Exercício 2 - NTT_CRT

April 2, 2024

1 Estruturas Criptográficas - Criptografia e Segurança da Informação

Grupo 03

(PG54177) Ricardo Alves Oliveira

(PG54236) Simão Oliveira Alvim Barroso

1.1 TP2 - Exercício 2

2. Uma das aplicações mais importantes do teorema chinês dos restos (CRT) em criptografia é a transformada NTT "Number Theoretic Transform". Esta transformada é uma componente importantes de "standards" PQC como o Kyber e o Dilithium mas também de outros algoritmos submetidos ao concurso NIST PQC.
   A transformação NTT tem várias opções e aquela que está apresentada no +Capítulo 4: Problemas Difíceis usa o CRT. Neste problema pretende-se uma implementação Sagemath do NTT-CRT tal como é descrito nesse documento.

1.2 Resolução

Para resolver este exercício comecamos por instalar e importar o SageMath.

[ ]:
```
from sage.all import *
import random
```

A função find_q(n) tem como objetivo encontrar um número primo que satisfaça a condição q 1mod(2N), onde N pertence a [32, 64, 128, 256, 512, 1024, 2048].

[ ]:
```
def find_q(n):
    if not  n in [32,64,128,256,512,1024,2048]:
        raise ValueError("improper argument ",n)

    q = 1 + 2*n
    while True:
        if (q).is_prime():
            return q
        q += 2*n
```

De seguida implementamos a função NTT que recebe como argumentos:

- f: Polinómio de entrada
- N: Tamanho do polinómio
- xi: Raiz primitiva de N
- F: Campo finito onde serão realizadas as operações

Esta função devolve o polinómio transformado de acordo com a transformada NTT. Para tal foram implementadas as seguintes funções auxiliares:

- _expand_: Esta função recebe um polinómio e um tamanho N e devolve um polinómio de tamanho N com os coeficientes do polinómio de entrada e os restantes coeficientes a 0.
- _ntt_: Esta função recebe um polinómio de entrada e devolve o polinómio transformado de acordo com a transformada NTT.

Por fim, a implementação da transformada segue o algoritmo descrito no Capítulo 4.

[ ]:
```
def ntt(f,N,xi,F):
    def _expand_(f):
        u = f.list()
        return u + [0]*(N-len(u))

    def _ntt_(xi,N,f):
        if N==1:
            return f
        N_ = N/2 ; xi2 =  xi^2
        f0 = [f[2*i]   for i in range(N_)] ; f1 = [f[2*i+1] for i in range(N_)]
        ff0 = _ntt_(xi2,N_,f0) ; ff1 = _ntt_(xi2,N_,f1)

        s  = xi ; ff = [F(0) for i in range(N)]
        for i in range(N_):
            a = ff0[i] ; b = s*ff1[i]
            ff[i] = a + b ; ff[i + N_] = a - b
            s = s * xi2
        return ff

    return _ntt_(xi,N,_expand_(f))
```

Para o cálculo da inversa da transformada NTT foi implementada a função ntt_inv que ao receber o polinómio transformado, o tamanho do polinómio e o array de bases do crt, devolve o polinómio original.

[ ]:
```
def ntt_inv(ff,N,base):
    return sum([ff[i]*base[i] for i in range(N)])
```

A função auxiliar random_pol e responsável por gerar um polinómio aleatório a partir de um anel polinomial R.

[ ]:
```
def random_pol(R,args=None):
    return R.random_element(args)
```

Para testar a implementação precisamos agora de obter o N desejado e gerar o q correspondente, com recurso à função find_q(n) mencionada anteriormente.

[ ]:
```
N = int(input("Enter N: "))
q = find_q(N)
print("q = ",q)
```

```
q =  12289
```

Com as variáveis de entrada defenidas podemos agora definir:

- O campo finito F
- O anel polinomial R com base no campo finito F
- O gerador do anel de polinômios R, representado por w
- O polinómio g, utilizado para encontrar as raízes primitivas
- As raízes primitivas xi
- As raízes rs que são geradas segundo a forma `xi^(2i+1)`
- A base do CRT calculada através da função `crt_basis`

[ ]:
```
F = FiniteField(q)
R = PolynomialRing(F, name="w")
w = R.gen()
g = (w^N + 1)
xi = g.roots(multiplicities=False)[-1]
rs = [xi^(2*i+1) for i in range(N)]
base = crt_basis([(w - r) for r in rs])
```

1.3 Testes

Com as variáveis definidas podemos agora testar a implementação da transformada NTT e da sua inversa. Para tal, geramos um polinómio aleatório de tamanho entre 1 e N-1.

