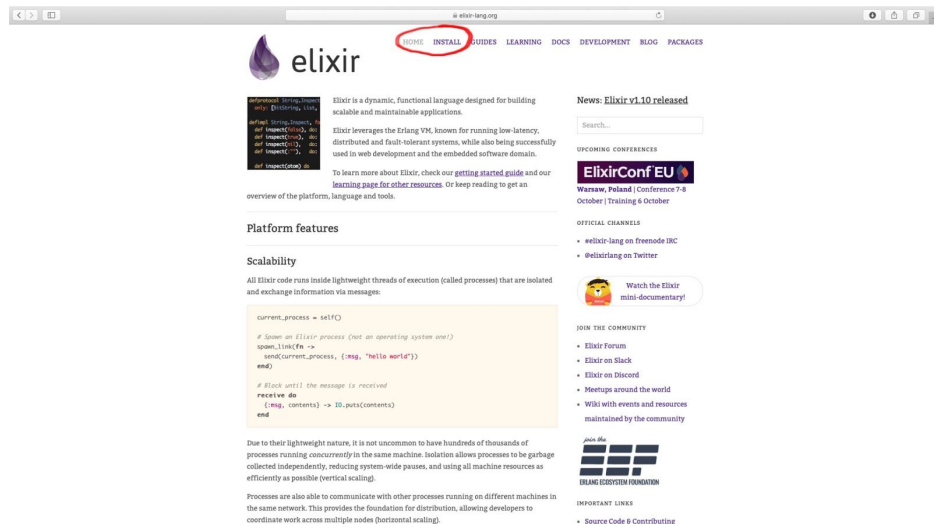


Manual del Pre-Compilador de C en Elixir

Equipo: RISC

- Antes de descargar, compilar y usar nuestro programa debes instalar Elixir dependiendo de tu sistema operativo desde su página oficial, a través del siguiente link: <https://elixir-lang.org>



- Una vez instalado Elixir puedes comprobar si lo tienes correctamente poniendo en tu terminal o shell, dependiendo del sistema operativo, poner el siguiente comando ">iex". Puedes observar como funciona el lenguaje con algunas operaciones, asignaciones etc.

