MACM 401/MATH 801/CMPT 981
Assignment 3, Spring 2021.

Michael Monagan

Due Monday March 1st at 11pm. Late Penalty: −20% for up to 24 hours late. Zero after that.

Question 1 (15 marks): Chinese Remaindering and Interpolation

Reference sections 5.6 and 5.7

(a) By hand, using Newton's method, find $f(x) \in \mathbb{Z}_5[x]$ such that $f(0) = 1, f(1) = 3, f(2) = 4$ such that $\deg(f) < 3$.

(b) Let $a = (9y - 7)x + (5y^2 + 12)$ and $b = (13y + 23)x^2 + (21y - 11)x + (11y - 13)$ be polynomials in $\mathbb{Z}[y][x]$. Compute the product $a \times b$ using modular homomorphisms ϕ_{p_i} then evaluation homomorphisms $\phi_{y=\beta_j}$ and $\phi_{x=\alpha_k}$ so that you end up multiplying in $\mathbb{Z}_p$. Then use polynomial interpolation and Chinese remaindering to reconstruct the product in $\mathbb{Z}[y][x]$.

First determine how many primes you need and put them in a list. Use $P = [23, 29, 31, 37, ...]$. Then determine how many evaluation points for x and y you need. Use $x = 0, 1, 2, \ldots$ and $y = 0, 1, 2, \ldots$.

The Maple command `Eval(a,x=2) mod p;` can be used to evaluate $a(2, y) \mod p$.
The Maple command for interpolation modulo p is `Interp(...) mod p;`
The Maple command for Chinese remaindering is `chrem(...);`
The Maple command for putting the coefficients of a polynomial a in the symmetric range for $\mathbb{Z}_m$ is `mods(a,m);`

Question 2: The Fast Fourier Transform (25 marks)

(a) Let $n = 2m$ and let ω be a primitive n'th root of unity. To apply the FFT recursively, we use the fact that ω^2 is a primitive m'th root of unity. Prove this.

Also, for $p = 97 = 3 \times 2^5 + 1$, find a primitve 8'th root of unity in $\mathbb{Z}_p$. Use the method in Section 4.8 which first finds a primitive element $1 < \alpha < p - 1$ of $\mathbb{Z}_p$.

(b) What is the Fourier Transform for the polynomial $a(x) = 1 + x + x^2 + \cdots + x^{n-1}$, i.e., what is the vector $[a(1), a(\omega), a(\omega^2), \ldots, a(\omega^{n-1})]$?

(c) Let $M(n)$ be the number of multiplications that the FFT does. A naive implementation of the algorithm leads to this recurrence:

$$M(n) = 2M(n/2) + n + 1 \quad \text{for } n > 1$$

with initial value $M(1) = 0$. In class we said that if we pre-compute the powers ω^i for $0 \le i \le n/2$ and store them in an array W, we can save half the multiplications in the FFT so that

$$M(n) = 2M(n/2) + \frac{n}{2} \quad \text{for } n > 1.$$

By hand, solve this recurrence and show that $M(n) = \frac{1}{2} n \log_2 n$.

(d) Program the FFT in Maple as a recursive procedure. Your Maple procedure should take as input (n, A, p, ω) where n is a power of 2, A is an array of size n storing the input coefficients $a_0, a_1, \ldots, a_{n-1}$, a prime p and ω a primitive n'th root of unity in $\mathbb{Z}_p$. If you want to precompute an array $W = [1, \omega, \omega^2, \ldots, \omega^{n/2-1}]$ of the powers of ω to save multiplications you may do so.

Test your procedure on the following input. Let $A = [1, 2, 3, 4, 3, 2, 1, 0]$, $p = 97$ and w be the primitive 8'th root of unity. To check that your output B is correct, verify that $FFT(n, B, p, \omega^{-1}) = nA \mod p$.

(e) Let $a(x) = -x^3 + 3x + 1$ and $b(x) = 2x^4 - 3x^3 - 2x^2 + x + 1$ be polynomials in $\mathbb{Z}_{97}[x]$. Calculate the product of $c(x) = a(x)b(x)$ using the FFT.

If you could not get your FFT procedure from part (c) to work, use the following one which computes $[a(1), a(\omega), \ldots, a(\omega^{n-1})]$ using ordinary evaluation.

```
SlowFourierTransform := proc(n,A,p,omega)
local f,x,i,C,wi;
    f := add(A[i]*x^i, i=0..n-1);
    C := Array(0..n-1);
    wi := 1;
    for i from 0 to n-1 do
        C[i] := Eval(f,x=wi) mod p;
        wi := wi*omega mod p;
    od;
    return C;
end:
```

Question 3: The Modular GCD Algorithm (15 marks)

Consider the following pairs of polynomials in $\mathbb{Z}[x]$.

$$
\begin{aligned}
a_1 &= 58\,x^4 - 415\,x^3 - 111\,x + 213 \\
b_1 &= 69\,x^3 - 112\,x^2 + 413\,x + 113 \\
a_2 &= x^5 - 111\,x^4 + 112\,x^3 + 8\,x^2 - 888\,x + 896 \\
b_2 &= x^5 - 114\,x^4 + 448\,x^3 - 672\,x^2 + 669\,x - 336 \\
a_3 &= 396\,x^5 - 36\,x^4 + 3498\,x^3 - 2532\,x^2 + 2844\,x - 1870 \\
b_3 &= 156\,x^5 + 69\,x^4 + 1371\,x^3 - 332\,x^2 + 593\,x - 697
\end{aligned}
$$

Compute the GCD(a_i, b_i) using the modular GCD algorithm. Use primes $p = 23, 29, 31, 37, 43, \ldots$. Identify which primes are bad primes and which are unlucky primes.