[ ]:
```
f = random_pol(R,N-random.randint(1,N-1))
print("f = ",end='')
for i in range(N-1):
    print(f"{f[i]}*w^{i} + ",end='')
print(f"{f[N-1]}*w^{N-1}")
```

```
f = 2985*w^0 + 10539*w^1 + 2837*w^2 + 728*w^3 + 917*w^4 + 3665*w^5 + 3724*w^6 +
4560*w^7 + 1125*w^8 + 939*w^9 + 5063*w^10 + 8343*w^11 + 9392*w^12 + 4593*w^13 +
8260*w^14 + 10329*w^15 + 3523*w^16 + 10935*w^17 + 5634*w^18 + 9333*w^19 +
9175*w^20 + 5411*w^21 + 7285*w^22 + 7498*w^23 + 1579*w^24 + 2717*w^25 +
11477*w^26 + 2852*w^27 + 763*w^28 + 3609*w^29 + 3373*w^30 + 11084*w^31 +
5907*w^32 + 12161*w^33 + 7957*w^34 + 8969*w^35 + 10227*w^36 + 5757*w^37 +
9378*w^38 + 11352*w^39 + 4113*w^40 + 136*w^41 + 3088*w^42 + 2100*w^43 +
12115*w^44 + 10175*w^45 + 9557*w^46 + 6705*w^47 + 8583*w^48 + 687*w^49 +
7063*w^50 + 1903*w^51 + 6148*w^52 + 5388*w^53 + 10156*w^54 + 11243*w^55 +
2336*w^56 + 5098*w^57 + 5700*w^58 + 12257*w^59 + 2567*w^60 + 2552*w^61 +
4659*w^62 + 9796*w^63 + 3425*w^64 + 5004*w^65 + 2999*w^66 + 7173*w^67 +
```

```
8152*w^68 + 4002*w^69 + 11856*w^70 + 2106*w^71 + 2163*w^72 + 9586*w^73 +
6891*w^74 + 11695*w^75 + 8203*w^76 + 5299*w^77 + 12069*w^78 + 1717*w^79 +
10452*w^80 + 3313*w^81 + 390*w^82 + 4366*w^83 + 9340*w^84 + 4100*w^85 +
10548*w^86 + 11650*w^87 + 12240*w^88 + 4500*w^89 + 7686*w^90 + 6662*w^91 +
4285*w^92 + 2214*w^93 + 6623*w^94 + 10891*w^95 + 2060*w^96 + 3939*w^97 +
7082*w^98 + 6050*w^99 + 7489*w^100 + 1408*w^101 + 5944*w^102 + 3884*w^103 +
6382*w^104 + 2190*w^105 + 7514*w^106 + 6597*w^107 + 4456*w^108 + 10319*w^109 +
2658*w^110 + 5888*w^111 + 11976*w^112 + 3821*w^113 + 8202*w^114 + 9360*w^115 +
407*w^116 + 9291*w^117 + 11926*w^118 + 3989*w^119 + 2698*w^120 + 7811*w^121 +
9379*w^122 + 3394*w^123 + 994*w^124 + 5612*w^125 + 9249*w^126 + 6500*w^127 +
11072*w^128 + 10336*w^129 + 2215*w^130 + 6116*w^131 + 553*w^132 + 9244*w^133 +
409*w^134 + 11480*w^135 + 7256*w^136 + 8045*w^137 + 6536*w^138 + 2888*w^139 +
10998*w^140 + 941*w^141 + 7242*w^142 + 4487*w^143 + 5784*w^144 + 8377*w^145 +
3942*w^146 + 754*w^147 + 3008*w^148 + 1867*w^149 + 3233*w^150 + 5996*w^151 +
1226*w^152 + 5468*w^153 + 10104*w^154 + 1161*w^155 + 6363*w^156 + 6727*w^157 +
8813*w^158 + 5186*w^159 + 3978*w^160 + 4337*w^161 + 2755*w^162 + 8006*w^163 +
2926*w^164 + 2271*w^165 + 6342*w^166 + 7007*w^167 + 11187*w^168 + 11595*w^169 +
8028*w^170 + 5804*w^171 + 9621*w^172 + 5308*w^173 + 6535*w^174 + 10773*w^175 +
6744*w^176 + 7929*w^177 + 5304*w^178 + 5696*w^179 + 3102*w^180 + 269*w^181 +
2722*w^182 + 3711*w^183 + 7045*w^184 + 1221*w^185 + 9551*w^186 + 10419*w^187 +
8256*w^188 + 9067*w^189 + 1455*w^190 + 7298*w^191 + 9502*w^192 + 6260*w^193 +
10406*w^194 + 5667*w^195 + 3704*w^196 + 9267*w^197 + 3734*w^198 + 6030*w^199 +
5819*w^200 + 2824*w^201 + 4199*w^202 + 11468*w^203 + 4607*w^204 + 3741*w^205 +
11522*w^206 + 4345*w^207 + 3224*w^208 + 5773*w^209 + 1610*w^210 + 7318*w^211 +
10181*w^212 + 11977*w^213 + 6434*w^214 + 10248*w^215 + 11977*w^216 + 8311*w^217
+ 826*w^218 + 6912*w^219 + 1338*w^220 + 4691*w^221 + 7398*w^222 + 3699*w^223 +
4406*w^224 + 407*w^225 + 7807*w^226 + 5209*w^227 + 9847*w^228 + 5021*w^229 +
585*w^230 + 6923*w^231 + 2129*w^232 + 1585*w^233 + 11834*w^234 + 9840*w^235 +
10612*w^236 + 1943*w^237 + 7960*w^238 + 10607*w^239 + 7989*w^240 + 10357*w^241 +
6581*w^242 + 2659*w^243 + 1854*w^244 + 6572*w^245 + 8167*w^246 + 7331*w^247 +
7453*w^248 + 2607*w^249 + 8376*w^250 + 8360*w^251 + 6865*w^252 + 129*w^253 +
1493*w^254 + 9754*w^255 + 10206*w^256 + 5369*w^257 + 12197*w^258 + 11081*w^259 +
11674*w^260 + 6749*w^261 + 4580*w^262 + 5829*w^263 + 1978*w^264 + 1972*w^265 +
5392*w^266 + 10141*w^267 + 748*w^268 + 4526*w^269 + 8358*w^270 + 11533*w^271 +
7096*w^272 + 6379*w^273 + 7118*w^274 + 4892*w^275 + 7370*w^276 + 11447*w^277 +
2617*w^278 + 2632*w^279 + 194*w^280 + 4912*w^281 + 5580*w^282 + 2058*w^283 +
2948*w^284 + 6240*w^285 + 121*w^286 + 7817*w^287 + 10363*w^288 + 4271*w^289 +
313*w^290 + 9960*w^291 + 5300*w^292 + 8821*w^293 + 3342*w^294 + 2254*w^295 +
181*w^296 + 5901*w^297 + 3639*w^298 + 3923*w^299 + 1428*w^300 + 2301*w^301 +
5009*w^302 + 11374*w^303 + 1419*w^304 + 8664*w^305 + 71*w^306 + 4446*w^307 +
3742*w^308 + 11409*w^309 + 4858*w^310 + 9377*w^311 + 7573*w^312 + 2645*w^313 +
4000*w^314 + 8607*w^315 + 12166*w^316 + 11657*w^317 + 6908*w^318 + 5337*w^319 +
8080*w^320 + 155*w^321 + 6476*w^322 + 5702*w^323 + 10051*w^324 + 3452*w^325 +
12070*w^326 + 8021*w^327 + 1691*w^328 + 3076*w^329 + 848*w^330 + 9437*w^331 +
6638*w^332 + 10935*w^333 + 3583*w^334 + 9298*w^335 + 6011*w^336 + 7029*w^337 +
4812*w^338 + 10781*w^339 + 7666*w^340 + 5259*w^341 + 2959*w^342 + 8098*w^343 +
6005*w^344 + 6327*w^345 + 10767*w^346 + 10692*w^347 + 3442*w^348 + 8245*w^349 +
3785*w^350 + 3712*w^351 + 9514*w^352 + 6881*w^353 + 4037*w^354 + 2539*w^355 +
```

```
6394*w^356 + 10607*w^357 + 3820*w^358 + 844*w^359 + 3648*w^360 + 3351*w^361 +
7022*w^362 + 42*w^363 + 1265*w^364 + 1689*w^365 + 5785*w^366 + 7752*w^367 +
8393*w^368 + 7975*w^369 + 1744*w^370 + 2967*w^371 + 5963*w^372 + 4351*w^373 +
11056*w^374 + 9734*w^375 + 4076*w^376 + 6647*w^377 + 685*w^378 + 1653*w^379 +
5958*w^380 + 7967*w^381 + 4696*w^382 + 6403*w^383 + 314*w^384 + 2079*w^385 +
5225*w^386 + 9492*w^387 + 3075*w^388 + 3551*w^389 + 657*w^390 + 5977*w^391 +
2085*w^392 + 10981*w^393 + 7277*w^394 + 9289*w^395 + 3677*w^396 + 12178*w^397 +
8820*w^398 + 4572*w^399 + 3098*w^400 + 181*w^401 + 5635*w^402 + 5859*w^403 +
6672*w^404 + 9066*w^405 + 176*w^406 + 1214*w^407 + 11903*w^408 + 1210*w^409 +
9510*w^410 + 2491*w^411 + 4498*w^412 + 11490*w^413 + 9366*w^414 + 812*w^415 +
4614*w^416 + 3500*w^417 + 3401*w^418 + 6724*w^419 + 2368*w^420 + 6180*w^421 +
6812*w^422 + 9289*w^423 + 6914*w^424 + 3651*w^425 + 3412*w^426 + 1746*w^427 +
7722*w^428 + 6895*w^429 + 11805*w^430 + 3581*w^431 + 11354*w^432 + 9947*w^433 +
9819*w^434 + 4537*w^435 + 4757*w^436 + 2398*w^437 + 10815*w^438 + 3890*w^439 +
7443*w^440 + 4150*w^441 + 3689*w^442 + 2816*w^443 + 1424*w^444 + 1204*w^445 +
10190*w^446 + 1974*w^447 + 4821*w^448 + 12116*w^449 + 4058*w^450 + 4637*w^451 +
2591*w^452 + 4398*w^453 + 5538*w^454 + 771*w^455 + 4143*w^456 + 3172*w^457 +
2765*w^458 + 2714*w^459 + 10084*w^460 + 10252*w^461 + 11462*w^462 + 10407*w^463
+ 3497*w^464 + 186*w^465 + 541*w^466 + 4112*w^467 + 2803*w^468 + 3196*w^469 +
4253*w^470 + 1350*w^471 + 9208*w^472 + 4105*w^473 + 9804*w^474 + 2092*w^475 +
7173*w^476 + 8295*w^477 + 7816*w^478 + 43*w^479 + 7778*w^480 + 6845*w^481 +
3076*w^482 + 905*w^483 + 560*w^484 + 6931*w^485 + 6180*w^486 + 4954*w^487 +
6616*w^488 + 8435*w^489 + 8886*w^490 + 8108*w^491 + 5074*w^492 + 5564*w^493 +
33*w^494 + 3170*w^495 + 2454*w^496 + 4144*w^497 + 7431*w^498 + 11957*w^499 +
8269*w^500 + 144*w^501 + 2041*w^502 + 11478*w^503 + 9758*w^504 + 2276*w^505 +
437*w^506 + 3933*w^507 + 11731*w^508 + 9564*w^509 + 3023*w^510 + 1758*w^511 +
690*w^512 + 4019*w^513 + 9946*w^514 + 12282*w^515 + 3771*w^516 + 6895*w^517 +
962*w^518 + 10369*w^519 + 10314*w^520 + 6808*w^521 + 2085*w^522 + 6536*w^523 +
10108*w^524 + 3989*w^525 + 4273*w^526 + 3085*w^527 + 8282*w^528 + 970*w^529 +
3186*w^530 + 6117*w^531 + 656*w^532 + 12224*w^533 + 11339*w^534 + 6123*w^535 +
4563*w^536 + 935*w^537 + 2551*w^538 + 3651*w^539 + 3028*w^540 + 11836*w^541 +
4894*w^542 + 998*w^543 + 2622*w^544 + 5235*w^545 + 8644*w^546 + 9806*w^547 +
2043*w^548 + 4636*w^549 + 221*w^550 + 5020*w^551 + 9176*w^552 + 11730*w^553 +
204*w^554 + 9141*w^555 + 8148*w^556 + 1547*w^557 + 4111*w^558 + 8644*w^559 +
7*w^560 + 1977*w^561 + 11618*w^562 + 3055*w^563 + 9017*w^564 + 6932*w^565 +
490*w^566 + 10854*w^567 + 7494*w^568 + 1898*w^569 + 6409*w^570 + 11656*w^571 +
8462*w^572 + 11716*w^573 + 6537*w^574 + 6175*w^575 + 819*w^576 + 1180*w^577 +
1805*w^578 + 1484*w^579 + 1009*w^580 + 5017*w^581 + 2857*w^582 + 8340*w^583 +
10362*w^584 + 6*w^585 + 11919*w^586 + 560*w^587 + 6871*w^588 + 10740*w^589 +
7740*w^590 + 5549*w^591 + 3330*w^592 + 2066*w^593 + 8796*w^594 + 8826*w^595 +
6700*w^596 + 8042*w^597 + 72*w^598 + 1313*w^599 + 10192*w^600 + 3643*w^601 +
4005*w^602 + 2579*w^603 + 7143*w^604 + 11087*w^605 + 6942*w^606 + 12048*w^607 +
6169*w^608 + 8448*w^609 + 2967*w^610 + 1313*w^611 + 7082*w^612 + 1395*w^613 +
4493*w^614 + 6170*w^615 + 12197*w^616 + 9910*w^617 + 10375*w^618 + 7686*w^619 +
936*w^620 + 6213*w^621 + 5017*w^622 + 7615*w^623 + 8186*w^624 + 9353*w^625 +
552*w^626 + 3763*w^627 + 6171*w^628 + 2264*w^629 + 8959*w^630 + 2986*w^631 +
6894*w^632 + 3340*w^633 + 6069*w^634 + 8860*w^635 + 3888*w^636 + 662*w^637 +
10095*w^638 + 11078*w^639 + 10066*w^640 + 441*w^641 + 1521*w^642 + 2959*w^643 +
```

```
11358*w^644 + 3011*w^645 + 2982*w^646 + 2638*w^647 + 10278*w^648 + 5674*w^649 +
5647*w^650 + 12269*w^651 + 11602*w^652 + 2772*w^653 + 2043*w^654 + 4605*w^655 +
11669*w^656 + 484*w^657 + 6778*w^658 + 2337*w^659 + 4361*w^660 + 4574*w^661 +
11571*w^662 + 1185*w^663 + 10138*w^664 + 7313*w^665 + 7107*w^666 + 4716*w^667 +
4242*w^668 + 264*w^669 + 5140*w^670 + 3272*w^671 + 4429*w^672 + 9219*w^673 +
3309*w^674 + 6570*w^675 + 4559*w^676 + 10808*w^677 + 1084*w^678 + 3260*w^679 +
3230*w^680 + 2619*w^681 + 11520*w^682 + 5513*w^683 + 10954*w^684 + 11318*w^685 +
11751*w^686 + 6289*w^687 + 9618*w^688 + 5660*w^689 + 5041*w^690 + 1766*w^691 +
3192*w^692 + 3503*w^693 + 1696*w^694 + 4925*w^695 + 9988*w^696 + 2459*w^697 +
2175*w^698 + 2207*w^699 + 7781*w^700 + 10303*w^701 + 10967*w^702 + 7459*w^703 +
8879*w^704 + 2355*w^705 + 8913*w^706 + 8235*w^707 + 7505*w^708 + 9935*w^709 +
2411*w^710 + 4634*w^711 + 11191*w^712 + 3354*w^713 + 5125*w^714 + 11814*w^715 +
6547*w^716 + 0*w^717 + 7460*w^718 + 9841*w^719 + 8803*w^720 + 11023*w^721 +
3301*w^722 + 2255*w^723 + 3551*w^724 + 10617*w^725 + 187*w^726 + 2753*w^727 +
7954*w^728 + 9417*w^729 + 11022*w^730 + 4119*w^731 + 8447*w^732 + 12028*w^733 +
11849*w^734 + 11907*w^735 + 10098*w^736 + 2164*w^737 + 10304*w^738 + 9743*w^739
+ 1918*w^740 + 4945*w^741 + 11893*w^742 + 653*w^743 + 10274*w^744 + 9996*w^745 +
1220*w^746 + 3730*w^747 + 2506*w^748 + 7524*w^749 + 5047*w^750 + 3681*w^751 +
6466*w^752 + 8885*w^753 + 6619*w^754 + 7527*w^755 + 569*w^756 + 2379*w^757 +
1208*w^758 + 12144*w^759 + 10257*w^760 + 10885*w^761 + 4166*w^762 + 5821*w^763 +
5112*w^764 + 11660*w^765 + 94*w^766 + 11367*w^767 + 7959*w^768 + 1413*w^769 +
52*w^770 + 2377*w^771 + 3575*w^772 + 10397*w^773 + 6420*w^774 + 8900*w^775 +
11182*w^776 + 5978*w^777 + 8142*w^778 + 7420*w^779 + 11949*w^780 + 10637*w^781 +
11717*w^782 + 1908*w^783 + 5474*w^784 + 5114*w^785 + 5704*w^786 + 1716*w^787 +
3520*w^788 + 10399*w^789 + 1636*w^790 + 5969*w^791 + 1834*w^792 + 368*w^793 +
11209*w^794 + 12150*w^795 + 305*w^796 + 2653*w^797 + 540*w^798 + 5917*w^799 +
6320*w^800 + 12061*w^801 + 6569*w^802 + 9736*w^803 + 3489*w^804 + 4807*w^805 +
8848*w^806 + 6031*w^807 + 2753*w^808 + 8873*w^809 + 4518*w^810 + 7584*w^811 +
1425*w^812 + 1488*w^813 + 4888*w^814 + 1879*w^815 + 5999*w^816 + 376*w^817 +
1879*w^818 + 10591*w^819 + 10817*w^820 + 5093*w^821 + 11509*w^822 + 10871*w^823
+ 3500*w^824 + 5939*w^825 + 4470*w^826 + 8215*w^827 + 967*w^828 + 11687*w^829 +
4826*w^830 + 301*w^831 + 2885*w^832 + 2976*w^833 + 8728*w^834 + 4192*w^835 +
8658*w^836 + 2841*w^837 + 8169*w^838 + 7443*w^839 + 11976*w^840 + 11945*w^841 +
8859*w^842 + 5221*w^843 + 12228*w^844 + 5773*w^845 + 6329*w^846 + 4993*w^847 +
3815*w^848 + 8151*w^849 + 7438*w^850 + 8648*w^851 + 11712*w^852 + 5989*w^853 +
2079*w^854 + 5745*w^855 + 8908*w^856 + 1605*w^857 + 10072*w^858 + 2557*w^859 +
4879*w^860 + 11167*w^861 + 7936*w^862 + 56*w^863 + 12067*w^864 + 3839*w^865 +
10678*w^866 + 7782*w^867 + 6222*w^868 + 5440*w^869 + 5088*w^870 + 10501*w^871 +
1112*w^872 + 6949*w^873 + 6069*w^874 + 8301*w^875 + 7156*w^876 + 9005*w^877 +
2311*w^878 + 1925*w^879 + 3528*w^880 + 6711*w^881 + 5941*w^882 + 8757*w^883 +
4347*w^884 + 9899*w^885 + 2569*w^886 + 8483*w^887 + 12239*w^888 + 887*w^889 +
2979*w^890 + 6598*w^891 + 10759*w^892 + 7719*w^893 + 10831*w^894 + 774*w^895 +
11560*w^896 + 7858*w^897 + 9561*w^898 + 9064*w^899 + 10539*w^900 + 2390*w^901 +
4339*w^902 + 5139*w^903 + 2128*w^904 + 4330*w^905 + 8235*w^906 + 1444*w^907 +
11466*w^908 + 8422*w^909 + 11270*w^910 + 5877*w^911 + 5447*w^912 + 10684*w^913 +
8364*w^914 + 3730*w^915 + 5537*w^916 + 7847*w^917 + 11037*w^918 + 4012*w^919 +
4944*w^920 + 5656*w^921 + 5282*w^922 + 4887*w^923 + 2389*w^924 + 8398*w^925 +
10092*w^926 + 5055*w^927 + 9588*w^928 + 10495*w^929 + 10916*w^930 + 7299*w^931 +
```

```
3322*w^932 + 6910*w^933 + 8344*w^934 + 6881*w^935 + 4102*w^936 + 3387*w^937 +
9284*w^938 + 5240*w^939 + 5847*w^940 + 0*w^941 + 0*w^942 + 0*w^943 + 0*w^944 +
0*w^945 + 0*w^946 + 0*w^947 + 0*w^948 + 0*w^949 + 0*w^950 + 0*w^951 + 0*w^952 +
0*w^953 + 0*w^954 + 0*w^955 + 0*w^956 + 0*w^957 + 0*w^958 + 0*w^959 + 0*w^960 +
0*w^961 + 0*w^962 + 0*w^963 + 0*w^964 + 0*w^965 + 0*w^966 + 0*w^967 + 0*w^968 +
0*w^969 + 0*w^970 + 0*w^971 + 0*w^972 + 0*w^973 + 0*w^974 + 0*w^975 + 0*w^976 +
0*w^977 + 0*w^978 + 0*w^979 + 0*w^980 + 0*w^981 + 0*w^982 + 0*w^983 + 0*w^984 +
0*w^985 + 0*w^986 + 0*w^987 + 0*w^988 + 0*w^989 + 0*w^990 + 0*w^991 + 0*w^992 +
0*w^993 + 0*w^994 + 0*w^995 + 0*w^996 + 0*w^997 + 0*w^998 + 0*w^999 + 0*w^1000 +
0*w^1001 + 0*w^1002 + 0*w^1003 + 0*w^1004 + 0*w^1005 + 0*w^1006 + 0*w^1007 +
0*w^1008 + 0*w^1009 + 0*w^1010 + 0*w^1011 + 0*w^1012 + 0*w^1013 + 0*w^1014 +
0*w^1015 + 0*w^1016 + 0*w^1017 + 0*w^1018 + 0*w^1019 + 0*w^1020 + 0*w^1021 +
0*w^1022 + 0*w^1023
```

