POLYNOM

🎯 A library to manipulate polynomials over finite fields.

🔐 Design for cryptographic operations.

🚀 Explore this project on GitHub!

https://github.com/groumage/PolyNom/

TECHNICAL OVERVIEW

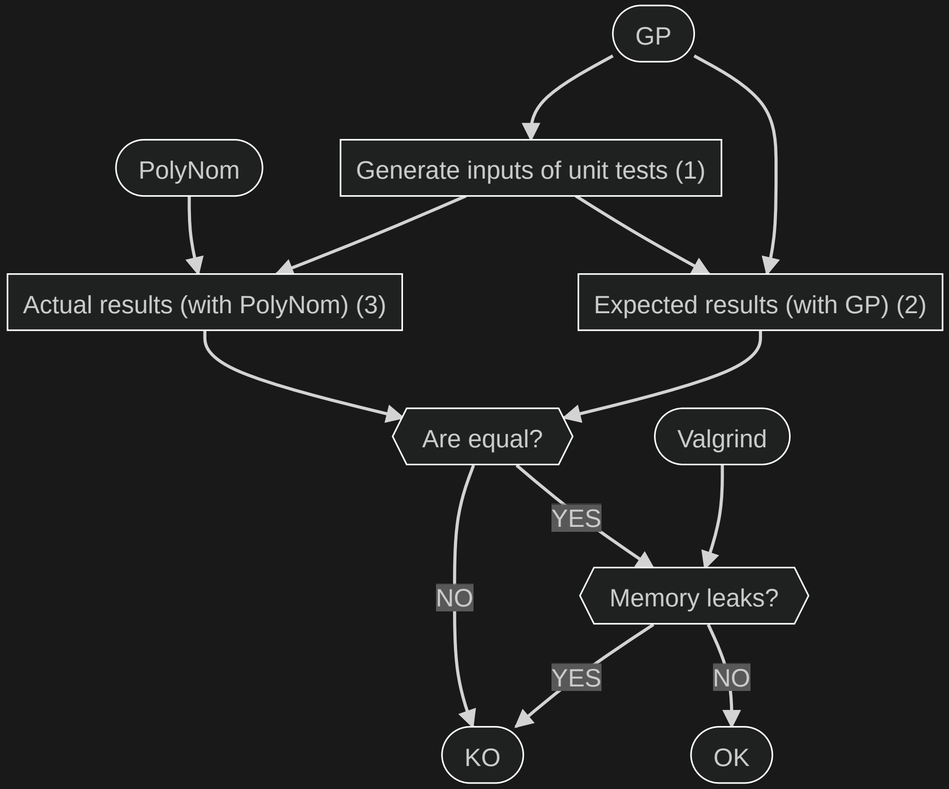
💻 Technology stacks:

- C language
- GMP library
- GP[1]
- Valgrind
- Gcovr

[1] An interactive shell providing fast computation functions in number theory.

TEST-DRIVEN DEVELOPMENT

🔎 GP generates random inputs and expected outputs that PolyNom's functions should return.


GP
PolyNom
Generate inputs of unit tests (1)
Actual results (with PolyNom) (3)
Expected results (with GP) (2)
Are equal?
Valgrind
YES
NO
Memory leaks?
YES
NO
KO
OK

