

# Arithmetic: A Programmatic Approach

Murisi Tarusenga

Monday 30<sup>th</sup> April, 2018 19:59

## 0 Introduction

**What is this?** What follows is an experiment where I construct programs according to certain rules. While I do not list what these rules are, the following are a sketch of the sort of rules I have in mind:

1. The instruction "verify that  $a = a$ " is legal if it occurs after "choose an integer  $a$ "
2. The instruction "verify that  $b = a$ " is legal if it occurs after "verify that  $a = b$ "
3. The instruction "verify that  $a = c$ " is legal if it occurs after "verify that  $a = b$ " and "verify that  $b = c$ "
4. The instruction "verify that  $(a + b) + c = a + (b + c)$ " is legal if it occurs after "choose integers  $a, b, c$ "

**Why was this made?** I made this because I want to see whether programs constructed according to certain rules can show their own potential to achieve their objectives on different inputs. In other words, I want to see whether programs can be constructed in such a way as to render a correctness proof unnecessary.

**How do I understand this?** The task of understanding the following procedures should be the same as that of understanding any codebase. Hence domain specific knowledge is required, which in this case comprises rational, formal polynomial, and matrix arithmetic as well as inequalities. Otherwise, running a debugger, that is, executing the following procedures step by step on some chosen input(s) and observing their control flows and sequences of program states should be equally helpful in making sense of them.

## Contents

<b>0 Introduction</b>	<b>1</b>
<b>1 Integer Arithmetic</b>	<b>1</b>
<b>2 Rational Arithmetic</b>	<b>24</b>
<b>3 Matrix Arithmetic</b>	<b>29</b>
<b>4 References</b>	<b>62</b>

## 1 Integer Arithmetic

### Procedure 1.00

#### Objective

Choose an integer  $a$  and a positive integer  $b$ . The objective of the following instructions is to construct integers  $a \div b$  and  $a \bmod b$  such that  $a = (a \div b)b + a \bmod b$  and  $0 \leq a \bmod b < b$ .

#### Implementation

1. Let  $n = 0$ .
2. While  $(n + 1)b \leq a$ , do the following:
  - (a) Let  $n$  receive  $n + 1$ .
  - (b) Verify that  $nb \leq a$ .
3. While  $nb > a$ , do the following:
  - (a) Let  $n$  receive  $n - 1$ .
  - (b) Verify that  $(n + 1)b > a$ .
4. Therefore verify that  $nb \leq a$ .
5. Also verify that  $(n + 1)b > a$ .
6. Let  $a \div b = n$ .

7. Let  $a \bmod b = a - nb$ .
8. **Now verify that**  $b > a - nb = a \bmod b \geq 0$ .
9. **Also verify that**  $a = bn + a - nb = (a \div b)b + a \bmod b$ .
10. **Yield**  $\langle a \div b, a \bmod b \rangle$ .

### Notation 1.00

Let us use the notation  $a \div b$  as a shorthand for "the first part of the pair yielded by executing **procedure 1.00** on  $\langle a, b \rangle$ ".

### Notation 1.01

Let us use the notation  $a \bmod b$  as a shorthand for "the second part of the pair yielded by executing **procedure 1.00** on  $\langle a, b \rangle$ ".

### Notation 1.02

Let us use the notation  $a \equiv b \pmod{c}$  as a shorthand for " $a \bmod c = b \bmod c$ ".

## Procedure 1.01

### Objective

Choose four integers  $a, b, c, d$  and a positive integer  $e$  in such a way that  $a \equiv c \pmod{e}$  and  $b \equiv d \pmod{e}$ . The objective of the following instructions is to show that  $a + b \equiv c + d \pmod{e}$ .

### Implementation

1. Verify that  $a + b$ 
  - (a)  $\equiv (a \div e)e + (a \bmod e) + (b \div e)e + (b \bmod e)$
  - (b)  $\equiv (a \bmod e) + (b \bmod e)$
  - (c)  $\equiv (c \bmod e) + (d \bmod e)$
  - (d)  $\equiv (c \div e)e + (c \bmod e) + (d \div e)e + (d \bmod e)$
  - (e)  $\equiv c + d \pmod{e}$ .

