

Problem C. Call It What You Want

Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 256 megabytes

As a stereotyped math fanatic, Taylor is expert on utilizing scientific computing tools but he is poor at programming infrastructures, which brings him endless powerlessness.

Recently he worked on factoring polynomials of the form $(x^n - 1)$ over the integers, which aims to express any polynomial of that form as some product of irreducible factors whose coefficients are all in the integers.

With knowledge of the cyclotomic polynomial, he has known that $x^n - 1 = \prod_{d|n} \Phi_d(x)$ where each factor of that is just an irreducible polynomial over the integers. Moreover, $\Phi_n(x) = \prod_{1 \leq k \leq n, \gcd(n,k)=1} (x - w_n^k)$, where $w_n = \cos\left(\frac{2\pi}{n}\right) + \sqrt{-1} \sin\left(\frac{2\pi}{n}\right)$ is the unit complex root of degree n and $\gcd(n, k)$ is the greatest common divisor of n and k .

Although he found such a conclusion, he couldn't obtain the result of some high-degree polynomial in a few seconds. Could you please help him accomplish some factorizations of $(x^n - 1)$?

Here are some examples:

- $\Phi_1(x) = x - 1$;
- $\Phi_2(x) = x + 1, x^2 - 1 = (x - 1)(x + 1)$;
- $\Phi_3(x) = x^2 + x + 1, x^3 - 1 = (x - 1)(x^2 + x + 1)$;
- $\Phi_4(x) = x^2 + 1, x^4 - 1 = (x - 1)(x + 1)(x^2 + 1)$;
- $\Phi_6(x) = x^2 - x + 1, x^6 - 1 = (x - 1)(x + 1)(x^2 - x + 1)(x^2 + x + 1)$;
- $\Phi_{12}(x) = x^4 - x^2 + 1, x^{12} - 1 = (x - 1)(x + 1)(x^2 - x + 1)(x^2 + 1)(x^2 + x + 1)(x^4 - x^2 + 1)$.

Oops! You might have some observations such as the degree of $\Phi_n(x)$ equals to $\varphi(n)$, coefficients of $\Phi_n(x)$ are the same back-to-front as front-to-back except for $\Phi_1(x), \Phi_{p^e}(x) = \Phi_p(x^{p^{e-1}})$ when p is prime, etc. , but they might be worthless for solving.

Input

The first line contains one integer T , indicating the number of test cases.

Each of the following T lines describes a test case and contains only one integer n .

$1 \leq T \leq 100, 2 \leq n \leq 10^5$.

It is guaranteed that the sum of n in all test cases does not exceed $5 \cdot 10^6$.

Output

For each test case, output the factorization as a string without any space in one line, where the polynomials should be sorted in a particular order and each polynomial should be printed in a particular format and enclosed in a pair of parentheses.

Order of polynomials: The order of polynomial $f(x)$ is lower than that of polynomial $g(x)$ if and only if there exists a non-negative integer m such that the coefficient of x^k ($k = m + 1, m + 2, \dots$) in $f(x)$ equals to that of $g(x)$ but the coefficient of x^m in $f(x)$ is less than that of $g(x)$.

Output format of one polynomial: Output all the terms of the polynomial from high-degree to low-degree, each of which should be formed as $\pm c_k x^k$. Additionally,

1. One term should be omitted if its coefficient is zero.
2. The sign of the first term (\pm) should be omitted if the coefficient of that is positive.
3. When $c_k = 1$, c_k should be omitted unless $k = 0$.
4. x^0 should be omitted while x^1 should be written as a simple x .

It is guaranteed that the size of the standard output file does not exceed 26 MiB.

Example

standard input	standard output
5	$(x-1)(x+1)$
2	$(x-1)(x^2+x+1)$
3	$(x-1)(x+1)(x^2+1)$
4	$(x-1)(x+1)(x^2-x+1)(x^2+x+1)$
6	$(x-1)(x+1)(x^2-x+1)(x^2+1)(x^2+x+1)(x^2-x+1)$
12	$4-x^2+1$