De seguida calculamos o polinómio transformado através da função NTT.

[ ]:
```
ff = ntt(f,N,xi,F)
print("ff = ",end='')
for i in range(N-1):
    print(f"{ff[i]}*w^{i} + ",end='')
print(f"{ff[N-1]}*w^{N-1}")
```

```
ff = 4169*w^0 + 531*w^1 + 12126*w^2 + 10913*w^3 + 4137*w^4 + 1774*w^5 +
10884*w^6 + 1324*w^7 + 10774*w^8 + 12067*w^9 + 8936*w^10 + 11542*w^11 +
2456*w^12 + 3749*w^13 + 9328*w^14 + 657*w^15 + 3105*w^16 + 6472*w^17 + 4180*w^18
+ 10751*w^19 + 10651*w^20 + 11180*w^21 + 10068*w^22 + 9229*w^23 + 10322*w^24 +
8704*w^25 + 971*w^26 + 4749*w^27 + 922*w^28 + 11833*w^29 + 5557*w^30 +
10396*w^31 + 7588*w^32 + 4966*w^33 + 1132*w^34 + 6611*w^35 + 3614*w^36 +
4626*w^37 + 11609*w^38 + 1129*w^39 + 2757*w^40 + 8157*w^41 + 8800*w^42 +
9249*w^43 + 3400*w^44 + 10067*w^45 + 11000*w^46 + 1667*w^47 + 7075*w^48 +
2279*w^49 + 7771*w^50 + 1600*w^51 + 3791*w^52 + 5925*w^53 + 2754*w^54 +
1055*w^55 + 2833*w^56 + 8684*w^57 + 8059*w^58 + 3996*w^59 + 10456*w^60 +
9371*w^61 + 2738*w^62 + 8962*w^63 + 3925*w^64 + 8480*w^65 + 2032*w^66 +
1444*w^67 + 5126*w^68 + 35*w^69 + 5691*w^70 + 3861*w^71 + 1993*w^72 + 3736*w^73
+ 952*w^74 + 8409*w^75 + 296*w^76 + 1848*w^77 + 2011*w^78 + 2664*w^79 + 522*w^80
+ 10136*w^81 + 9448*w^82 + 9586*w^83 + 10314*w^84 + 657*w^85 + 9082*w^86 +
11579*w^87 + 11319*w^88 + 7359*w^89 + 623*w^90 + 11357*w^91 + 7281*w^92 +
9870*w^93 + 3763*w^94 + 7538*w^95 + 565*w^96 + 1720*w^97 + 639*w^98 + 7632*w^99
+ 7611*w^100 + 6481*w^101 + 4141*w^102 + 9182*w^103 + 306*w^104 + 10591*w^105 +
10737*w^106 + 2605*w^107 + 8816*w^108 + 1114*w^109 + 2725*w^110 + 7793*w^111 +
5392*w^112 + 9294*w^113 + 2257*w^114 + 6544*w^115 + 4636*w^116 + 8748*w^117 +
11146*w^118 + 6580*w^119 + 5123*w^120 + 9237*w^121 + 9584*w^122 + 2311*w^123 +
3178*w^124 + 7865*w^125 + 4492*w^126 + 11209*w^127 + 9505*w^128 + 8882*w^129 +
8013*w^130 + 9465*w^131 + 11903*w^132 + 7905*w^133 + 6905*w^134 + 948*w^135 +
2245*w^136 + 2677*w^137 + 5889*w^138 + 2186*w^139 + 9973*w^140 + 10369*w^141 +
11076*w^142 + 6412*w^143 + 11299*w^144 + 7775*w^145 + 8644*w^146 + 3527*w^147 +
6324*w^148 + 9323*w^149 + 8439*w^150 + 979*w^151 + 5020*w^152 + 1174*w^153 +
764*w^154 + 3928*w^155 + 7250*w^156 + 7206*w^157 + 1505*w^158 + 1239*w^159 +
```

```
12001*w^160 + 9879*w^161 + 5571*w^162 + 7249*w^163 + 8013*w^164 + 4425*w^165 +
2077*w^166 + 2307*w^167 + 7118*w^168 + 3526*w^169 + 1585*w^170 + 6650*w^171 +
2877*w^172 + 3194*w^173 + 5151*w^174 + 1097*w^175 + 9922*w^176 + 2632*w^177 +
10916*w^178 + 6585*w^179 + 4345*w^180 + 4400*w^181 + 6340*w^182 + 8128*w^183 +
5852*w^184 + 2637*w^185 + 5131*w^186 + 6490*w^187 + 1510*w^188 + 614*w^189 +
7004*w^190 + 2434*w^191 + 1417*w^192 + 2138*w^193 + 8414*w^194 + 6374*w^195 +
7388*w^196 + 4334*w^197 + 2901*w^198 + 10796*w^199 + 7601*w^200 + 2798*w^201 +
10736*w^202 + 5548*w^203 + 7247*w^204 + 3865*w^205 + 12196*w^206 + 2643*w^207 +
1937*w^208 + 2090*w^209 + 11892*w^210 + 4323*w^211 + 1624*w^212 + 2766*w^213 +
5198*w^214 + 5238*w^215 + 78*w^216 + 2068*w^217 + 11079*w^218 + 865*w^219 +
2346*w^220 + 5929*w^221 + 6146*w^222 + 1505*w^223 + 298*w^224 + 10762*w^225 +
9157*w^226 + 9903*w^227 + 11801*w^228 + 7079*w^229 + 5239*w^230 + 11017*w^231 +
1758*w^232 + 7764*w^233 + 11499*w^234 + 3370*w^235 + 584*w^236 + 10144*w^237 +
7039*w^238 + 1698*w^239 + 5647*w^240 + 2835*w^241 + 3327*w^242 + 7154*w^243 +
6815*w^244 + 3311*w^245 + 690*w^246 + 10642*w^247 + 4116*w^248 + 6968*w^249 +
3114*w^250 + 8503*w^251 + 1411*w^252 + 6703*w^253 + 4799*w^254 + 170*w^255 +
428*w^256 + 2524*w^257 + 6374*w^258 + 6834*w^259 + 7430*w^260 + 5778*w^261 +
4981*w^262 + 7716*w^263 + 283*w^264 + 10810*w^265 + 1231*w^266 + 6829*w^267 +
11688*w^268 + 8754*w^269 + 4148*w^270 + 8865*w^271 + 10788*w^272 + 11602*w^273 +
7717*w^274 + 10351*w^275 + 6173*w^276 + 4255*w^277 + 526*w^278 + 1146*w^279 +
2024*w^280 + 293*w^281 + 7357*w^282 + 3478*w^283 + 5534*w^284 + 9755*w^285 +
1738*w^286 + 6863*w^287 + 5624*w^288 + 1056*w^289 + 4932*w^290 + 4237*w^291 +
10769*w^292 + 7812*w^293 + 7564*w^294 + 3888*w^295 + 12204*w^296 + 2913*w^297 +
4988*w^298 + 10350*w^299 + 8712*w^300 + 10006*w^301 + 1097*w^302 + 3333*w^303 +
1562*w^304 + 997*w^305 + 4284*w^306 + 1655*w^307 + 10482*w^308 + 7403*w^309 +
6887*w^310 + 909*w^311 + 4583*w^312 + 11769*w^313 + 2240*w^314 + 7464*w^315 +
1485*w^316 + 5410*w^317 + 1341*w^318 + 10746*w^319 + 4610*w^320 + 3564*w^321 +
6789*w^322 + 3705*w^323 + 10154*w^324 + 4982*w^325 + 7809*w^326 + 5722*w^327 +
7711*w^328 + 7401*w^329 + 8970*w^330 + 7105*w^331 + 3506*w^332 + 1893*w^333 +
107*w^334 + 4099*w^335 + 9964*w^336 + 8992*w^337 + 2966*w^338 + 3530*w^339 +
11848*w^340 + 9196*w^341 + 9839*w^342 + 10991*w^343 + 6616*w^344 + 4604*w^345 +
6881*w^346 + 6478*w^347 + 1830*w^348 + 1463*w^349 + 2511*w^350 + 9269*w^351 +
10308*w^352 + 1956*w^353 + 6973*w^354 + 7778*w^355 + 1846*w^356 + 7758*w^357 +
7144*w^358 + 12137*w^359 + 4950*w^360 + 1318*w^361 + 7726*w^362 + 3351*w^363 +
1222*w^364 + 1356*w^365 + 87*w^366 + 12059*w^367 + 1573*w^368 + 8927*w^369 +
5034*w^370 + 6126*w^371 + 10827*w^372 + 1713*w^373 + 11650*w^374 + 6039*w^375 +
3438*w^376 + 6881*w^377 + 7785*w^378 + 6193*w^379 + 3576*w^380 + 5888*w^381 +
517*w^382 + 8141*w^383 + 4730*w^384 + 1637*w^385 + 11492*w^386 + 9524*w^387 +
10169*w^388 + 7133*w^389 + 8435*w^390 + 8789*w^391 + 2314*w^392 + 4487*w^393 +
11910*w^394 + 10594*w^395 + 5756*w^396 + 7893*w^397 + 1005*w^398 + 6030*w^399 +
12287*w^400 + 2467*w^401 + 3800*w^402 + 6772*w^403 + 11376*w^404 + 1510*w^405 +
547*w^406 + 4974*w^407 + 664*w^408 + 11352*w^409 + 7251*w^410 + 9514*w^411 +
4431*w^412 + 11925*w^413 + 1004*w^414 + 7283*w^415 + 11951*w^416 + 5019*w^417 +
2656*w^418 + 1580*w^419 + 6108*w^420 + 9561*w^421 + 8442*w^422 + 12199*w^423 +
9332*w^424 + 3500*w^425 + 10299*w^426 + 3154*w^427 + 9164*w^428 + 11728*w^429 +
10137*w^430 + 8601*w^431 + 5574*w^432 + 5198*w^433 + 4260*w^434 + 10350*w^435 +
12163*w^436 + 9309*w^437 + 7489*w^438 + 8862*w^439 + 3133*w^440 + 153*w^441 +
8051*w^442 + 6203*w^443 + 9322*w^444 + 10881*w^445 + 7195*w^446 + 2216*w^447 +
```

```
735*w^448 + 214*w^449 + 7223*w^450 + 408*w^451 + 10754*w^452 + 9763*w^453 +
6669*w^454 + 7929*w^455 + 8117*w^456 + 4516*w^457 + 12084*w^458 + 8880*w^459 +
6603*w^460 + 8109*w^461 + 9262*w^462 + 6397*w^463 + 11521*w^464 + 2399*w^465 +
11686*w^466 + 4475*w^467 + 6539*w^468 + 9762*w^469 + 2266*w^470 + 6796*w^471 +
6003*w^472 + 4423*w^473 + 5865*w^474 + 6687*w^475 + 5216*w^476 + 7520*w^477 +
9626*w^478 + 11814*w^479 + 5641*w^480 + 9670*w^481 + 5289*w^482 + 6194*w^483 +
2248*w^484 + 11270*w^485 + 5469*w^486 + 8634*w^487 + 3771*w^488 + 1980*w^489 +
6055*w^490 + 1460*w^491 + 11205*w^492 + 2128*w^493 + 10615*w^494 + 5711*w^495 +
5135*w^496 + 890*w^497 + 7029*w^498 + 2504*w^499 + 9467*w^500 + 1767*w^501 +
883*w^502 + 9069*w^503 + 10168*w^504 + 1426*w^505 + 1662*w^506 + 7272*w^507 +