## Procedure 1.02

### Objective

Choose four integers  $a, b, c, d$  and a positive integer  $e$  in such a way that  $a \equiv c \pmod{e}$  and  $b \equiv d \pmod{e}$ . The objective of the following instructions is to show that  $ab \equiv cd \pmod{e}$ .

### Implementation

1. Verify that  $ab$ 
  - (a)  $\equiv ((a \div e)e + (a \bmod e))((b \div e)e + (b \bmod e))$
  - (b)  $\equiv (a \div e)(b \div e)e^2 + (a \div e)(b \bmod e)e + (a \bmod e)(b \div e)e + (a \bmod e)(b \bmod e)$
  - (c)  $\equiv (a \bmod e)(b \bmod e)$
  - (d)  $\equiv (c \bmod e)(d \bmod e)$
  - (e)  $\equiv (c \div e)(d \div e)e^2 + (c \div e)(d \bmod e)e + (c \bmod e)(d \div e)e + (c \bmod e)(d \bmod e)$
  - (f)  $\equiv cd \pmod{e}$ .

## Procedure 1.03

### Objective

Choose an integer  $a$  and two positive integers  $b, c$ . The objective of the following instructions is to show that  $(a \bmod bc) \bmod b = a \bmod b$ .

### Implementation

1. **Verify that**  $(a \bmod bc) \bmod b = (a - (a \div bc)bc) \bmod b = a \bmod b$ .

## Procedure 1.04

### Objective

Choose a positive integer  $a$  and four integers  $b_1, b_0, c_1, c_0$  such that  $0 \leq b_0 < a$ ,  $0 \leq c_0 < a$ , and  $b_1a + b_0 = c_1a + c_0$ . The objective of the following instructions is to show that  $b_1 = c_1$  and  $b_0 = c_0$ .

### Implementation

1. **Verify that**  $b_0 = b_0 \bmod a = (b_1 a + b_0) \bmod a = (c_1 a + c_0) \bmod a = c_0 \bmod a = c_0$ .
2. Therefore verify that  $b_1 a = c_1 a$ .
3. **Therefore verify that**  $b_1 = c_1$ .

### Procedure 1.05

#### Objective

Choose an integer  $a$  and two positive integers  $b, c$ . The objective of the following instructions is to show that  $ca \bmod cb = c(a \bmod b)$  and that  $ca \div cb = a \div b$ .

### Implementation

1. Verify that  $bc(a \div b) + c(a \bmod b) = c(b(a \div b) + a \bmod b) = ca = cb(ca \div cb) + ca \bmod cb$ .
2. Now verify that  $0 \leq a \bmod b < b$ .
3. Therefore verify that  $0 \leq c(a \bmod b) < cb$ .
4. Now verify that  $0 \leq ca \bmod cb < cb$ .
5. Execute **procedure 1.04** on  $\langle bc, a \div b, c(a \bmod b), ca \div cb, ca \bmod cb \rangle$ .
6. **Therefore verify that**  $c(a \bmod b) = ca \bmod cb$ .
7. **Also verify that**  $a \div b = ca \div cb$ .

### Procedure 1.06

#### Objective

Choose two integers  $a, b$  and a positive integer  $c$  such that  $a \bmod c + b \bmod c < c$ . The objective of the following instructions is to show that  $a \div c + b \div c = (a + b) \div c$  and  $a \bmod c + b \bmod c = (a + b) \bmod c$ .

### Implementation

1. Verify that  $a = c(a \div c) + a \bmod c$ .
2. Verify that  $b = c(b \div c) + b \bmod c$ .

3. Therefore verify that  $a + b = c(a \div c + b \div c) + (a \bmod c + b \bmod c)$ .
4. Verify that  $0 \leq a \bmod c + b \bmod c < c$ .
5. Also verify that  $a + b = ((a + b) \div c)c + (a + b) \bmod c$ .
6. Verify that  $0 \leq (a + b) \bmod c < c$ .
7. Execute **procedure 1.04** on  $\langle c, a \div c + b \div c, a \bmod c + b \bmod c, (a + b) \div c, (a + b) \bmod c \rangle$ .
8. **Therefore verify that**  $a \div c + b \div c = (a + b) \div c$ .
9. **Also verify that**  $a \bmod c + b \bmod c = (a + b) \bmod c$ .