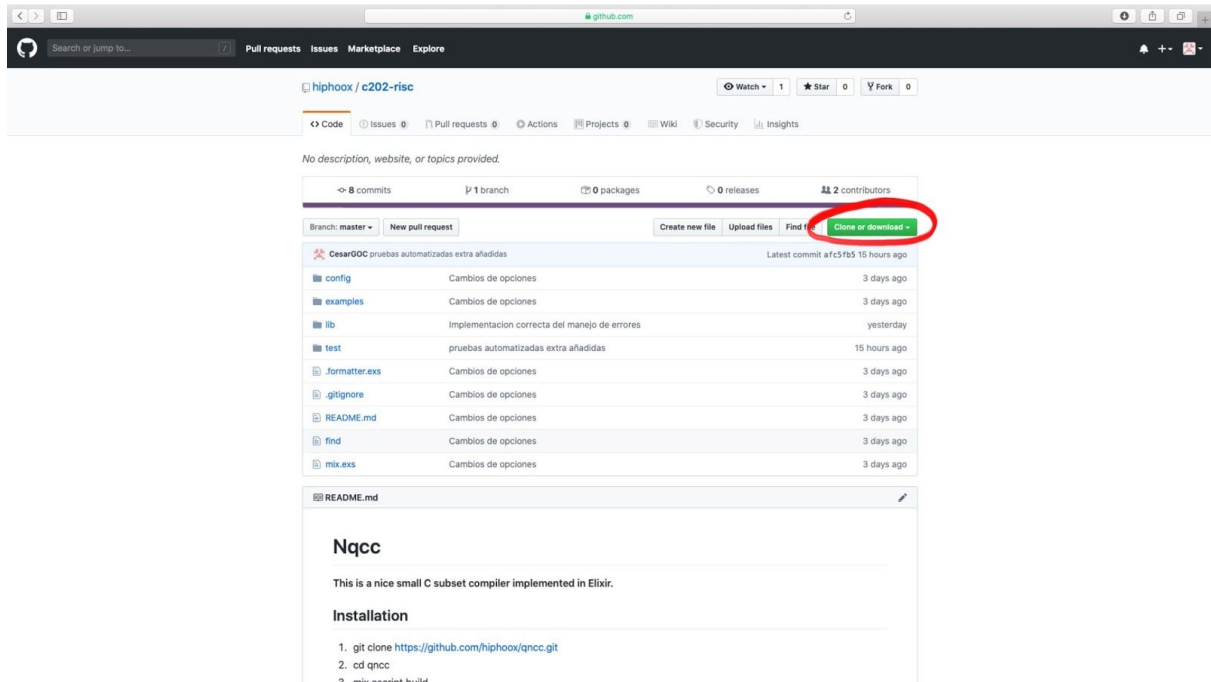
```
cesar — beam.smp -- -root /usr/local/Cellar/erlang/22.2.7/lib/erlang -prognam...
Last login: Wed Mar 18 02:07:56 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
iMac-de-Cesar:~ cesar$ iex
Erlang/OTP 22 [erts-10.6.4] [source] [64-bit] [smp:4:4] [ds:4:4:10] [async-threa
ds:1] [hipe] [dtrace]

Interactive Elixir (1.10.1) - press Ctrl+C to exit (type h() ENTER for help)
iex(1)> 4+5
9
iex(2)> suma = 100 + 123
223
iex(3)> cadena = "Cesar Alejandro"
"Cesar Alejandro"
iex(4)> lista = "*"
** (SyntaxError) iex:4: syntax error before: '*'

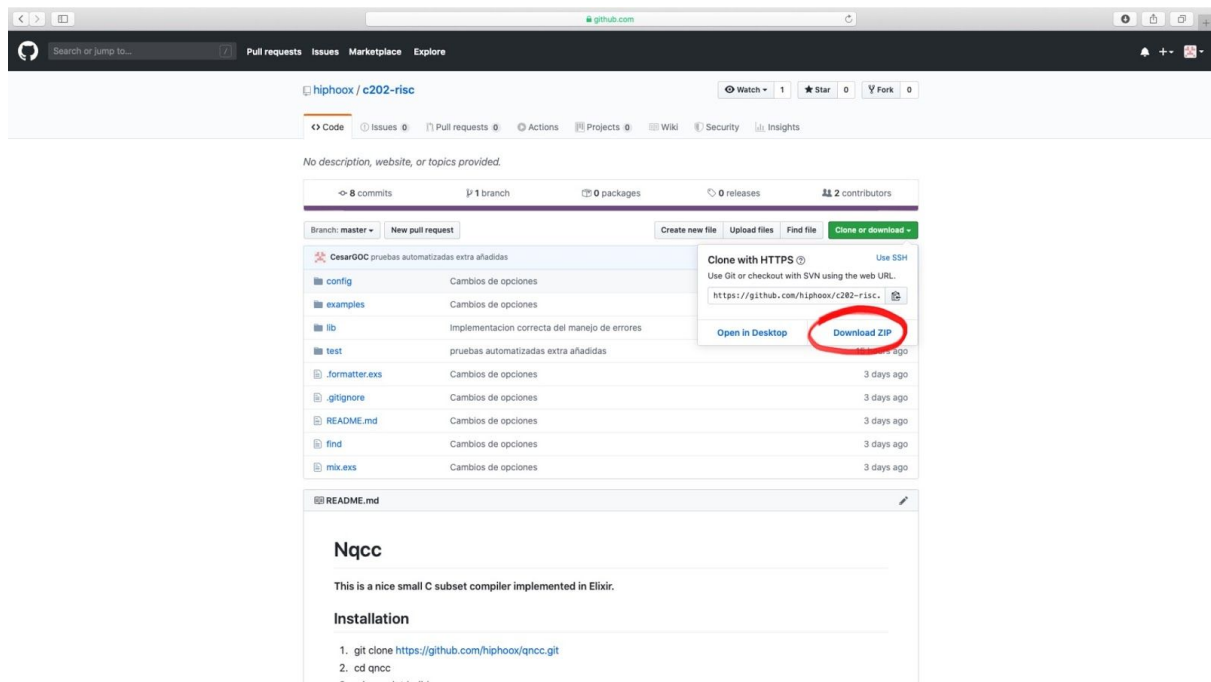
iex(4)> lista = ["1", "2", "a"]
["1", "2", "a"]
iex(5)>
```

- Para comenzar a usar nuestro compilador, primero tienes que descargarlo desde el siguiente repositorio de GitHub a través de este enlace:
<https://github.com/hiphoox/c202-risc>
- Ya en la pagina presiona el botón de color verde que dice “clone or

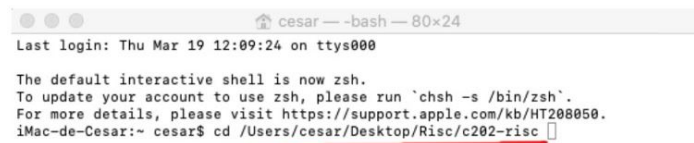


download”

- Presiona donde dice “Download ZIP”, esto descargara el proyecto completo. Solo faltaría descomprimirlo.



