User manual: Team "Gold Compilers"

Contents.

Objective.

The user will learn how to download the project and how to learn to use it.

Introduction.

The following writing represents a manual for the use of the Foxy compiler. We show the installation requirements, the steps to follow for the installation, as well as the commands necessary to run and test the Foxy compiler, it has been designed so that a user with basic knowledge of Github and Ubuntu can easily execute it without any problem.

User manual.

Installation Requirements.

- Install git bash.
- Install elixir ((You will have to see which version you have)).

Installation.

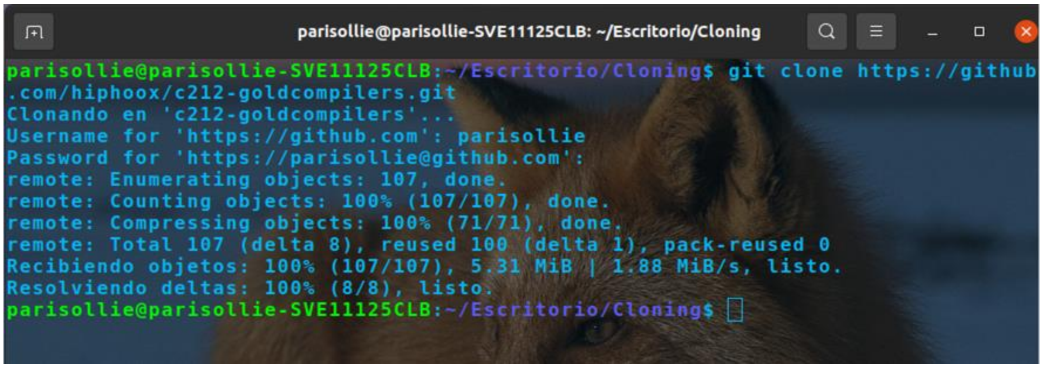
There are two ways to do this step

Using git bash.

Open the Git Bash or git terminal

Type the following command:

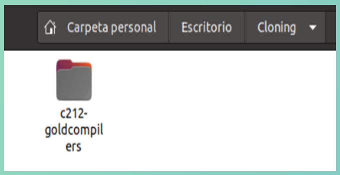
git clone https://github.com/hiphoox/c212-goldcompilers.git



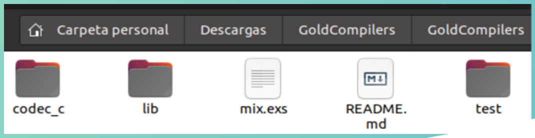
parisollie@parisollie-SVE11125CLB: ~/Escritorio/Cloning

```
parisollie@parisollie-SVE11125CLB:~/Escritorio/Cloning$ git clone https://github
.com/hiphoox/c212-goldcompilers.git
Clonando en 'c212-goldcompilers'...
Username for 'https://github.com': parisollie
Password for 'https://parisollie@github.com':
remote: Enumerating objects: 107, done.
remote: Counting objects: 100% (107/107), done.
remote: Compressing objects: 100% (71/71), done.
remote: Total 107 (delta 8), reused 100 (delta 1), pack-reused 0
Recibiendo objetos: 100% (107/107), 5.31 MiB | 1.88 MiB/s, listo.
Resolviendo deltas: 100% (8/8), listo.
parisollie@parisollie-SVE11125CLB:~/Escritorio/Cloning$ 
```

Once all this is done, a folder with the project will be generated:



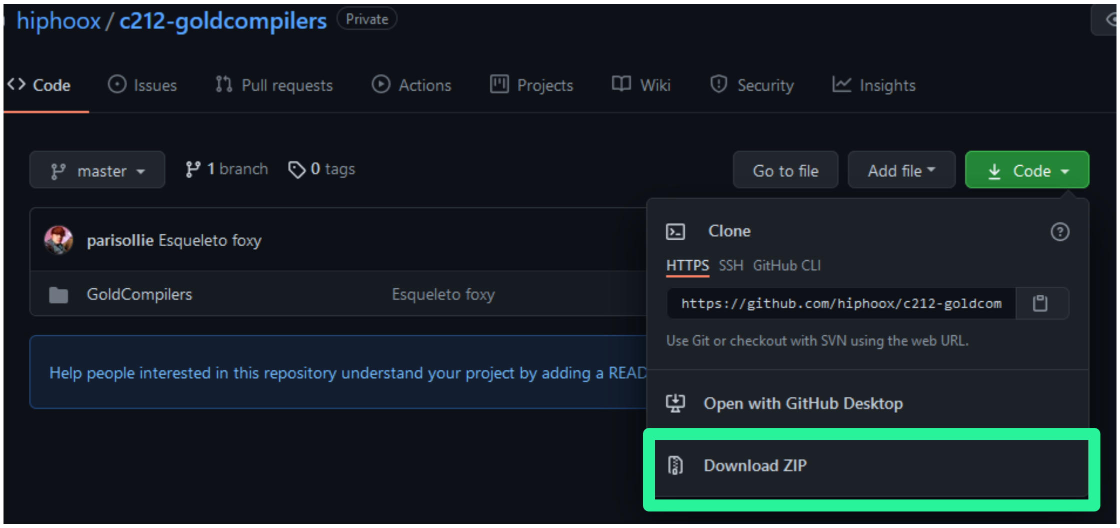
The project will contain:



➢ Download from github.

You should go to the following link:
https://github.com/hiphoox/c212-goldcompilers

Once inside the page download it this way:



https://github.com/hiphoox/c212-goldcom

Use Git or checkout with SVN using the web URL.

Help people interested in this repository understand your project by adding a READ

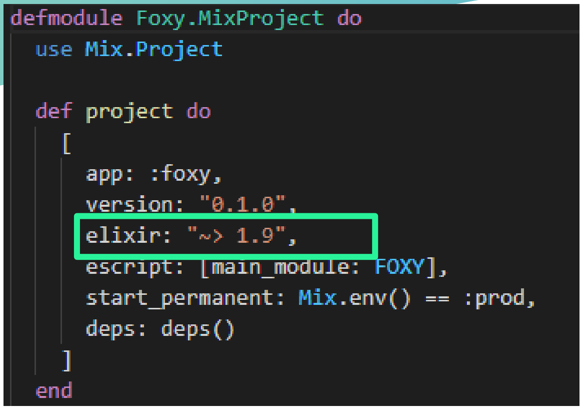
Execution in Ubuntu.

It goes to the Foxy compiler and opens a terminal there

Commands to start running.

First at all:

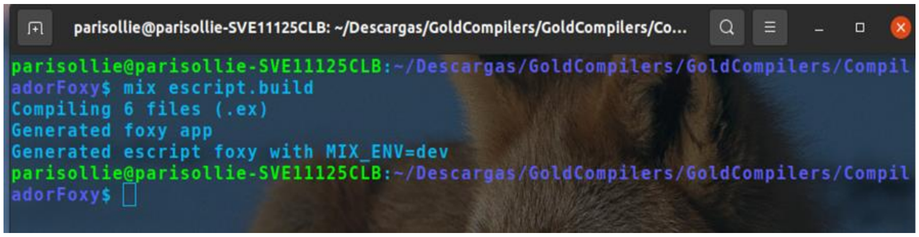
- Make sure you have updated what you are going to occupy.
- Make sure you have the corresponding elixir version with this project in this case we have the version: 1.9(**You can this information in the mix.exs of the project**)



```
defmodule Foxy.MixProject do
  use Mix.Project

  def project do
    [
      app: :foxy,
      version: "0.1.0",
      elixir: "~> 1.9",
      escript: [main_module: FOXY],
      start_permanent: Mix.env() == :prod,
      deps: deps()
    ]
  end
```

We write the following command to generate an executable:

