MACM 401/MATH 701/MATH 801
Assignment 3, Spring 2017.

Michael Monagan

Due Monday February 27th at 4pm.
Late Penalty: -20% for up to 48 hours late. Zero after that.
For problems involving Maple calculations and Maple programming, you should submit a printout of a Maple worksheet of your Maple session.

MATH 701 and 801 students should do all questions.
MACM 401 students should do questions 1-3 and either question 4 or 5.

Question 1: The Fast Fourier Transform (30 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 $p = 97 = 3 \times 2^5$, 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 would lead 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 transform 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) Using Maple's rsolve command, solve the following recurrences. Please simplify the output from rsolve. $T(1) = d, T(n) = 3T(n/2) + cn$ for $n > 1$ (Karatsuba), $T(1) = 0, T(n) = 2T(n/2) + n/2$ for $n > 1$ (optimized FFT) and $T(1) = 0, T(n) = T(n-1) + (n-1)^2$ for $n > 1$ (Gaussian elimination).

(e) Program the FFT in Maple as a recursive procedure. Your Maple procedure should take as input (n, A, p, w) 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 w a primitive n'th root of unity in $\mathbb{Z}_p$. If you want to precompute an array $W = [1, w, w^2, \ldots, w^{n/2-1}]$ of the powers of w 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 $FFT(n, B, p, w^{-1}) = nA \mod p$.

(f) 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 $[a(1), a(w), \ldots, a(w^{n-1})]$ using ordinary evaluation.

```
FT := proc(n,A,p,w)
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*w mod p;
    od;
    return C;
end:
```

Question 2: 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 $\mathrm{GCD}(a_i, b_i)$ via multiple modular mappings and Chinese remaindering. Use primes $p = 23, 29, 31, 37, 43, \ldots$. Identify which primes are bad primes, and which are unlucky primes. Use `Gcd(...) mod p` to compute a GCD modulo p in Maple and the Maple commands `chrem` to put the modular images together, `mods` to put the coefficients in the symmetric range, and `divide` for testing if the calculated GCD g_i divides a_i and b_i, and any others that you need.

PLEASE make sure you input the polynomials correctly!

Question 3: Resultants (15 marks)

(a) Calculate the resultant of $A = 3x^2 + 3$ and $B = (x - 2)(x + 5)$ by hand.

(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 4.

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

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

Using Maple, compute the resultant R and identify all unlucky primes. For each unlucky prime p compute the gcd of the polynomials $\bar{A}$ and $\bar{B}$ modulo p to verify that the primes are indeed unlucky.

Question 4: The Schwartz-Zippel Lemma (15 marks)

Let D be an integral domain and let $S \subset D$. Let f be a non-zero polynomial in $D[x_1, x_2, \ldots, x_n]$. If $\alpha_1, \alpha_2, \ldots, \alpha_n$ are chosen at random from S then

$$\text{Prob}[f(\alpha_1, \alpha_2, \ldots, \alpha_n) = 0] \leq \frac{\deg f}{|S|}.$$

Prove the Lemma by induction on n the number of variables.
Note, $\deg f$ is the total degree of f.

Question 5: The Chinese remainder theorem in $F[y]$ (15 marks).

Consider the problem of computing GCDs in $\mathbb{Z}_q[y][x]$, q a prime. If q is large then we can use evaluation and interpolation for y, i.e., we can evaluate at $y = 0, 1, 2, ...$ and interpolate the coefficients of the GCD in $\mathbb{Z}_q[y]$. If q is small, e.g. $q = 2$, this will not work as there will be insufficient evaluation points in $\mathbb{Z}_q$. Moreover, $y = 0$ and $y = 1$ may be bad or unlucky.

But $\mathbb{Z}_q[y]$ is a Euclidean domain and there are an infinite number of primes (irreducibles) in $\mathbb{Z}_q[y]$ which can play the role of integer primes in the modular GCD algorithm for computing GCDs in $\mathbb{Z}[x]$. For example, here are the irreducibles in $\mathbb{Z}_2[y]$ up to degree 4.

$$y, y+1, y^2+y+1, y^3+y+1, y^3+y^2+1, y^4+y+1, y^4+y^3+1, y^4+y^3+y^2+y+1.$$

To do this we need to solve the Chinese remainder problem in $\mathbb{Z}_q[y]$.

Theorem: Let F be any field (e.g. $\mathbb{Z}_q$) and let $m_1, m_2, \ldots, m_n$ and $u_1, u_2, \ldots, u_n$ be polynomials in $F[y]$ with $deg(m_i) > 0$ for $1 \leq i \leq n$. If $\gcd(m_i, m_j) = 1$ for $1 \leq i < j \leq n$ then there exists a unique polynomial u in $F[y]$ s.t.
(i) $u \equiv u_i \pmod{m_i}$ for $1 \leq i \leq n$ and
(ii) $u = 0$ or $\deg u < \sum_{i=1}^{n} \deg m_i$.

Prove the theorem by modifying the proof of the Chinese remainder theorem for $\mathbb{Z}$ to work for $F[y]$. Now solve the following Chinese remainder problem: find $u \in \mathbb{Z}_2[y]$ such that

$$u \equiv y^2 \pmod{y^3+y+1} \;\text{ and }\; u \equiv y^2+y+1 \pmod{y^3+y^2+1}.$$

Note, in the statement of the theorem the congruence relation $u \equiv u_i \pmod{m_i}$ means $m_i | (u - u_i)$ in $F[y]$. For the extended Euclidean algorithm in $\mathbb{Z}_q[y]$, use Maple's `Gcdex(...) mod q` command to compute the required inverse.