CODE COVERAGE

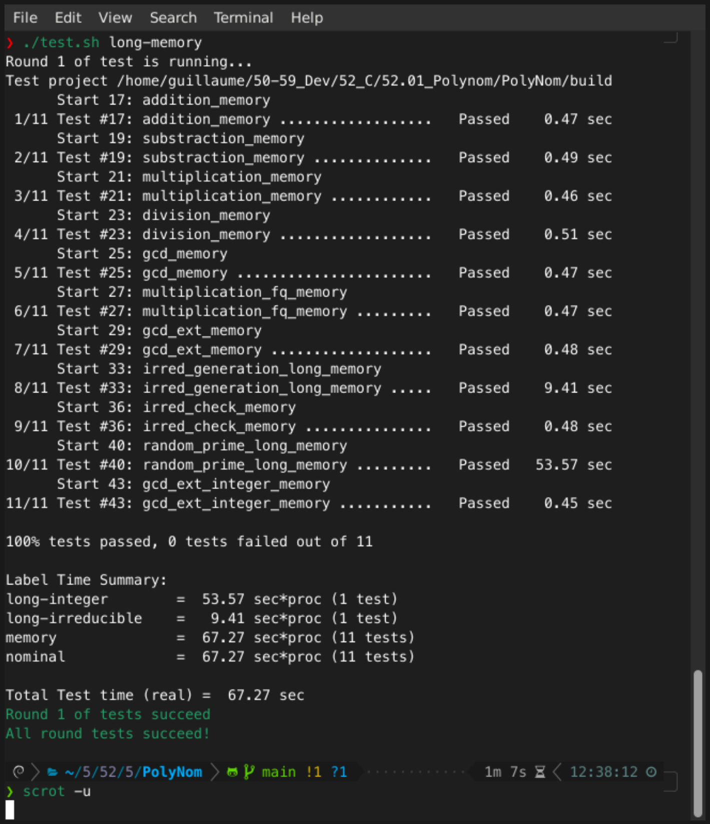
🔦 Code coverage of PolyNom is performed Gcovr.

⚙ 76% of lines of codes, 79 % of functions and 82 % of branches are covered.

Code coverage report can be found here (open it in your favorite browser).

DEMONSTRATION

📷 Unit tests execution of usual operations $(512 \leq deg(P_{\text{inputs}}) \leq 2048)$ along with Valgrind.



```
❯ ./test.sh long-memory
Round 1 of test is running...
Test project /home/guillaume/50-59_Dev/52_C/52.01_Polynom/PolyNom/build
      Start 17: addition_memory
 1/11 Test #17: addition_memory ..................   Passed    0.47 sec
      Start 19: substraction_memory
 2/11 Test #19: substraction_memory ..............   Passed    0.49 sec
      Start 21: multiplication_memory
 3/11 Test #21: multiplication_memory ............   Passed    0.46 sec
      Start 23: division_memory
 4/11 Test #23: division_memory ..................   Passed    0.51 sec
      Start 25: gcd_memory
 5/11 Test #25: gcd_memory .......................   Passed    0.47 sec
      Start 27: multiplication_fq_memory
 6/11 Test #27: multiplication_fq_memory .........   Passed    0.47 sec
      Start 29: gcd_ext_memory
 7/11 Test #29: gcd_ext_memory ...................   Passed    0.48 sec
      Start 33: irred_generation_long_memory
 8/11 Test #33: irred_generation_long_memory .....   Passed    9.41 sec
      Start 36: irred_check_memory
 9/11 Test #36: irred_check_memory ...............   Passed    0.48 sec
      Start 40: random_prime_long_memory
10/11 Test #40: random_prime_long_memory .........   Passed   53.57 sec
      Start 43: gcd_ext_integer_memory
11/11 Test #43: gcd_ext_integer_memory ...........   Passed    0.45 sec

100% tests passed, 0 tests failed out of 11

Label Time Summary:
long-integer        =  53.57 sec*proc (1 test)
long-irreducible    =   9.41 sec*proc (1 test)
memory              =  67.27 sec*proc (11 tests)
nominal             =  67.27 sec*proc (11 tests)

Total Test time (real) =  67.27 sec
Round 1 of tests succeed
All round tests succeed!

 ❯ ~/5/52/5/PolyNom ❯  main !1 ?1 ············ 1m 7s ⧗ ❮ 12:38:12
❯ scrot -u
```

CONCLUSION

💡 PolyNom manipulates arbitrary long polynomials over finite fields.

⚙ PolyNom's functions are tested with Valgrind.

🔦 A code coverage of PolyNom is performed.

🚀 Checkout PolyNom and its nice documentation!

BONUS: TECHNICAL OVERVIEW

```c
typedef struct fp_poly_t
{
    mpz_t index_coeff;
    list_t *coeffs;
} fp_poly_t;
```

$$P(x) = 2 + x^2 + x^3$$

$$P_{\text{index coeff}} = 1011_2 = 11_{10}$$

$$P_{\text{coeffs}} = \{2\} \rightarrow \{1\} \rightarrow \{1\} \rightarrow NULL$$