4605*w^508 + 11372*w^509 + 2125*w^510 + 274*w^511 + 4942*w^512 + 7425*w^513 +
4149*w^514 + 4708*w^515 + 11144*w^516 + 2036*w^517 + 514*w^518 + 9650*w^519 +
12235*w^520 + 6417*w^521 + 11424*w^522 + 2461*w^523 + 1758*w^524 + 11299*w^525 +
4761*w^526 + 1165*w^527 + 2541*w^528 + 9050*w^529 + 9267*w^530 + 12031*w^531 +
2047*w^532 + 4479*w^533 + 10100*w^534 + 3458*w^535 + 5518*w^536 + 2208*w^537 +
4531*w^538 + 815*w^539 + 9998*w^540 + 6392*w^541 + 2102*w^542 + 119*w^543 +
4120*w^544 + 5513*w^545 + 2366*w^546 + 1861*w^547 + 5416*w^548 + 3441*w^549 +
1385*w^550 + 4530*w^551 + 11550*w^552 + 3778*w^553 + 11184*w^554 + 8333*w^555 +
9806*w^556 + 871*w^557 + 2838*w^558 + 3617*w^559 + 10850*w^560 + 186*w^561 +
10313*w^562 + 8843*w^563 + 12273*w^564 + 51*w^565 + 6499*w^566 + 6705*w^567 +
1491*w^568 + 11029*w^569 + 8954*w^570 + 1375*w^571 + 8485*w^572 + 587*w^573 +
9783*w^574 + 1985*w^575 + 5684*w^576 + 1030*w^577 + 3122*w^578 + 4141*w^579 +
2055*w^580 + 486*w^581 + 5688*w^582 + 3916*w^583 + 4022*w^584 + 5424*w^585 +
6900*w^586 + 10009*w^587 + 802*w^588 + 1939*w^589 + 11997*w^590 + 11652*w^591 +
10123*w^592 + 10556*w^593 + 5197*w^594 + 2724*w^595 + 3790*w^596 + 6050*w^597 +
2929*w^598 + 2940*w^599 + 5907*w^600 + 7060*w^601 + 5819*w^602 + 2005*w^603 +
306*w^604 + 10955*w^605 + 2239*w^606 + 10235*w^607 + 1400*w^608 + 3735*w^609 +
11717*w^610 + 6265*w^611 + 8521*w^612 + 2715*w^613 + 7454*w^614 + 12247*w^615 +
399*w^616 + 9045*w^617 + 7648*w^618 + 4310*w^619 + 8462*w^620 + 405*w^621 +
3029*w^622 + 11211*w^623 + 3550*w^624 + 462*w^625 + 2033*w^626 + 7048*w^627 +
6905*w^628 + 1712*w^629 + 11661*w^630 + 2710*w^631 + 8779*w^632 + 7954*w^633 +
82*w^634 + 10160*w^635 + 10509*w^636 + 3459*w^637 + 548*w^638 + 673*w^639 +
6925*w^640 + 6475*w^641 + 10992*w^642 + 2155*w^643 + 10630*w^644 + 7166*w^645 +
6168*w^646 + 11224*w^647 + 5441*w^648 + 8645*w^649 + 10029*w^650 + 4996*w^651 +
7382*w^652 + 11208*w^653 + 8010*w^654 + 1254*w^655 + 1002*w^656 + 8632*w^657 +
8102*w^658 + 2439*w^659 + 2060*w^660 + 6786*w^661 + 11545*w^662 + 6320*w^663 +
890*w^664 + 3096*w^665 + 1300*w^666 + 7206*w^667 + 6468*w^668 + 12203*w^669 +
6112*w^670 + 1999*w^671 + 4512*w^672 + 11267*w^673 + 9329*w^674 + 4132*w^675 +
6724*w^676 + 4865*w^677 + 6617*w^678 + 4650*w^679 + 3546*w^680 + 2649*w^681 +
11530*w^682 + 3821*w^683 + 11536*w^684 + 11801*w^685 + 2541*w^686 + 6908*w^687 +
6964*w^688 + 11978*w^689 + 8502*w^690 + 5727*w^691 + 2998*w^692 + 496*w^693 +
8932*w^694 + 5686*w^695 + 866*w^696 + 11914*w^697 + 1165*w^698 + 7233*w^699 +
9739*w^700 + 2530*w^701 + 11589*w^702 + 12098*w^703 + 3925*w^704 + 10846*w^705 +
5558*w^706 + 8370*w^707 + 7023*w^708 + 6234*w^709 + 9533*w^710 + 10386*w^711 +
6150*w^712 + 5794*w^713 + 5828*w^714 + 9327*w^715 + 428*w^716 + 5925*w^717 +
7102*w^718 + 8222*w^719 + 7144*w^720 + 5858*w^721 + 7745*w^722 + 10934*w^723 +
5070*w^724 + 11029*w^725 + 9459*w^726 + 10881*w^727 + 5737*w^728 + 5614*w^729 +
10880*w^730 + 8663*w^731 + 10914*w^732 + 1180*w^733 + 9486*w^734 + 9067*w^735 +
```

```
3882*w^736 + 5296*w^737 + 1542*w^738 + 9405*w^739 + 4146*w^740 + 9938*w^741 +
198*w^742 + 8191*w^743 + 11835*w^744 + 9356*w^745 + 10584*w^746 + 12199*w^747 +
11362*w^748 + 10374*w^749 + 8960*w^750 + 1344*w^751 + 2257*w^752 + 11768*w^753 +
2912*w^754 + 8489*w^755 + 3849*w^756 + 4925*w^757 + 9368*w^758 + 6645*w^759 +
8215*w^760 + 6751*w^761 + 4303*w^762 + 11938*w^763 + 8528*w^764 + 24*w^765 +
5203*w^766 + 160*w^767 + 556*w^768 + 2600*w^769 + 5275*w^770 + 9825*w^771 +
2925*w^772 + 3734*w^773 + 6201*w^774 + 5657*w^775 + 6054*w^776 + 6670*w^777 +
11150*w^778 + 5060*w^779 + 4388*w^780 + 4278*w^781 + 10163*w^782 + 617*w^783 +
597*w^784 + 6414*w^785 + 11990*w^786 + 1104*w^787 + 6330*w^788 + 947*w^789 +
7392*w^790 + 5258*w^791 + 3155*w^792 + 4886*w^793 + 11766*w^794 + 923*w^795 +
2490*w^796 + 5850*w^797 + 8145*w^798 + 9414*w^799 + 8719*w^800 + 11945*w^801 +
3963*w^802 + 7810*w^803 + 10260*w^804 + 8312*w^805 + 7040*w^806 + 7009*w^807 +
8933*w^808 + 1801*w^809 + 5855*w^810 + 4908*w^811 + 11359*w^812 + 3072*w^813 +
11105*w^814 + 4714*w^815 + 9903*w^816 + 5276*w^817 + 9507*w^818 + 504*w^819 +
6634*w^820 + 4954*w^821 + 2405*w^822 + 1808*w^823 + 6712*w^824 + 9064*w^825 +
6357*w^826 + 4182*w^827 + 6496*w^828 + 9623*w^829 + 7523*w^830 + 5772*w^831 +
1107*w^832 + 2647*w^833 + 6289*w^834 + 12237*w^835 + 7618*w^836 + 3024*w^837 +
11562*w^838 + 1178*w^839 + 7281*w^840 + 8766*w^841 + 10526*w^842 + 8224*w^843 +
44*w^844 + 6059*w^845 + 10071*w^846 + 2536*w^847 + 8503*w^848 + 4587*w^849 +
8737*w^850 + 4996*w^851 + 28*w^852 + 5023*w^853 + 5601*w^854 + 6078*w^855 +
6441*w^856 + 6705*w^857 + 5867*w^858 + 11237*w^859 + 997*w^860 + 8226*w^861 +
8803*w^862 + 6733*w^863 + 6838*w^864 + 9472*w^865 + 771*w^866 + 9618*w^867 +
5265*w^868 + 8375*w^869 + 6482*w^870 + 2907*w^871 + 12137*w^872 + 3849*w^873 +
7843*w^874 + 8993*w^875 + 7877*w^876 + 2535*w^877 + 5498*w^878 + 1182*w^879 +
7404*w^880 + 944*w^881 + 10995*w^882 + 7813*w^883 + 11495*w^884 + 2953*w^885 +
11849*w^886 + 1187*w^887 + 4564*w^888 + 8512*w^889 + 6927*w^890 + 1402*w^891 +
8985*w^892 + 11137*w^893 + 11879*w^894 + 8420*w^895 + 8059*w^896 + 10569*w^897 +
7992*w^898 + 4276*w^899 + 8939*w^900 + 1965*w^901 + 12278*w^902 + 4873*w^903 +
10180*w^904 + 3905*w^905 + 6014*w^906 + 10134*w^907 + 6063*w^908 + 7396*w^909 +
624*w^910 + 7533*w^911 + 10456*w^912 + 6666*w^913 + 1406*w^914 + 1104*w^915 +
3411*w^916 + 9177*w^917 + 9683*w^918 + 5272*w^919 + 10056*w^920 + 4246*w^921 +
9735*w^922 + 4819*w^923 + 1752*w^924 + 3907*w^925 + 8955*w^926 + 1989*w^927 +
11480*w^928 + 5199*w^929 + 11854*w^930 + 2840*w^931 + 10549*w^932 + 2808*w^933 +
4466*w^934 + 8704*w^935 + 12048*w^936 + 8663*w^937 + 3981*w^938 + 6014*w^939 +
11022*w^940 + 2882*w^941 + 1068*w^942 + 9001*w^943 + 4989*w^944 + 3907*w^945 +
6272*w^946 + 11408*w^947 + 9167*w^948 + 7629*w^949 + 1708*w^950 + 5453*w^951 +
12119*w^952 + 2736*w^953 + 8664*w^954 + 7896*w^955 + 3494*w^956 + 5569*w^957 +
7879*w^958 + 4152*w^959 + 6616*w^960 + 1836*w^961 + 10342*w^962 + 6268*w^963 +
898*w^964 + 8890*w^965 + 839*w^966 + 10490*w^967 + 189*w^968 + 743*w^969 +
5701*w^970 + 11357*w^971 + 7478*w^972 + 1922*w^973 + 3821*w^974 + 1378*w^975 +
2711*w^976 + 8714*w^977 + 275*w^978 + 4125*w^979 + 10844*w^980 + 540*w^981 +
3497*w^982 + 4642*w^983 + 1586*w^984 + 3183*w^985 + 3125*w^986 + 3188*w^987 +
4197*w^988 + 2559*w^989 + 99*w^990 + 765*w^991 + 12080*w^992 + 4136*w^993 +
10925*w^994 + 7872*w^995 + 3929*w^996 + 5435*w^997 + 12143*w^998 + 8458*w^999 +
8778*w^1000 + 7782*w^1001 + 7382*w^1002 + 7496*w^1003 + 11881*w^1004 +
6026*w^1005 + 6988*w^1006 + 9613*w^1007 + 8426*w^1008 + 12015*w^1009 +
2536*w^1010 + 6894*w^1011 + 9426*w^1012 + 8821*w^1013 + 11754*w^1014 +
8952*w^1015 + 1166*w^1016 + 8413*w^1017 + 4007*w^1018 + 3578*w^1019 +
```

```
5110*w^1020 + 1906*w^1021 + 3295*w^1022 + 8666*w^1023
```