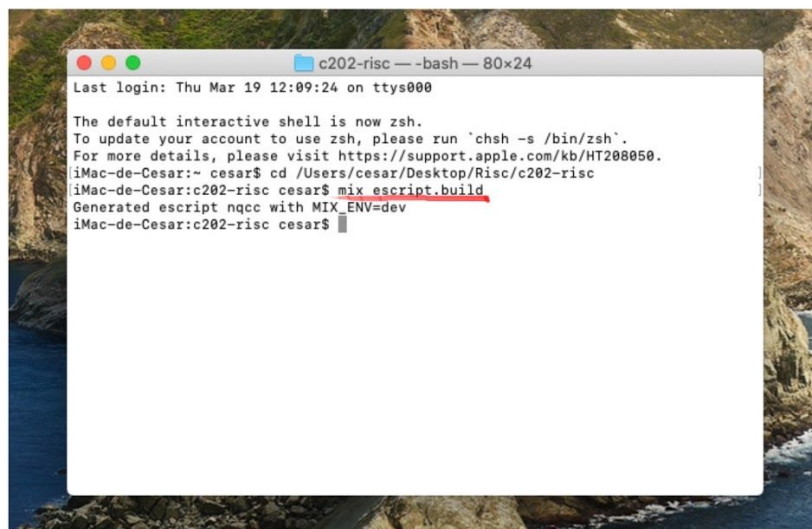
- Ahora abre tu terminal o shell y dirígete a la carpeta anteriormente descargada con el comando “cd” por ejemplo “>cd c202-risc-master” o como se muestra en la imagen a continuación.



```
cesar — -bash — 80x24
Last login: Thu Mar 19 12:09:24 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
iMac-de-Cesar:~ cesar$ cd /Users/cesar/Desktop/Risc/c202-risc
```

- Una vez estando en la carpeta del proyecto ejecuta el siguiente comando “>mix escript.build” para compilar el proyecto y generar la app del compilador llamado “nqcc”, el comando mix sirve para compilar programas de Elixir.



```
c202-risc — -bash — 80x24
Last login: Thu Mar 19 12:09:24 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
iMac-de-Cesar:~ cesar$ cd /Users/cesar/Desktop/Risc/c202-risc
iMac-de-Cesar:c202-risc cesar$ mix escript.build
Generated escript nqcc with MIX_ENV=dev
iMac-de-Cesar:c202-risc cesar$
```

- Ya una vez compilado, ahora vamos a probar el proyecto con un archivo “.c”, para esto, en la misma carpeta ejecuta el siguiente comando “./nqcc nombre_del_archivo.c” en el caso del proyecto puedes poner el siguiente comando “./nqcc examples/return_2.c”, ya que el proyecto cuenta con pequeño ejemplo.

- Una vez ejecutado el comando verás en tu terminal el archivo que se está compilando y sólo obtendras la salida del ensamblador como final como se muestra en la imagen.

```

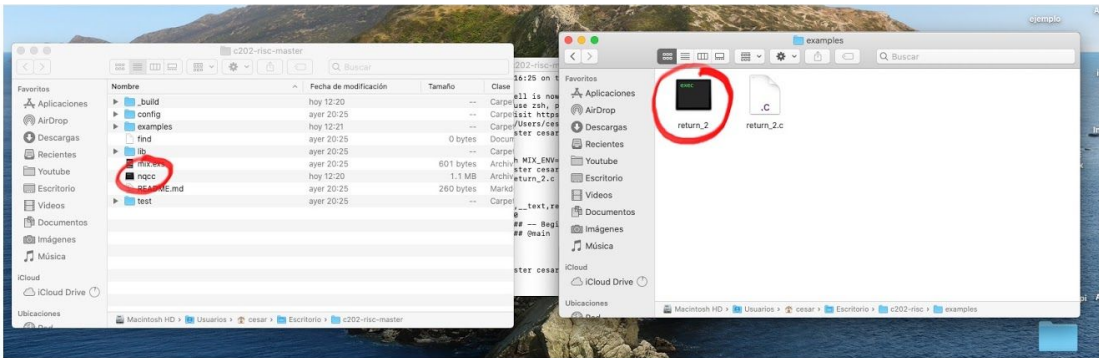
c202-risc-master -- -bash -- 80x24
Last login: Thu Mar 19 12:16:25 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT200050.
iMac-de-Cesar:~ cesar$ cd /Users/cesar/Desktop/c202-risc-master
iMac-de-Cesar:c202-risc-master cesar$ mix escript.build
Compiling 8 files (.ex)
Generated nqcc app
Generated escript nqcc with MIX_ENV=dev
iMac-de-Cesar:c202-risc-master cesar$ ./nqcc examples/return_2.c
Compiling file: examples/return_2.c

Code Generator output:
.section      __TEXT,__text,regular,pure_instructions
.p2align      4, 0x90
.globl _main
_main:
    movl     $2, %eax
    ret

iMac-de-Cesar:c202-risc-master cesar$
  