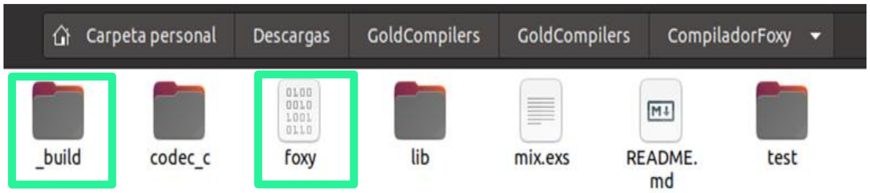
mix escript.build



parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

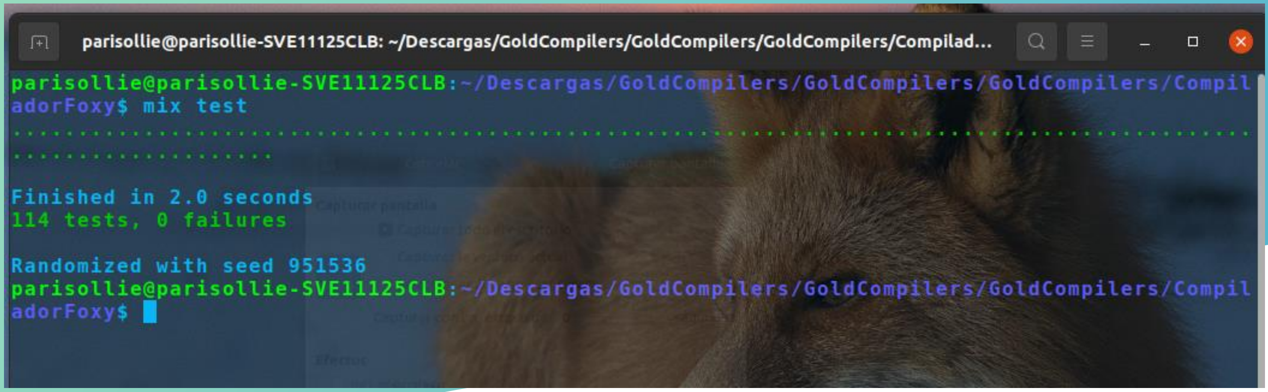
```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ mix escript.build
Compiling 6 files (.ex)
Generated foxy app
Generated escript foxy with MIX_ENV=dev
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$
```

If we look inside the project, the following will be generated:



We do a mix test to check that there are no errors:

mix test


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compilad...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compil
adorFoxy$ mix test
..............................................................................................
....................

Finished in 2.0 seconds
114 tests, 0 failures

Randomized with seed 951536
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

Before starting to run our compiler app you will have to make sure that you have these commands in your terminal:

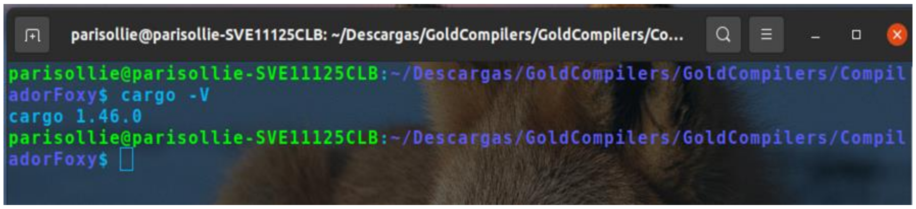
cargo -V

We install both:

Rust & Cargo

And last all this

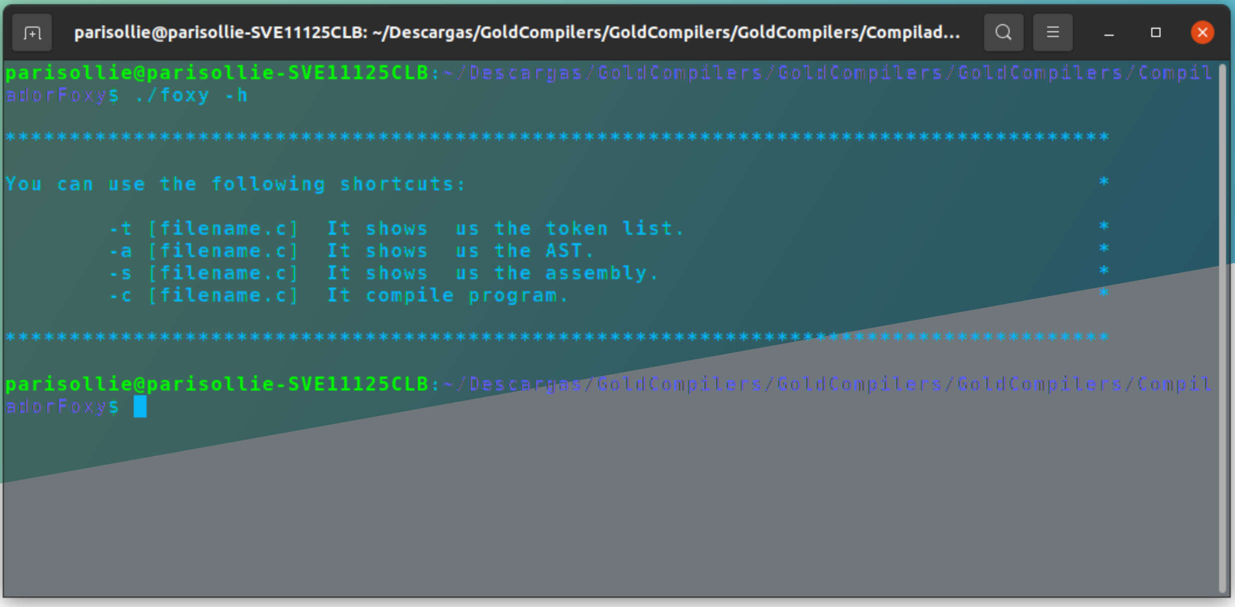
sudo mix deps.compile --all


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ cargo -V
cargo 1.46.0
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

We see the options that our compiler has:

./ foxy -h

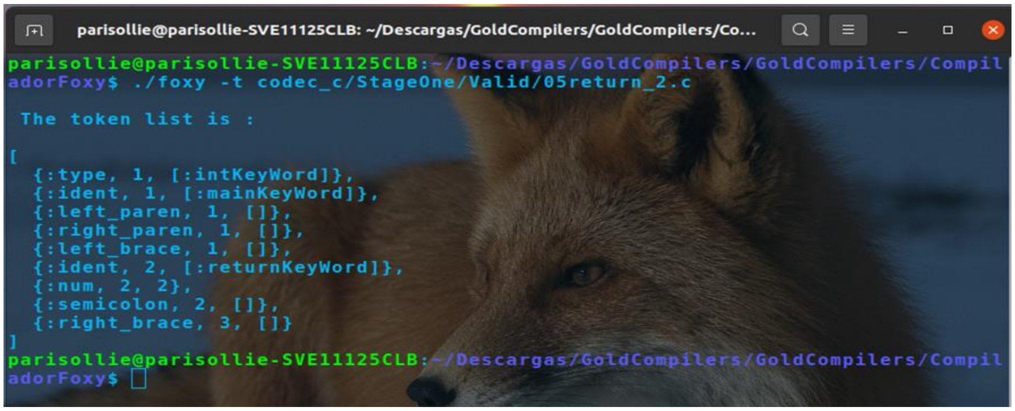

parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compilad...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compil
adorFoxy$ ./foxy -h

**************************************************************************************

You can use the following shortcuts:                                                 *

        -t [filename.c]  It shows  us the token list.                                *
        -a [filename.c]  It shows  us the AST.                                       *
        -s [filename.c]  It shows  us the assembly.                                  *
        -c [filename.c]  It compile program.                                         *

**************************************************************************************

parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

This example will show you how to use the Foxy compiler, valid and invalid examples will be used:

It shows us the token list:

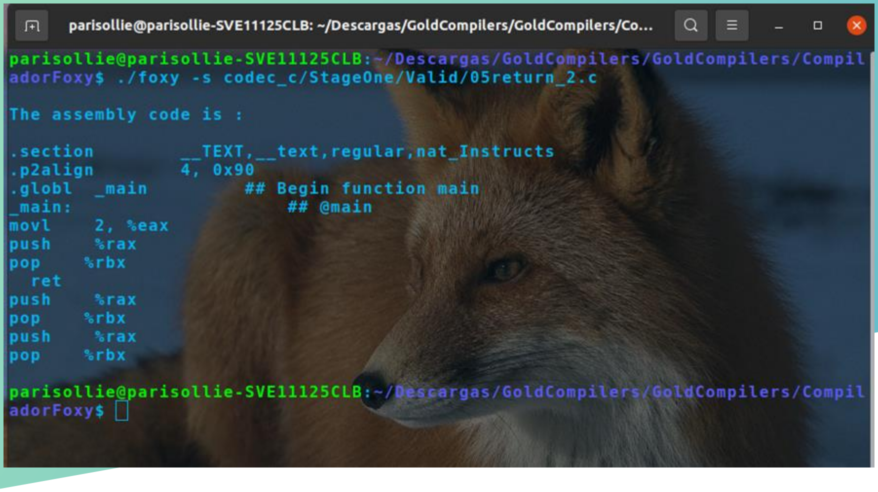
./foxy -t codec_c/StageOne/Valid/05return_2.c


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ ./foxy -t codec_c/StageOne/Valid/05return_2.c

 The token list is :

[
  {:type, 1, [:intKeyWord]},
  {:ident, 1, [:mainKeyWord]},
  {:left_paren, 1, []},
  {:right_paren, 1, []},
  {:left_brace, 1, []},
  {:ident, 2, [:returnKeyWord]},
  {:num, 2, 2},
  {:semicolon, 2, []},
  {:right_brace, 3, []}
]
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

It shows us the assembly:

./foxy -s codec_c/StageOne/Valid/05return_2.c

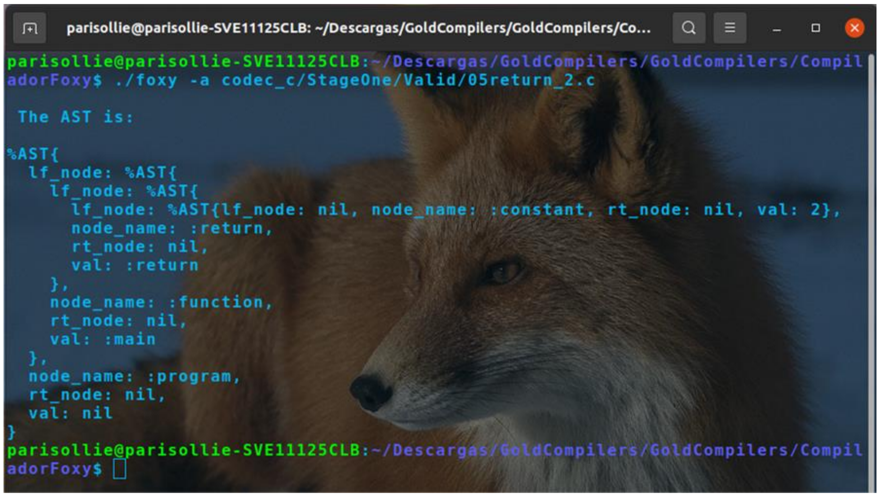


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ ./foxy -s codec_c/StageOne/Valid/05return_2.c

The assembly code is :

.section        __TEXT,__text,regular,nat_Instructs
.p2align        4, 0x90
.globl  _main         ## Begin function main
_main:                    ## @main
movl     2, %eax
push     %rax
pop     %rbx
  ret
push     %rax
pop     %rbx
push     %rax
pop     %rbx

parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

It shows us the AST:

