CS 487: Assignment #3 Due: Wed Mar 22, 2023 at 11:59pm

Submission Instruction: Submit your solutions for each question to Crowdmark before the due date and time. Question 7 require you to submit Maple code. A starter file `myGcdstarter.mpl` is provided to you that you should rename to `FirstLastnameA3.mpl`. Your submission to Crowdmark for the questions that involve Maple coding should include the result of executing the Maple commands, i.e., the plain text file `fileout` produce by running

```
maple < FirstLastnameA3.mpl > out
```

1. Consider the following polynomial from $\mathbb{Z}[x,y]$.

$$f = 3x^2y + 54y^2x - 6x^2 + 18y^2 - 216x - 72$$

   Considering $f \in \mathsf{R}[x]$ with $\mathsf{R} = \mathbb{Z}[y]$, express f as the product of $\text{cont}(f)$ and $\text{pp}(f)$.

2. Do question 1 again, but considering $f \in \mathsf{R}[y]$ with $\mathsf{R} = \mathbb{Z}[x]$.

3. If $f = 1/2 + x/3 + x^2/5 \in \mathbb{Q}[x]$, what is $\text{pp}(f)$?

4. Consider the ring $R = \mathbb{Z}[1/2] = \{a/2^n : a \in \mathbb{Z}, n \in \mathbb{N}\}$ of binary rationals.

   (a) What are the units of R?

   (b) You may use the fact that R is a UFD and that any two elements of R have a gcd which is unique up to associates. Find a normal form on R and use this to define a gcd function on R. Recall that a normal form has the following properties: two elements of R have the same normal form if and only if they are associates; the normal form of a product is equal to the products of the normal forms; for any $a \in R$, $a = \text{lu}(a) \cdot \text{normal}(a)$ where $\text{lu}(u)$ is the leading unit of a; $\text{lu}(0) = 1$ and $\text{normal}(0) = 0$.

   (c) Determine the content and primitive part of the polynomial $f = 2x^2 + 6x - 4$ with respect to the rings $\mathbb{Z}$ and R. Is f primitive with respect to R?

5. Let $\mathsf{R} = \mathbb{Q}[y]$ and suppose $f, g \in \mathsf{R}[x]$ both have $\deg_x = 10$ and $\deg_y = 6$, and suppose $h := \gcd(f,g)$ has $\deg_x h = 4$ and $\deg_y h = 2$. Derive an upper bound (as good as possible) on the number of distinct integers i such that $\gcd(f(x,i), g(x,i)) \in \mathbb{Q}[x]$ has degree not equal to 4.

6. Consider the polynomials $f = 3x^4 - 30x^3 + 105x^2 - 150x + 72$ and $g = 6x^4 + 3x^3 + ax^2 + 3x + 9$. Note that the coefficient of x^2 of g is given as an indeterminate a.

   (a) Considering $f, g \in \mathbb{Q}[x]$, find all values of $a \in \mathbb{Q}$ (if any) such that $\gcd(f,g) \in \mathbb{Q}[x]$ is nontrivial. *Hint:* You need to use Maple and do some computation. Include in your written solutions the output of your Maple computations.

   (b) Now consider $f, g \in \mathbb{Z}[x]$, and find all values of $a \in \mathbb{Z}$ (if any) such that $\gcd(f,g) \in \mathbb{Z}[x]$ is nontrivial. Note: the only trivial normalized gcd over $\mathbb{Z}[x]$ is 1.

7. A serious practical issue with the big prime modular gcd algorithm for $\mathbb{Z}[x]$ that is given in Script 9 is that it is based on Mignotte's bound (Theorem 9.17) which can be very pessimistic. If $f, g \in \mathbb{Z}[x]$ have degree bounded by n and coefficients bounded in magnitude by A, Algorithm `ModularGCD` uses the bound $(n+1)^{1/2}2^n A$ for the magnitude of coefficients in $\gcd(f,g)$. For example, if $n = 100$ and $A = 99$ (hence coefficients of f and g have length 2 decimal digits) the bound states that $\gcd(f,g)$ can have coefficients with more than $\log_{10}(101^{1/2}2^{100}(99)) > 132$ decimal digits. In practice, though, the gcd will typically have coefficients about the same size as the input polynomials.

   In this question you are to implement a deterministic version of the so-called small primes modular gcd algorithm for $\mathbb{Z}[x]$. Your implementation will use an *output sensitive* approach and be simpler than the algorithm from the textbook. Your procedure will consider each of the primes $2,3,5,7,\ldots$ in turn, computing $\gcd(f \bmod p, g \bmod p) \in \mathbb{Z}_p[x]$. These modular gcds are chinese remaindered together using incremental chinese remaindering. (A Maple input file `primer.mpl` is provided to you and gives examples of useful Maple functions in action, including a builtin incremental chinese remaindering algorithm.) You must detect so-called bad primes along the way, and continue the chinese remaindering until a termination condition is met. Note that your termination condition is checked every pass through the loop and so should be cheap, that is, be based on computing norms and not depend on multiplying together polynomials.

   A starter file `myGcdstarter.mpl` is provided to you. In addition to completing the procedure `myGcd`, also test your procedure on some example polynomials (of your construction). Note that according to the precondition the input polynomials may not be primitive; your algorithm should handle arbitrary inputs.

```
#  Pre: f, g in Z[x] with deg f >= deg g >= 1
# Post: gcd(f,g) in Z[x]
myGcd := proc(f,g,x)
   local p; # declare all local variables here
   # - preprocess to handle case when f and g not primitive
   # - initialize modular gcd algorithm and Chinese remaindering
   `mod` := mods;
   p := 1;
   do
      p := nextprime(p);
      # - determine if p should be avoided a priori
      # - if deg Gcd(f,g) mod p is inconsistent with previous images
      #   then react accordingly
      # - add current image to solution by incremental Chinese remaindering
      # - exit loop if termination condition is satisfied
   od;
   # return gcd, taking into account preprocessing (Norm check)
end:
```