```

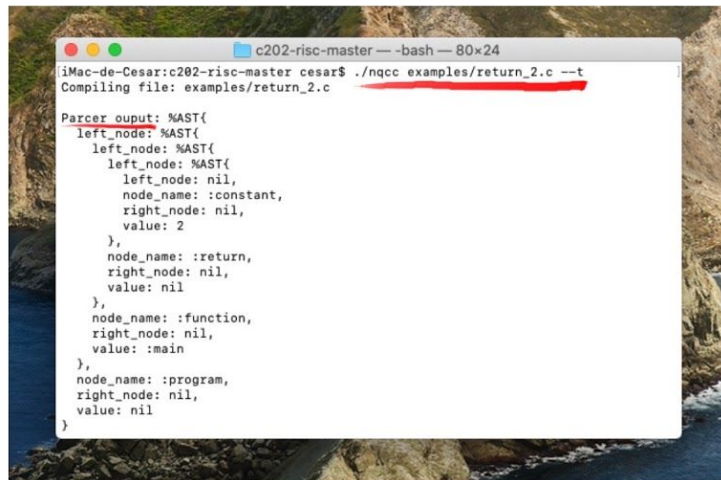
- Puedes dirigirte a tu carpeta del proyecto y veras que se han creado los dos ejecutables el de nuestro proyecto llamado “nqcc” como del ejemplo anteriormente utilizado en al carpeta examples llamado “return_2”.



- Si deseas ver mas a fondo las salidas de nuestro compilador o ayuda puedes ejecutar algunos de los siguientes comandos.

Bandera	función
--h	Muestra un menú de ayuda con toda la información contenida en esta tabla
--a	Muestra en la terminal el código ensamblador
--t	Imprimirá el árbol AST
--s	Mostrará la salida del sanitizer
--l	Mostrará la salida del lexer

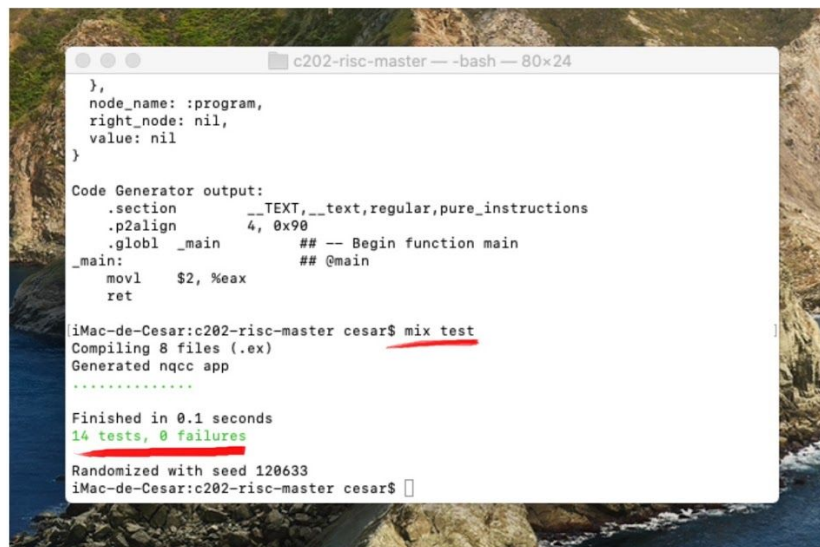
- Por ejemplo ejecutamos --t donde obtenemos el árbol generado.



```
iMac-de-Cesar:c202-risc-master cesar$ ./nqcc examples/return_2.c --t
Compiling file: examples/return_2.c

Parcer ouput: %AST(
  left_node: %AST(
    left_node: %AST(
      left_node: %AST(
        left_node: nil,
        node_name: :constant,
        right_node: nil,
        value: 2
      ),
      node_name: :return,
      right_node: nil,
      value: nil
    ),
    node_name: :function,
    right_node: nil,
    value: :main
  ),
  node_name: :program,
  right_node: nil,
  value: nil
)
```

- Por último si quieres realizar las pruebas automatizadas solo tienes que poner el siguiente comando “mix test” y se ejecutarán con éxito todas nuestras pruebas puestas en el proyecto.



```
},
node_name: :program,
right_node: nil,
value: nil
}

Code Generator output:
.section      __TEXT,__text,regular,pure_instructions
.p2align     4, 0x90
.globl _main      ## -- Begin function main
_main:
    movl     $2, %eax
    ret

iMac-de-Cesar:c202-risc-master cesar$ mix test
Compiling 8 files (.ex)
Generated nqcc app
.....

Finished in 0.1 seconds
14 tests, 0 failures
Randomized with seed 120633
iMac-de-Cesar:c202-risc-master cesar$
```