./foxy -a codec_c/StageOne/Valid/05return_2.c

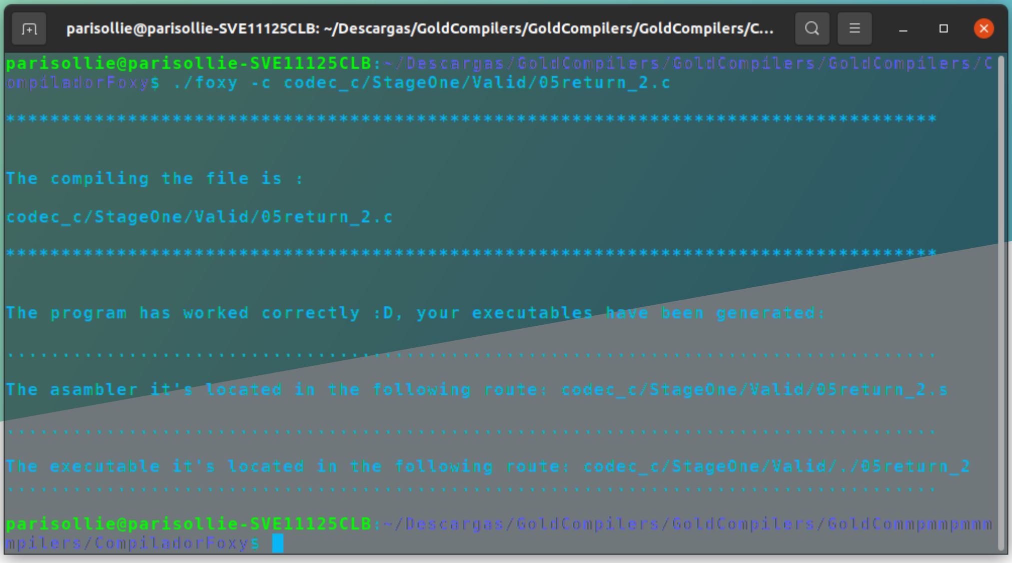


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ ./foxy -a codec_c/StageOne/Valid/05return_2.c

 The AST is:

%AST{
  lf_node: %AST{
    lf_node: %AST{
      lf_node: %AST{lf_node: nil, node_name: :constant, rt_node: nil, val: 2},
      node_name: :return,
      rt_node: nil,
      val: :return
    },
    node_name: :function,
    rt_node: nil,
    val: :main
  },
  node_name: :program,
  rt_node: nil,
  val: nil
}
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

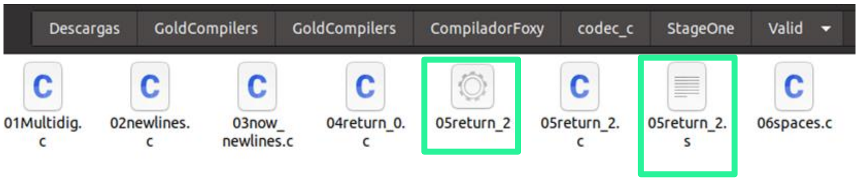
It compiles program:

./foxy -c codec_c/StageOne/Valid/05return_2.c


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/C...

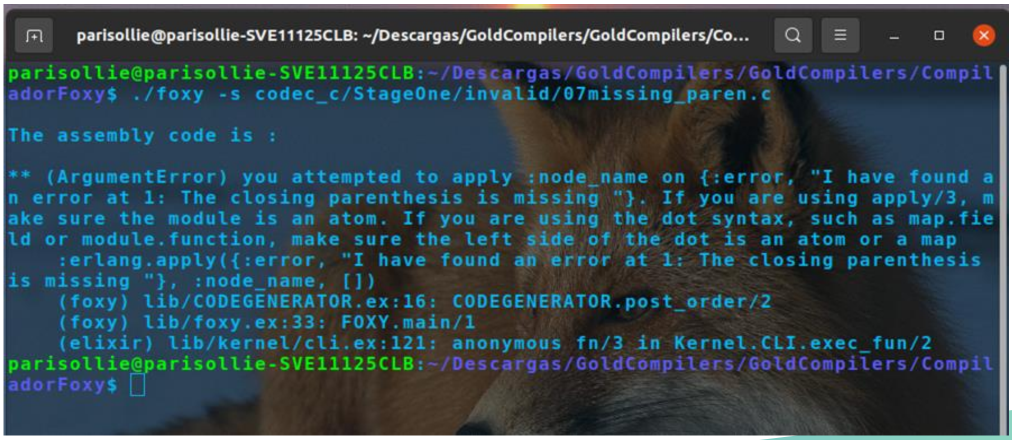
```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/C
ompiladorFoxy$ ./foxy -c codec_c/StageOne/Valid/05return_2.c

************************************************************************************


The compiling the file is :

codec_c/StageOne/Valid/05return_2.c

************************************************************************************


The program has worked correctly :D, your executables have been generated:

....................................................................................

The asambler it's located in the following route: codec_c/StageOne/Valid/05return_2.s

....................................................................................

The executable it's located in the following route: codec_c/StageOne/Valid/./05return_2
....................................................................................

parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/GoldCommpmmpmmm
mpilers/CompiladorFoxy$ 
```