### Procedure 1.07

#### Objective

Choose two integers  $a, b$  and a positive integer  $c$  such that  $a \bmod c + b \bmod c \geq c$ . The objective of the following instructions is to show that  $1 + a \div c + b \div c = (a + b) \div c$  and  $a \bmod c + b \bmod c - c = (a + b) \bmod c$ .

### Implementation

1. Verify that  $a = c(a \div c) + a \bmod c$ .
2. Verify that  $b = c(b \div c) + b \bmod c$ .
3. Therefore verify that  $a + b = c(a \div c + b \div c) + a \bmod c + b \bmod c = c(1 + a \div c + b \div c) + (a \bmod c + b \bmod c - c)$ .
4. Verify that  $c \leq a \bmod c + b \bmod c < 2c$ .
5. Therefore verify that  $0 \leq a \bmod c + b \bmod c - c < c$ .
6. Also verify that  $a + b = c((a + b) \div c) + (a + b) \bmod c$ .
7. Verify that  $0 \leq (a + b) \bmod c < c$ .
8. Execute **procedure 1.04** on  $\langle c, 1 + a \div c + b \div c, a \bmod c + b \bmod c - c, (a + b) \div c, (a + b) \bmod c \rangle$ .
9. **Therefore verify that**  $1 + a \div c + b \div c = (a + b) \div c$ .

10. **Therefore verify that**  $a \bmod c + b \bmod c - c = (a + b) \bmod c$ .

$fa + gb = c$ , and if  $b = 0$ , then  $c = |a|$ , otherwise  $0 < c \leq b$ .

## Procedure 1.08

### Objective

Choose an integer  $a$  and two positive integers  $b, c$ . The objective of the following instructions is to show that  $a \div bc = (a \div b) \div c$  and  $a \bmod bc = ((a \div b) \bmod c)b + a \bmod b$ .

### Implementation

1. Verify that  $a = (a \div b)b + a \bmod b$ .
2. Verify that  $a \div b = ((a \div b) \div c)c + (a \div b) \bmod c$ .
3. Therefore verify that  $a = (((a \div b) \div c)c + (a \div b) \bmod c)b + a \bmod b = ((a \div b) \div c)bc + ((a \div b) \bmod c)b + a \bmod b$ .
4. Verify that  $0 \leq (a \div b) \bmod c \leq c - 1$ .
5. Therefore verify that  $0 \leq ((a \div b) \bmod c)b \leq cb - b$ .
6. Verify that  $0 \leq a \bmod b < b$ .
7. Therefore verify that  $0 \leq ((a \div b) \bmod c)b + a \bmod b < cb$ .
8. Now verify that  $a = (a \div bc)bc + a \bmod bc$ .
9. Verify that  $0 \leq a \bmod bc < bc$ .
10. Execute **procedure 1.04** on  $\langle bc, (a \div b) \div c, ((a \div b) \bmod c)b + a \bmod b, a \div bc, a \bmod bc \rangle$ .
11. **Therefore verify that**  $(a \div b) \div c = a \div bc$ .
12. **Also verify that**  $((a \div b) \bmod c)b + a \bmod b = a \bmod bc$ .

## Procedure 1.09

### Objective

Choose an integer  $a$  and a non-negative integer  $b$ . The objective of the following instructions is to construct integers  $c, d, e, f, g$  such that  $a = cd$ ,  $b = ce$ ,