Por fim calculamos o polinómio original através da função ntt_inv, utilizando para isso o polinómio transformado, o tamanho do polinómio e o array de bases do crt.

Para verificar fazemos a comparação entre o polinómio original e o polinómio obtido através da inversa da transformada NTT do polinómio transformado.

[ ]:
```
fff = ntt_inv(ff,N,base)
print("fff = ",end='')
for i in range(N-1):
    print(f"{fff[i]}*w^{i} + ",end='')
print(f"{fff[N-1]}*w^{N-1}")

print("Correto ? ",f == fff)
```

```
fff = 2985*w^0 + 10539*w^1 + 2837*w^2 + 728*w^3 + 917*w^4 + 3665*w^5 + 3724*w^6
+ 4560*w^7 + 1125*w^8 + 939*w^9 + 5063*w^10 + 8343*w^11 + 9392*w^12 + 4593*w^13
+ 8260*w^14 + 10329*w^15 + 3523*w^16 + 10935*w^17 + 5634*w^18 + 9333*w^19 +
9175*w^20 + 5411*w^21 + 7285*w^22 + 7498*w^23 + 1579*w^24 + 2717*w^25 +
11477*w^26 + 2852*w^27 + 763*w^28 + 3609*w^29 + 3373*w^30 + 11084*w^31 +
5907*w^32 + 12161*w^33 + 7957*w^34 + 8969*w^35 + 10227*w^36 + 5757*w^37 +
9378*w^38 + 11352*w^39 + 4113*w^40 + 136*w^41 + 3088*w^42 + 2100*w^43 +
12115*w^44 + 10175*w^45 + 9557*w^46 + 6705*w^47 + 8583*w^48 + 687*w^49 +
7063*w^50 + 1903*w^51 + 6148*w^52 + 5388*w^53 + 10156*w^54 + 11243*w^55 +
2336*w^56 + 5098*w^57 + 5700*w^58 + 12257*w^59 + 2567*w^60 + 2552*w^61 +
4659*w^62 + 9796*w^63 + 3425*w^64 + 5004*w^65 + 2999*w^66 + 7173*w^67 +
8152*w^68 + 4002*w^69 + 11856*w^70 + 2106*w^71 + 2163*w^72 + 9586*w^73 +
6891*w^74 + 11695*w^75 + 8203*w^76 + 5299*w^77 + 12069*w^78 + 1717*w^79 +
10452*w^80 + 3313*w^81 + 390*w^82 + 4366*w^83 + 9340*w^84 + 4100*w^85 +
10548*w^86 + 11650*w^87 + 12240*w^88 + 4500*w^89 + 7686*w^90 + 6662*w^91 +
4285*w^92 + 2214*w^93 + 6623*w^94 + 10891*w^95 + 2060*w^96 + 3939*w^97 +
7082*w^98 + 6050*w^99 + 7489*w^100 + 1408*w^101 + 5944*w^102 + 3884*w^103 +
6382*w^104 + 2190*w^105 + 7514*w^106 + 6597*w^107 + 4456*w^108 + 10319*w^109 +
2658*w^110 + 5888*w^111 + 11976*w^112 + 3821*w^113 + 8202*w^114 + 9360*w^115 +
407*w^116 + 9291*w^117 + 11926*w^118 + 3989*w^119 + 2698*w^120 + 7811*w^121 +
9379*w^122 + 3394*w^123 + 994*w^124 + 5612*w^125 + 9249*w^126 + 6500*w^127 +
11072*w^128 + 10336*w^129 + 2215*w^130 + 6116*w^131 + 553*w^132 + 9244*w^133 +
409*w^134 + 11480*w^135 + 7256*w^136 + 8045*w^137 + 6536*w^138 + 2888*w^139 +
10998*w^140 + 941*w^141 + 7242*w^142 + 4487*w^143 + 5784*w^144 + 8377*w^145 +
3942*w^146 + 754*w^147 + 3008*w^148 + 1867*w^149 + 3233*w^150 + 5996*w^151 +
1226*w^152 + 5468*w^153 + 10104*w^154 + 1161*w^155 + 6363*w^156 + 6727*w^157 +
8813*w^158 + 5186*w^159 + 3978*w^160 + 4337*w^161 + 2755*w^162 + 8006*w^163 +
2926*w^164 + 2271*w^165 + 6342*w^166 + 7007*w^167 + 11187*w^168 + 11595*w^169 +
8028*w^170 + 5804*w^171 + 9621*w^172 + 5308*w^173 + 6535*w^174 + 10773*w^175 +
6744*w^176 + 7929*w^177 + 5304*w^178 + 5696*w^179 + 3102*w^180 + 269*w^181 +
2722*w^182 + 3711*w^183 + 7045*w^184 + 1221*w^185 + 9551*w^186 + 10419*w^187 +
8256*w^188 + 9067*w^189 + 1455*w^190 + 7298*w^191 + 9502*w^192 + 6260*w^193 +
10406*w^194 + 5667*w^195 + 3704*w^196 + 9267*w^197 + 3734*w^198 + 6030*w^199 +
```

```
5819*w^200 + 2824*w^201 + 4199*w^202 + 11468*w^203 + 4607*w^204 + 3741*w^205 +
11522*w^206 + 4345*w^207 + 3224*w^208 + 5773*w^209 + 1610*w^210 + 7318*w^211 +
10181*w^212 + 11977*w^213 + 6434*w^214 + 10248*w^215 + 11977*w^216 + 8311*w^217
+ 826*w^218 + 6912*w^219 + 1338*w^220 + 4691*w^221 + 7398*w^222 + 3699*w^223 +
4406*w^224 + 407*w^225 + 7807*w^226 + 5209*w^227 + 9847*w^228 + 5021*w^229 +
585*w^230 + 6923*w^231 + 2129*w^232 + 1585*w^233 + 11834*w^234 + 9840*w^235 +
10612*w^236 + 1943*w^237 + 7960*w^238 + 10607*w^239 + 7989*w^240 + 10357*w^241 +
6581*w^242 + 2659*w^243 + 1854*w^244 + 6572*w^245 + 8167*w^246 + 7331*w^247 +
7453*w^248 + 2607*w^249 + 8376*w^250 + 8360*w^251 + 6865*w^252 + 129*w^253 +
1493*w^254 + 9754*w^255 + 10206*w^256 + 5369*w^257 + 12197*w^258 + 11081*w^259 +
11674*w^260 + 6749*w^261 + 4580*w^262 + 5829*w^263 + 1978*w^264 + 1972*w^265 +
5392*w^266 + 10141*w^267 + 748*w^268 + 4526*w^269 + 8358*w^270 + 11533*w^271 +
7096*w^272 + 6379*w^273 + 7118*w^274 + 4892*w^275 + 7370*w^276 + 11447*w^277 +
2617*w^278 + 2632*w^279 + 194*w^280 + 4912*w^281 + 5580*w^282 + 2058*w^283 +
2948*w^284 + 6240*w^285 + 121*w^286 + 7817*w^287 + 10363*w^288 + 4271*w^289 +
313*w^290 + 9960*w^291 + 5300*w^292 + 8821*w^293 + 3342*w^294 + 2254*w^295 +
181*w^296 + 5901*w^297 + 3639*w^298 + 3923*w^299 + 1428*w^300 + 2301*w^301 +
5009*w^302 + 11374*w^303 + 1419*w^304 + 8664*w^305 + 71*w^306 + 4446*w^307 +
3742*w^308 + 11409*w^309 + 4858*w^310 + 9377*w^311 + 7573*w^312 + 2645*w^313 +
4000*w^314 + 8607*w^315 + 12166*w^316 + 11657*w^317 + 6908*w^318 + 5337*w^319 +
8080*w^320 + 155*w^321 + 6476*w^322 + 5702*w^323 + 10051*w^324 + 3452*w^325 +
12070*w^326 + 8021*w^327 + 1691*w^328 + 3076*w^329 + 848*w^330 + 9437*w^331 +
6638*w^332 + 10935*w^333 + 3583*w^334 + 9298*w^335 + 6011*w^336 + 7029*w^337 +
4812*w^338 + 10781*w^339 + 7666*w^340 + 5259*w^341 + 2959*w^342 + 8098*w^343 +
6005*w^344 + 6327*w^345 + 10767*w^346 + 10692*w^347 + 3442*w^348 + 8245*w^349 +
3785*w^350 + 3712*w^351 + 9514*w^352 + 6881*w^353 + 4037*w^354 + 2539*w^355 +
6394*w^356 + 10607*w^357 + 3820*w^358 + 844*w^359 + 3648*w^360 + 3351*w^361 +
7022*w^362 + 42*w^363 + 1265*w^364 + 1689*w^365 + 5785*w^366 + 7752*w^367 +
8393*w^368 + 7975*w^369 + 1744*w^370 + 2967*w^371 + 5963*w^372 + 4351*w^373 +
11056*w^374 + 9734*w^375 + 4076*w^376 + 6647*w^377 + 685*w^378 + 1653*w^379 +
5958*w^380 + 7967*w^381 + 4696*w^382 + 6403*w^383 + 314*w^384 + 2079*w^385 +
5225*w^386 + 9492*w^387 + 3075*w^388 + 3551*w^389 + 657*w^390 + 5977*w^391 +
2085*w^392 + 10981*w^393 + 7277*w^394 + 9289*w^395 + 3677*w^396 + 12178*w^397 +
8820*w^398 + 4572*w^399 + 3098*w^400 + 181*w^401 + 5635*w^402 + 5859*w^403 +
6672*w^404 + 9066*w^405 + 176*w^406 + 1214*w^407 + 11903*w^408 + 1210*w^409 +
9510*w^410 + 2491*w^411 + 4498*w^412 + 11490*w^413 + 9366*w^414 + 812*w^415 +
4614*w^416 + 3500*w^417 + 3401*w^418 + 6724*w^419 + 2368*w^420 + 6180*w^421 +
6812*w^422 + 9289*w^423 + 6914*w^424 + 3651*w^425 + 3412*w^426 + 1746*w^427 +
7722*w^428 + 6895*w^429 + 11805*w^430 + 3581*w^431 + 11354*w^432 + 9947*w^433 +
9819*w^434 + 4537*w^435 + 4757*w^436 + 2398*w^437 + 10815*w^438 + 3890*w^439 +
7443*w^440 + 4150*w^441 + 3689*w^442 + 2816*w^443 + 1424*w^444 + 1204*w^445 +
10190*w^446 + 1974*w^447 + 4821*w^448 + 12116*w^449 + 4058*w^450 + 4637*w^451 +
2591*w^452 + 4398*w^453 + 5538*w^454 + 771*w^455 + 4143*w^456 + 3172*w^457 +
2765*w^458 + 2714*w^459 + 10084*w^460 + 10252*w^461 + 11462*w^462 + 10407*w^463
+ 3497*w^464 + 186*w^465 + 541*w^466 + 4112*w^467 + 2803*w^468 + 3196*w^469 +
4253*w^470 + 1350*w^471 + 9208*w^472 + 4105*w^473 + 9804*w^474 + 2092*w^475 +
7173*w^476 + 8295*w^477 + 7816*w^478 + 43*w^479 + 7778*w^480 + 6845*w^481 +
3076*w^482 + 905*w^483 + 560*w^484 + 6931*w^485 + 6180*w^486 + 4954*w^487 +
```

```
