax \leftarrow i
                       $0x20,%rsp
              sub
edx \leftarrow arg1
                       %edi,-0x14(%rbp)
              mov
eax \leftarrow arg1 + i
                       $0x0,-0x4(%rbp)
              movl
esi \leftarrow arg1 + i
                       4005fe <even+0x31>
              jmp
                       -0x4(%rbp),%eax
              mov
                       -0x14(%rbp),%edx
                                                         arg1
              mov
                                                                  rbp-0x14
                       %edx,%eax
              add
                       %eax,%esi
              mov
                       $0x4006c1,%edi
              mov
                       $0x0,%eax
              mov
               printf("The number is \%d", arg1 + i);
                                                  βp
              CIIID L
                       $0X9,-0X4(8rbp)
              ile
                       4005e1 <even+0x14>
                      $0xa,%edi
              mov
              callq 400480 <putchar@plt:
              leaveg
              retq
```

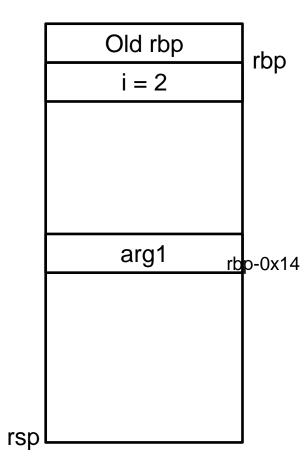


```
%rbp
             push
                     %rsp,%rbp
             mov
eax \leftarrow i
                      $0x20,%rsp
             sub
edx \leftarrow arg1
                     %edi,-0x14(%rbp)
             mov
                      $0x0,-0x4(%rbp)
eax \leftarrow arg1 + i
             movl
esi \leftarrow arg1 + i
                      4005fe <even+0x31>
             jmp
                     -0x4(%rbp),%eax
             mov
                     -0x14(%rbp),%edx
             mov
                     %edx,%eax
             add
                     %eax,%esi
             mov
                      $0x4006c1,%edi
             mov
                     $0x0,%eax
             mov
             callq
                    4004a0 <printf@plt>
                     $0x2,-0x4(%rbp)
             addl
                     $0x9,-0x4(%rbp)
             cmpl
                      4005e1 <even+0x14>
             ile
                     $0xa,%edi
             mov
             callq 400480 <putchar@plt:
             leaveq
```



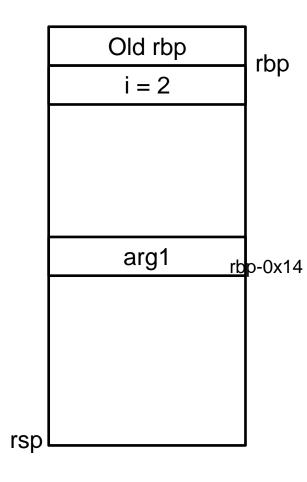


	push	%rbp
oar (i	mov	%rsp,%rbp
$eax \leftarrow i$	sub	\$0x20,%rsp
$\underline{edx} \leftarrow \underline{arg1}$	mov	%edi,-0x14(%rbp)
$eax \leftarrow arg1 + i$	movl	\$0x0,-0x4(%rbp)
$\underline{esi} \leftarrow \underline{arg1} + \underline{i}$	jmp	4005fe <even+0x31></even+0x31>
	mov	-0x4(%rbp),%eax
	mov	-0x14(%rbp),%edx
	add	%edx,%eax
	mov	%eax,%esi
	mov	\$0x4006c1,%edi
	mov	\$0x0,%eax
	callq	4004a0 <printf@plt></printf@plt>
	addl	\$0x2 , -0x4(%rbp)
	cmpl	\$0x9 , -0x4(%rbp)
	jle	4005e1 <even+0x14></even+0x14>
	mov	\$0xa,%edi
	callq	400480 <putchar@plt:< th=""></putchar@plt:<>
	leaveq	
	retq	
•		





_		
	push	%rbp
oar – i	mov	%rsp,%rbp
$\underline{eax} \leftarrow \underline{i}$	sub	\$0x20,%rsp
$\underline{edx} \leftarrow \underline{arg1}$	mov	%edi,-0x14(%rbp)
$eax \leftarrow arg1 + i$	movl	\$0x0,-0x4(%rbp)
$esi \leftarrow arg1 + i$	jmp	4005fe <even+0x31></even+0x31>
	mov	-0x4(%rbp),%eax
	mov	-0x14(%rbp),%edx
	add	%edx,%eax
	mov	%eax,%esi
	mov	\$0x4006c1,%edi
	mov	\$0x0,%eax
	callq	4004a0 <printf@plt></printf@plt>
	addl	\$0x2 , -0x4(%rbp)
$\underline{i} \leq 9$?	cmpl	\$0x9 , -0x4(%rbp)
	j le	4005e1 <even+0x14></even+0x14>
,	mov	\$0xa,%edi
	callq	400480 <putchar@plt:< th=""></putchar@plt:<>
	leaveq	
	retq	





```
0000000000400610 <main>:
                                        push
                                                %rbp
  400610:
                55
  400611:
                48 89 e5
                                                %rsp,%rbp
                                        mov
                b8 00 00 00 00
  400614:
                                                $0x0,%eax
                                        mov
  400619:
                e8 9f ff ff ff
                                        callq
                                                4005bd <hello>
                                                $0x4,%edi
  400<sup>-1</sup>e:
                bf 04 00 00 00
                                        mov
  40
                e8 a5 ff ff ff
                                        callq
                                                4005cd <even>
        Cleaning up the stack!
                b8
                   00 00 00 00
                                                $0x0,%eax
  4Ն
                                        mov
  40062
                                                %rbp
                                        pop
  40062e:
                                        retq
  40062f:
                                        nop
```



The C code:

```
1 #include <stdio.h>
3 void hello() {
       printf("Hello there!\n");
5
  void even(int a) {
       int i;
       for (i = 0 ; i < 10 ; i += 2) {
10
           printf("%d ", (a+i));
11
12
       printf("\n");
13
14 }
15
16 int main() {
17
       hello();
18
       even(4);
19
20
       return 0;
21 }
```