### Implementation

1. If  $b = 0$ , then do the following:
  - (a) **Verify that**  $a = \text{sgn}(a)|a|$ .
  - (b) **Verify that**  $b = 0|a|$ .
  - (c) **Verify that**  $|a| = \text{sgn}(a)a + 0b$ .
  - (d) **Yield**  $\langle |a|, \text{sgn}(a), 0, \text{sgn}(a), 0 \rangle$ .
2. Otherwise do the following:
  - (a) Verify that  $0 \leq a \bmod b < b$ .
  - (b) Execute **procedure 1.09** on  $\langle b, a \bmod b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
  - (c) **Now verify that**  $b = cd$ .
  - (d) Also verify that  $a \bmod b = ce$ .
  - (e) **Therefore verify that**  $a = (a \div b)b + (a \bmod b) = c(d(a \div b) + e)$ .
  - (f) **Also verify that**  $(f - g(a \div b))b + ga = fb + g(a - (a \div b)b) = fb + g(a \bmod b) = c$ .
  - (g) If  $a \bmod b = 0$ , then do the following:
    - i. **Using (O), (2) and (b), verify that**  $0 < b = c \leq b$ .
  - (h) Otherwise do the following:
    - i. **Using (b), verify that**  $0 < c \leq a \bmod b < b$ .
  - (i) **Therefore yield**  $\langle c, d(a \div b) + e, d, g, f - g(a \div b) \rangle$ .

### Notation 1.03

Let us use the notation  $(a, b)$  as a shorthand for "the first part of the quintuple yielded by executing **procedure 1.09** on the pair  $\langle a, b \rangle$ ".

## Procedure 1.10

### Objective

Choose an integer  $a$  and a positive integer  $b$ . Let  $1 \leq c \leq b$  be the largest integer such that  $a \bmod c =$

$0$  and  $b \bmod c = 0$ . The objective of the following instructions is to either show that  $0 \neq 0$  or  $(a, b) = c$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. Verify that  $0 < d \leq b$ .
3. If  $d > c$ , then do the following:
  - (a) Using (O), verify that  $a \bmod d \neq 0$  or  $b \bmod d \neq 0$ .
  - (b) If  $a \bmod d \neq 0$ , then do the following:
    - i. Using (1), verify that  $a = ed$ .
    - ii. Therefore verify that  $a \bmod d = 0$ .
    - iii. **Therefore using (3b) and (3bii), verify that  $0 \neq 0$ .**
    - iv. **Abort procedure.**
  - (c) Otherwise if  $b \bmod d \neq 0$ , then do the following:
    - i. Using (1), verify that  $b = fd$ .
    - ii. Therefore verify that  $b \bmod d = 0$ .
    - iii. **Therefore using (3c) and (3cii), verify that  $0 \neq 0$ .**
    - iv. **Abort procedure.**
4. Otherwise if  $d < c$ , then do the following:
  - (a) Verify that  $ga + hb = d$ .
  - (b) Therefore verify that  $0 \equiv gc(a \operatorname{div} c) + hc(b \operatorname{div} c) = g(c(a \operatorname{div} c) + a \bmod c) + h(c(b \operatorname{div} c) + b \bmod c) = ga + hb = d \not\equiv 0(\bmod c)$ .
  - (c) **Therefore verify that  $0 \neq 0$ .**
  - (d) **Abort procedure.**
5. **Otherwise verify that  $(a, b) = d = c$ .**

### Procedure 1.11

#### Objective

Choose integers  $a, c, d, j$  and a non-negative integer  $b$ . Execute **procedure 1.09** on  $\langle a, b \rangle$  and let

$\langle e, f, g, h, i \rangle$  receive. The objective of the following instructions is to show that  $ca + db = (c + gj)a + (d - fj)b$ .

### Implementation

1. **Verify that  $(c + gj)a + (d - fj)b = ca + db + gja - fjb = ca + db + gje f - fje g = ca + db$ .**

### Procedure 1.12

#### Objective

Choose integers  $a, c, d$  and a non-negative integer  $b$  such that  $ca + db = (a, b)$ . Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle e, f, g, h, i \rangle$  receive. The objective of the following instructions is to construct a  $j$  such that  $c = h + gj$  and  $d = i - fj$ .