6616*w^488 + 8435*w^489 + 8886*w^490 + 8108*w^491 + 5074*w^492 + 5564*w^493 +
33*w^494 + 3170*w^495 + 2454*w^496 + 4144*w^497 + 7431*w^498 + 11957*w^499 +
8269*w^500 + 144*w^501 + 2041*w^502 + 11478*w^503 + 9758*w^504 + 2276*w^505 +
437*w^506 + 3933*w^507 + 11731*w^508 + 9564*w^509 + 3023*w^510 + 1758*w^511 +
690*w^512 + 4019*w^513 + 9946*w^514 + 12282*w^515 + 3771*w^516 + 6895*w^517 +
962*w^518 + 10369*w^519 + 10314*w^520 + 6808*w^521 + 2085*w^522 + 6536*w^523 +
10108*w^524 + 3989*w^525 + 4273*w^526 + 3085*w^527 + 8282*w^528 + 970*w^529 +
3186*w^530 + 6117*w^531 + 656*w^532 + 12224*w^533 + 11339*w^534 + 6123*w^535 +
4563*w^536 + 935*w^537 + 2551*w^538 + 3651*w^539 + 3028*w^540 + 11836*w^541 +
4894*w^542 + 998*w^543 + 2622*w^544 + 5235*w^545 + 8644*w^546 + 9806*w^547 +
2043*w^548 + 4636*w^549 + 221*w^550 + 5020*w^551 + 9176*w^552 + 11730*w^553 +
204*w^554 + 9141*w^555 + 8148*w^556 + 1547*w^557 + 4111*w^558 + 8644*w^559 +
7*w^560 + 1977*w^561 + 11618*w^562 + 3055*w^563 + 9017*w^564 + 6932*w^565 +
490*w^566 + 10854*w^567 + 7494*w^568 + 1898*w^569 + 6409*w^570 + 11656*w^571 +
8462*w^572 + 11716*w^573 + 6537*w^574 + 6175*w^575 + 819*w^576 + 1180*w^577 +
1805*w^578 + 1484*w^579 + 1009*w^580 + 5017*w^581 + 2857*w^582 + 8340*w^583 +
10362*w^584 + 6*w^585 + 11919*w^586 + 560*w^587 + 6871*w^588 + 10740*w^589 +
7740*w^590 + 5549*w^591 + 3330*w^592 + 2066*w^593 + 8796*w^594 + 8826*w^595 +
6700*w^596 + 8042*w^597 + 72*w^598 + 1313*w^599 + 10192*w^600 + 3643*w^601 +
4005*w^602 + 2579*w^603 + 7143*w^604 + 11087*w^605 + 6942*w^606 + 12048*w^607 +
6169*w^608 + 8448*w^609 + 2967*w^610 + 1313*w^611 + 7082*w^612 + 1395*w^613 +
4493*w^614 + 6170*w^615 + 12197*w^616 + 9910*w^617 + 10375*w^618 + 7686*w^619 +
936*w^620 + 6213*w^621 + 5017*w^622 + 7615*w^623 + 8186*w^624 + 9353*w^625 +
552*w^626 + 3763*w^627 + 6171*w^628 + 2264*w^629 + 8959*w^630 + 2986*w^631 +
6894*w^632 + 3340*w^633 + 6069*w^634 + 8860*w^635 + 3888*w^636 + 662*w^637 +
10095*w^638 + 11078*w^639 + 10066*w^640 + 441*w^641 + 1521*w^642 + 2959*w^643 +
11358*w^644 + 3011*w^645 + 2982*w^646 + 2638*w^647 + 10278*w^648 + 5674*w^649 +
5647*w^650 + 12269*w^651 + 11602*w^652 + 2772*w^653 + 2043*w^654 + 4605*w^655 +
11669*w^656 + 484*w^657 + 6778*w^658 + 2337*w^659 + 4361*w^660 + 4574*w^661 +
11571*w^662 + 1185*w^663 + 10138*w^664 + 7313*w^665 + 7107*w^666 + 4716*w^667 +
4242*w^668 + 264*w^669 + 5140*w^670 + 3272*w^671 + 4429*w^672 + 9219*w^673 +
3309*w^674 + 6570*w^675 + 4559*w^676 + 10808*w^677 + 1084*w^678 + 3260*w^679 +
3230*w^680 + 2619*w^681 + 11520*w^682 + 5513*w^683 + 10954*w^684 + 11318*w^685 +
11751*w^686 + 6289*w^687 + 9618*w^688 + 5660*w^689 + 5041*w^690 + 1766*w^691 +
3192*w^692 + 3503*w^693 + 1696*w^694 + 4925*w^695 + 9988*w^696 + 2459*w^697 +
2175*w^698 + 2207*w^699 + 7781*w^700 + 10303*w^701 + 10967*w^702 + 7459*w^703 +
8879*w^704 + 2355*w^705 + 8913*w^706 + 8235*w^707 + 7505*w^708 + 9935*w^709 +
2411*w^710 + 4634*w^711 + 11191*w^712 + 3354*w^713 + 5125*w^714 + 11814*w^715 +
6547*w^716 + 0*w^717 + 7460*w^718 + 9841*w^719 + 8803*w^720 + 11023*w^721 +
3301*w^722 + 2255*w^723 + 3551*w^724 + 10617*w^725 + 187*w^726 + 2753*w^727 +
7954*w^728 + 9417*w^729 + 11022*w^730 + 4119*w^731 + 8447*w^732 + 12028*w^733 +
11849*w^734 + 11907*w^735 + 10098*w^736 + 2164*w^737 + 10304*w^738 + 9743*w^739
+ 1918*w^740 + 4945*w^741 + 11893*w^742 + 653*w^743 + 10274*w^744 + 9996*w^745 +
1220*w^746 + 3730*w^747 + 2506*w^748 + 7524*w^749 + 5047*w^750 + 3681*w^751 +
6466*w^752 + 8885*w^753 + 6619*w^754 + 7527*w^755 + 569*w^756 + 2379*w^757 +
1208*w^758 + 12144*w^759 + 10257*w^760 + 10885*w^761 + 4166*w^762 + 5821*w^763 +
5112*w^764 + 11660*w^765 + 94*w^766 + 11367*w^767 + 7959*w^768 + 1413*w^769 +
52*w^770 + 2377*w^771 + 3575*w^772 + 10397*w^773 + 6420*w^774 + 8900*w^775 +
```

```
11182*w^776 + 5978*w^777 + 8142*w^778 + 7420*w^779 + 11949*w^780 + 10637*w^781 +
11717*w^782 + 1908*w^783 + 5474*w^784 + 5114*w^785 + 5704*w^786 + 1716*w^787 +
3520*w^788 + 10399*w^789 + 1636*w^790 + 5969*w^791 + 1834*w^792 + 368*w^793 +
11209*w^794 + 12150*w^795 + 305*w^796 + 2653*w^797 + 540*w^798 + 5917*w^799 +
6320*w^800 + 12061*w^801 + 6569*w^802 + 9736*w^803 + 3489*w^804 + 4807*w^805 +
8848*w^806 + 6031*w^807 + 2753*w^808 + 8873*w^809 + 4518*w^810 + 7584*w^811 +
1425*w^812 + 1488*w^813 + 4888*w^814 + 1879*w^815 + 5999*w^816 + 376*w^817 +
1879*w^818 + 10591*w^819 + 10817*w^820 + 5093*w^821 + 11509*w^822 + 10871*w^823
+ 3500*w^824 + 5939*w^825 + 4470*w^826 + 8215*w^827 + 967*w^828 + 11687*w^829 +
4826*w^830 + 301*w^831 + 2885*w^832 + 2976*w^833 + 8728*w^834 + 4192*w^835 +
8658*w^836 + 2841*w^837 + 8169*w^838 + 7443*w^839 + 11976*w^840 + 11945*w^841 +
8859*w^842 + 5221*w^843 + 12228*w^844 + 5773*w^845 + 6329*w^846 + 4993*w^847 +
3815*w^848 + 8151*w^849 + 7438*w^850 + 8648*w^851 + 11712*w^852 + 5989*w^853 +
2079*w^854 + 5745*w^855 + 8908*w^856 + 1605*w^857 + 10072*w^858 + 2557*w^859 +
4879*w^860 + 11167*w^861 + 7936*w^862 + 56*w^863 + 12067*w^864 + 3839*w^865 +
10678*w^866 + 7782*w^867 + 6222*w^868 + 5440*w^869 + 5088*w^870 + 10501*w^871 +
1112*w^872 + 6949*w^873 + 6069*w^874 + 8301*w^875 + 7156*w^876 + 9005*w^877 +
2311*w^878 + 1925*w^879 + 3528*w^880 + 6711*w^881 + 5941*w^882 + 8757*w^883 +
4347*w^884 + 9899*w^885 + 2569*w^886 + 8483*w^887 + 12239*w^888 + 887*w^889 +
2979*w^890 + 6598*w^891 + 10759*w^892 + 7719*w^893 + 10831*w^894 + 774*w^895 +
11560*w^896 + 7858*w^897 + 9561*w^898 + 9064*w^899 + 10539*w^900 + 2390*w^901 +
4339*w^902 + 5139*w^903 + 2128*w^904 + 4330*w^905 + 8235*w^906 + 1444*w^907 +
11466*w^908 + 8422*w^909 + 11270*w^910 + 5877*w^911 + 5447*w^912 + 10684*w^913 +
8364*w^914 + 3730*w^915 + 5537*w^916 + 7847*w^917 + 11037*w^918 + 4012*w^919 +
4944*w^920 + 5656*w^921 + 5282*w^922 + 4887*w^923 + 2389*w^924 + 8398*w^925 +
10092*w^926 + 5055*w^927 + 9588*w^928 + 10495*w^929 + 10916*w^930 + 7299*w^931 +
3322*w^932 + 6910*w^933 + 8344*w^934 + 6881*w^935 + 4102*w^936 + 3387*w^937 +
9284*w^938 + 5240*w^939 + 5847*w^940 + 0*w^941 + 0*w^942 + 0*w^943 + 0*w^944 +
0*w^945 + 0*w^946 + 0*w^947 + 0*w^948 + 0*w^949 + 0*w^950 + 0*w^951 + 0*w^952 +
0*w^953 + 0*w^954 + 0*w^955 + 0*w^956 + 0*w^957 + 0*w^958 + 0*w^959 + 0*w^960 +
0*w^961 + 0*w^962 + 0*w^963 + 0*w^964 + 0*w^965 + 0*w^966 + 0*w^967 + 0*w^968 +
0*w^969 + 0*w^970 + 0*w^971 + 0*w^972 + 0*w^973 + 0*w^974 + 0*w^975 + 0*w^976 +
0*w^977 + 0*w^978 + 0*w^979 + 0*w^980 + 0*w^981 + 0*w^982 + 0*w^983 + 0*w^984 +
0*w^985 + 0*w^986 + 0*w^987 + 0*w^988 + 0*w^989 + 0*w^990 + 0*w^991 + 0*w^992 +
0*w^993 + 0*w^994 + 0*w^995 + 0*w^996 + 0*w^997 + 0*w^998 + 0*w^999 + 0*w^1000 +
0*w^1001 + 0*w^1002 + 0*w^1003 + 0*w^1004 + 0*w^1005 + 0*w^1006 + 0*w^1007 +
0*w^1008 + 0*w^1009 + 0*w^1010 + 0*w^1011 + 0*w^1012 + 0*w^1013 + 0*w^1014 +
0*w^1015 + 0*w^1016 + 0*w^1017 + 0*w^1018 + 0*w^1019 + 0*w^1020 + 0*w^1021 +
0*w^1022 + 0*w^1023
Correto ?  True
```