It’s created inside the folder:



In case of error, our compiler shows the error:

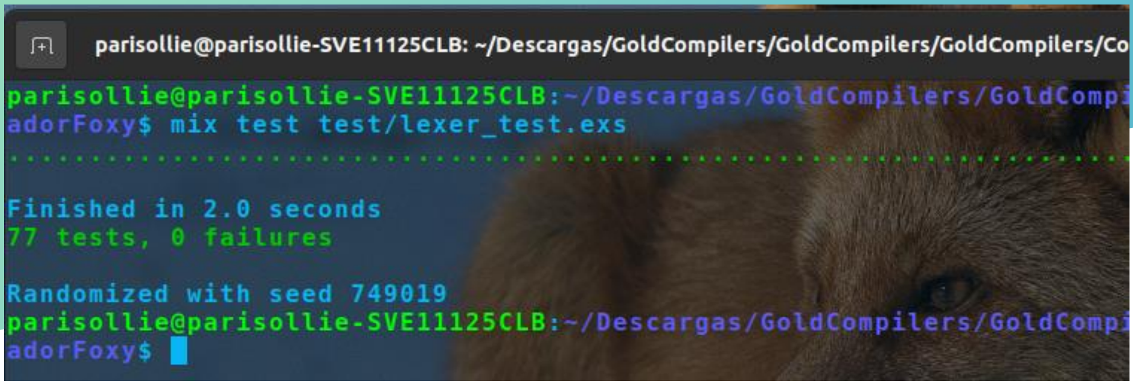
./foxy -s codec_c/StageOne/invalid/07missing_paren.c


parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/Co...

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ ./foxy -s codec_c/StageOne/invalid/07missing_paren.c

The assembly code is :

** (ArgumentError) you attempted to apply :node_name on {:error, "I have found a
n error at 1: The closing parenthesis is missing "}. If you are using apply/3, m
ake sure the module is an atom. If you are using the dot syntax, such as map.fie
ld or module.function, make sure the left side of the dot is an atom or a map
    :erlang.apply({:error, "I have found an error at 1: The closing parenthesis 
is missing "}, :node_name, [])
    (foxy) lib/CODEGENERATOR.ex:16: CODEGENERATOR.post_order/2
    (foxy) lib/foxy.ex:33: FOXY.main/1
    (elixir) lib/kernel/cli.ex:121: anonymous fn/3 in Kernel.CLI.exec_fun/2
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompilers/Compil
adorFoxy$ 
```

With this command you can do the tests of each test, you just must change the termination to do it to each one

mix test test/lexer_test.exs



parisollie@parisollie-SVE11125CLB: ~/Descargas/GoldCompilers/GoldCompilers/GoldCompilers/Co

```
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompi
adorFoxy$ mix test test/lexer_test.exs
.............................................................................

Finished in 2.0 seconds
77 tests, 0 failures

Randomized with seed 749019
parisollie@parisollie-SVE11125CLB:~/Descargas/GoldCompilers/GoldCompi
adorFoxy$
```

Historic:

Author:	Description:	Version:	Date:
Felix Flores Paul Jaime	First Version	1.0	26-07-21
Felix Flores Paul Jaime	Second Version	2.0	07-08-21
Felix Flores Paul Jaime	Third Version	3.0	13-08-21