To compute $\gcd(\phi_p(a), \phi_p(b))$ in Maple, use `Gcd(a,b) mod p` . Use the Maple commands `chrem` for Chinese remaindering, `mods` to put the coefficients in the symmetric range, and any other Maple commands that you need.

PLEASE make sure you input the polynomials correctly!

Question 4: Resultants (15 marks)

(a) Calculate the resultant of $A = 3x^2 + 3$ and $B = (x-2)(x+5)$ by hand.
Also, calculate the resultant using Maple. See `?resultant`

(b) Let A, B, C be non-constant polynomials in $R[x]$.
Show that $\mathrm{res}(A, BC) = \mathrm{res}(A, B) \cdot \mathrm{res}(A, C)$.

(c) Let A, B be two non-zero polynomials in $\mathbb{Z}[x]$. Let $A = G\bar{A}$ and $B = G\bar{B}$ where $G = \gcd(A, B)$.
Recall that a prime p in the modular gcd algorithm is unlucky iff $p|R$ where $R = \mathrm{res}(\bar{A}, \bar{B}) \in \mathbb{Z}$.
Consider the following pair of polynomials from question 2.

$$\bar{A} = 58\,x^4 - 415\,x^3 - 111\,x + 213$$

$$\bar{B} = 69\,x^3 - 112\,x^2 + 413\,x + 113$$

Using Maple, compute the resultant R and identify all unlucky primes. For each unlucky prime p compute `Gcd(`$\bar{A}, \bar{B}$`) mod p` in Maple to verify that the primes are indeed unlucky.

Question 5: Multivariate Polynomial Division (15 marks)

In assignment 2 question 2 we were given two polynomials $A, B \in \mathbb{Z}[x_1, x_2, \ldots, x_n]$ with $B \neq 0$, and I asked you to divide then by hand. The algorithm is recursive because we have to do divisions in $\mathbb{Z}[x_2, \ldots, x_n]$. Here is pseudo-code for the division.

Algorithm DIVIDE(A,B)
Input: $A, B \in \mathbb{Z}[x_1, x_2, \ldots, x_n]$ satisfying $B \neq 0$ and $n \geq 0$.
Output $Q \in \mathbb{Z}[x_1, x_2, \ldots, x_n]$ s.t. $A = BQ$ or FAIL meaning B does not divide A.

> **if** $n = 0$ **then**
> > **set** q and r to be the quotient and remainder of $A \div B$ in $\mathbb{Z}$
> > **if** $r = 0$ **then** output q **else** output FAIL **end if**
>
> **end if**
> **set** $Q = 0, R = A, N = \deg(A, x_1)$ and $M = \deg(B, x_1)$
> **while** $R \neq 0$ and $N \geq M$ **do**
> > **set** $LA = \mathrm{lc}(A, x_1)$ and $LB = \mathrm{lc}(B, x_1)$
> > **set** T = DIVIDE(LA,LB) // divide LA by LB in $\mathbb{Z}[x_2, \ldots, x_n]$
> > **if** $T = FAIL$ **then** output FAIL **end if**
> > **set** $T = T \times x_1^{N-M}$, $Q = Q + T$, $R = R - T \times B$ and $N = \deg(R, x_1)$
>
> **end while**
> **if** $R = 0$ **then** output Q **else** output FAIL **end if**

When one writes pseudo-code, one cannot "test it" so there is a chance that there is an error in it. I hope not. You are to implement this recursive multivariate division algorithm in Maple as the Maple procedure `DIVIDE(A,B)` to divide A by B. Test you program on the following inputs

```
> A := (6*y^2-5*y*z+z^2)*x^2+(7*y^2*z-3*y*z^2)*x+2*y^2*z^2;
> B := (2*y-z)*x+y*z;
> Q := DIVIDE(A,B);
```

```
> Q := DIVIDE(A+x,B);
> Q := DIVIDE(A+2,B);
> C := expand(A*B);
> Q := DIVIDE(C,B);
```

The following operations will be helpful.

```
> X := indets(A) union indets(B);  # set of all variables
```

$$X := \{x, y, z\}$$

```
> var := X[1];
```

$$var := x$$

```
> degree(B,var);
```

$$2$$

```
> lcoeff(B,var); # leading coefficient
```

$$2y - z$$

I suggest that you get your procedure working with zero variables first, then one variable, then two variables then three variables. If your procedure is not working you may insert a `print(...);` command anywhere in your procedure to print out any value. Also, you may trace the execution of your procedure by using

```
> trace(DIVIDE);
```

Maple will display everything that is computed. To turn tracing off use

```
> untrace(DIVIDE);
```