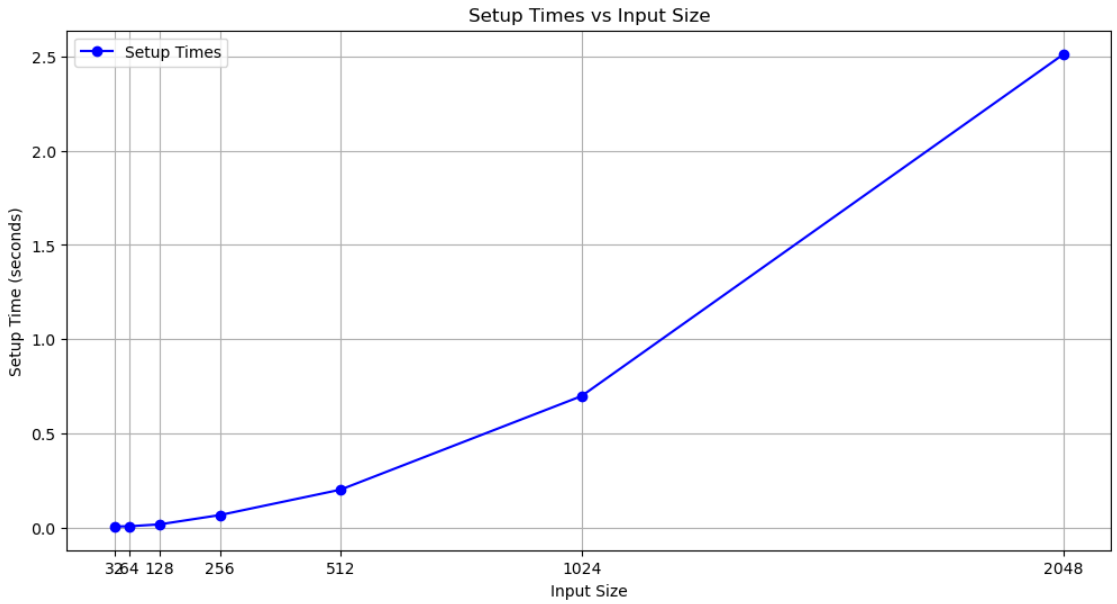
1.3.1 Testes de Performance

Para testar a performance da implementação da transformada NTT e da sua inversa, geramos 100 polinómios de tamanho aleatório entre 1 e N-1 e calculamos o tempo médio de execução da transformada, da sua inversa e do setup necessário para a execução da transformada. Este processo foi realizado para os valores de N: 32, 64, 128, 256, 512, 1024, 2048.

