

A. Electronic submission of a file (possibly scanned paper) of answers (answers only, not the detailed working), with your candidate number. For example, an answer to question 4a (if the polynomials were $x + 6$ and $x + 8$) might be “the g.c.d. is 1: 1 used two primes 2 and 3, but 2 was bad”.

B. Electronic submission of a Maple worksheet. The sheets should be named number-1.mw etc., so that a student whose examination number is 123456’s answer to question 3 would be called 123456-3.mw. In view of the fact that Maple’s ordering is session-dependent, the worksheets should be verified in a fresh instance of Maple 2021 before being submitted. Details on the electronic submission will be provided on Moodle. It is acceptable to provide Maple input .mpl files instead.

When I ask you, say, to run Bareiss’s algorithm, it is up to you whether you program it completely, run it by hand, or use a mixture of the two. It is not acceptable to use a built-in programme as part of your submission (though you may wish to use one to check your results!)¹. However, what you do must be comprehensible to an outsider (such as JHD), so programs must be commented/self-explanatory, and sets of commands must have explanations, e.g. against a line like

```
tmp:=r12:r12:=r13:r13:=tmp;
```

I would expect a comment such as

```
#swap rows 12 and 13 since a(12,12)=0
```

Note that

```
M:=SylvesterMatrix(A,B,x); # Sylvester Matrix of A and B with respect to x
```

is useless commenting: a comment should explain why the code is the way it is.

1. Run the sub-resultant algorithm (remembering that the point is that the sub-resultant algorithm doesn’t generate fractions) on the following two polynomials.

$$\begin{aligned} f &:= (y^2 - 1) [(y + 1)x^4 + (y^2 - 1)x^3 + (y^3 - 1)x^2 + (y^4 - 1)x + y^5 - 1]; \\ g &:= (y - 1)x^5 + (y^2 - 1)x^4 + (y^3 - 1)x^3 + (y^4 - 1)x^2 + (y^5 - 1)x + y^6 - 1 \end{aligned}$$

They should be regarded as polynomials in x whose coefficients are polynomials in y . What do you get as the last non-zero member of the subresultant sequence? What is the true greatest common divisor of these two polynomials? Any other greatest common divisors you require should be computed by subresultants.

- Functions that compute polynomial gcds, such as `content` (and its mirror `primpart`), `normal` or `simplify`, as well as the built-in `gcd` are not allowed. `factor` calls `gcd` internally, as do many other functions: if you don’t understand it, you’re probably safer not using it! As `prem` is tightly coupled to the cancellation properties of sub-resultant, you should not be using the built-in `prem` either.

- `igcd`, `rem`, `quo` etc. are legal.