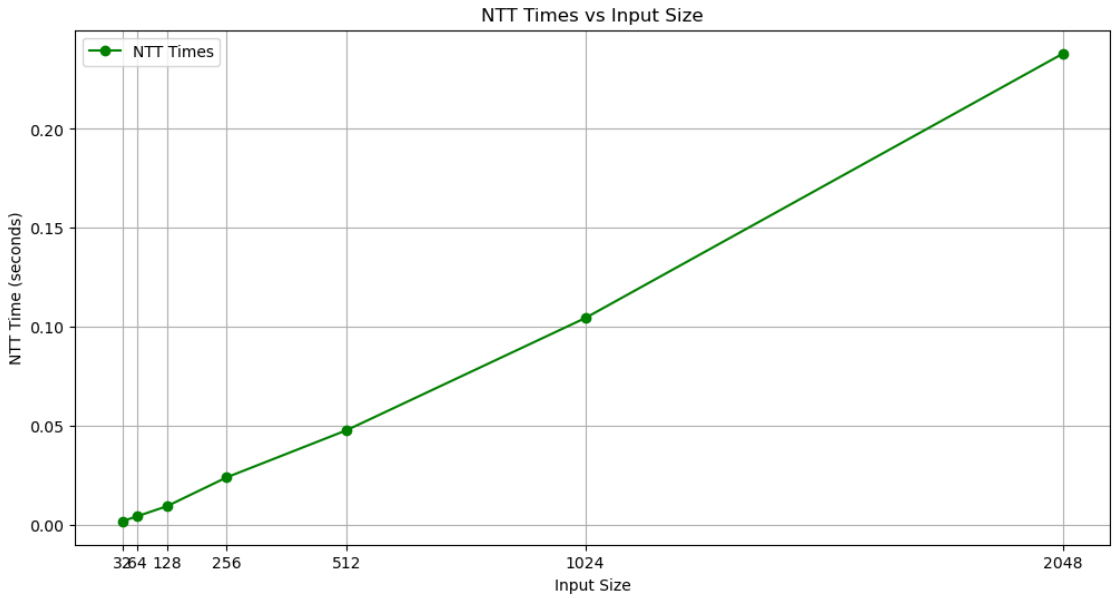
[ ]:
```
import time

avarage_setup_times = []
avarage_ntt_times = []
avarage_inv_times = []

for i in [32,64,128,256,512,1024,2048]:
    times_ntt = []
    times_inv = []
    time_start = time.time()
    correct = 0

    q = find_q(i)
    F = FiniteField(q)
    R = PolynomialRing(F, name="w")
    w = R.gen()
    g = (w^i + 1)
    xi = g.roots(multiplicities=False)[-1]
    rs = [xi^(2*j+1) for j in range(i)]
    base = crt_basis([(w - r) for r in rs])

    time_end = time.time()
    setup_time = time_end-time_start
    avarage_setup_times.append(setup_time)

    print(f'''Vars:
n = {i}
q = {q}

Polynomials:''')

    for _ in range(100):
        f = random_pol(R,i-random.randint(1,i-1))
        print("f = ",f)

        time_start = time.time()
        ff = ntt(f,i,xi,F)
        time_end = time.time()
        times_ntt.append(time_end-time_start)

        time_start = time.time()
        fff = ntt_inv(ff,i,base)
        time_end = time.time()
        times_inv.append(time_end-time_start)

        correct += f == fff
```

```
    avrg_ntt = sum(times_ntt)/len(times_ntt)
    avarage_ntt_times.append(avrg_ntt)
    avrg_inv = sum(times_inv)/len(times_inv)
    avarage_inv_times.append(avrg_inv)

    print(f'''
Results:
Setup Time: {setup_time}
Avarage NTT Time: {avrg_ntt}
Avarage Inv Time: {avrg_inv}
Correct: {correct}/100
''')
```

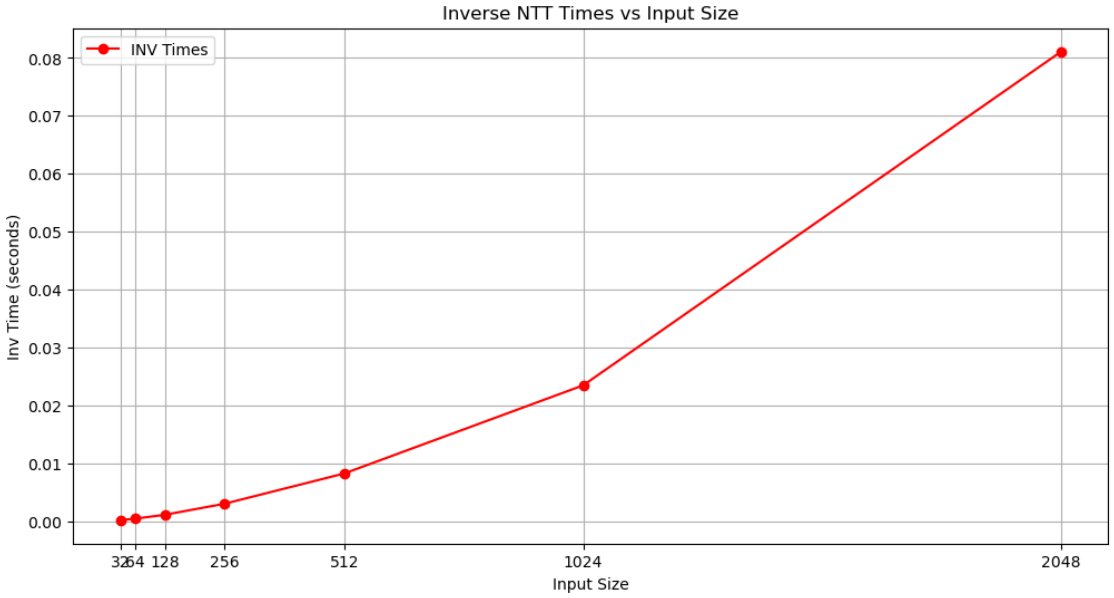
[ ]:
```
import matplotlib.pyplot as plt

input_sizes = [32, 64, 128, 256, 512, 1024, 2048]

plt.figure(figsize=(12, 6))
plt.plot(input_sizes, avarage_setup_times, marker='o', color='b', label='Setup␣
↪Times')
plt.title('Setup Times vs Input Size')
plt.xlabel('Input Size')
plt.ylabel('Setup Time (seconds)')
plt.xticks(input_sizes)
plt.grid(True)
plt.legend()
plt.show()
```



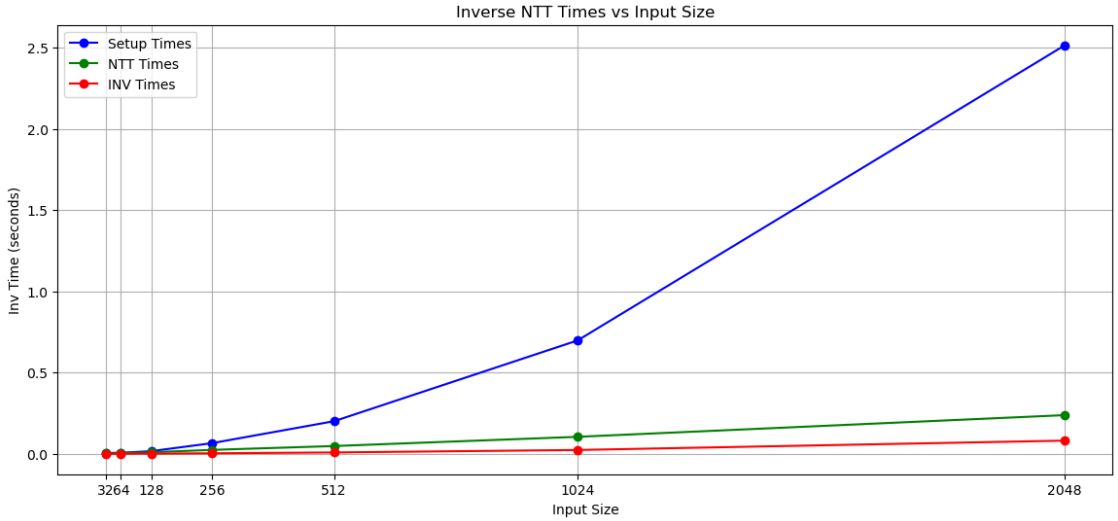
[ ]:
```
plt.figure(figsize=(12, 6))
plt.plot(input_sizes, avarage_ntt_times , marker='o', color='g', label='NTT␣
 ↪Times')
plt.title('NTT Times vs Input Size')
plt.xlabel('Input Size')
plt.ylabel('NTT Time (seconds)')
plt.xticks(input_sizes)
plt.grid(True)
plt.legend()
plt.show()
```



[ ]:
```
plt.figure(figsize=(12, 6))
plt.plot(input_sizes, avarage_inv_times , marker='o', color='r', label='INV␣
 ↪Times')
plt.title('Inverse NTT Times vs Input Size')
plt.xlabel('Input Size')
plt.ylabel('Inv Time (seconds)')
plt.xticks(input_sizes)
plt.grid(True)
plt.legend()
plt.show()
```



[ ]:
```
plt.figure(figsize=(14, 6))
plt.plot(input_sizes, avarage_setup_times, marker='o', color='b', label='Setup␣
↪Times')
plt.plot(input_sizes, avarage_ntt_times , marker='o', color='g', label='NTT␣
↪Times')
plt.plot(input_sizes, avarage_inv_times , marker='o', color='r', label='INV␣
↪Times')
plt.title('Inverse NTT Times vs Input Size')
plt.xlabel('Input Size')
plt.ylabel('Inv Time (seconds)')
plt.xticks(input_sizes)
plt.grid(True)
plt.legend()
plt.show()
```


Inverse NTT Times vs Input Size
Setup Times
NTT Times
INV Times
Inv Time (seconds)
0.0
0.5
1.0
1.5
2.0
2.5
3264
128
256
512
1024
2048
Input Size