### Implementation

1. Verify that  $cef + deg = ca + db = (a, b) = e$ .
2. Therefore verify that  $cf + dg = 1$ .
3. Now verify that  $hef + ieg = ha + ib = e$ .
4. Therefore verify that  $hf + ig = 1$ .
5. Let  $j = ci - hd$ .
6. Now verify that  $cf = 1 - dg$ .
7. Therefore verify that  $c - cig = c(1 - ig) = chf = h(1 - dg) = h - hdg$ .
8. **Therefore verify that  $c = h + cig - hdg = h + g(ci - hd) = h + gj$ .**
9. Now verify that  $dg = 1 - cf$ .
10. Therefore verify that  $d - dhf = d(1 - hf) = dig = i(1 - cf) = i - icf$ .
11. **Therefore verify that  $d = i - icf + dhf = i - f(ic - dh) = i - fj$ .**
12. **Yield  $\langle j \rangle$ .**

## Procedure 1.13

### Objective

Choose an integer  $a$  and a positive integer  $b$  such that  $0 < (a, b) < b$ . The objective of the following instructions is to show that  $0 \neq 0$  or  $a \bmod b \neq 0$ .

### Implementation

1. If  $a \bmod b = 0$ , then do the following:
  - (a) Using (1), verify that  $af \equiv 0f \equiv 0(\bmod b)$ .
  - (b) Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
  - (c) Verify that  $0 < (a, b) = c = fa + gb < b$ .
  - (d) Therefore verify  $fa \equiv (a, b) \neq 0(\bmod b)$ .
  - (e) Therefore using (1a) and (1d), verify that  $0 \neq 0$ .
  - (f) **Abort ptocedure.**
2. **Otherwise verify that**  $a \bmod b \neq 0$ .

## Procedure 1.14

### Objective

Choose five integers  $a, d, e, f, g$  and two non-negative integers  $b, c$  such that  $a = cd$ ,  $b = ce$ , and  $fa + gb = c$ . The objective of the following instructions is to show that  $0 < 0$  or  $(a, b) = c$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle u, v, x, y, z \rangle$  receive.
2. Verify that  $u \geq 0$ .
3. Verify that  $a = uv$ .
4. Verify that  $b = xu$ .
5. Therefore verify that  $c = fa + gb = (fv + gx)u$ .
6. If  $u = 0$ , then do the following:
  - (a) **Verify that**  $c = (fv + gx)u = 0 = u = (a, b)$ .
  - (b) **Yield.**

7. Also using (1) and (O), verify that  $u = ya + zb = (yd + ze)c$ .
8. If  $c = 0$ , then do the following:
  - (a) **Verify that**  $(a, b) = u = (yd + ze)c = 0 = c$ .
  - (b) **Yield.**
9. Verify that  $c > 0$ .
10. Now verify that  $c = (fv + gx)u = (fv + gx)(yd + ze)c$ .
11. Therefore verify that  $(fv + gx)(yd + ze) = 1$ .
12. Therefore verify that  $fv + gx = yd + ze = \pm 1$ .
13. If  $fv + gx = yd + ze = -1$ , then do the following:
  - (a) Using (7) and (9), verify that  $u = (yd + ze)c = -c < 0$ .
  - (b) **Therefore using (2) and (13a), verify that**  $0 \leq u < 0$ .
  - (c) **Abort procedure.**
14. Otherwise, do the following:
  - (a) Verify that  $fv + gx = yd + ze = 1$ .
  - (b) **Therefore verify that**  $c = (fv + gx)u = u = (a, b)$ .

## Procedure 1.15

### Objective

Choose an integer  $a$  and a non-negative integer  $b$ . The objective of the following instructions is to show that  $0 < 0$  or  $(a, b) = (-a, b)$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
2. Verify that  $a = dc$ .
3. Therefore verify that  $-a = (-d)c$ .
4. Verify that  $b = ec$ .
5. Verify that  $fa + gb = c$ .
6. Therefore verify that  $(-f)(-a) + gb = c$ .
7. Execute **procedure 1.14** on  $\langle -a, b, c, -d, e, -f, g \rangle$ .

8. Therefore verify that  $(-a, b) = c = (a, b)$ .

### Procedure 1.16

#### Objective

Choose two non-negative integers  $a, b$ . The objective of the following instructions is to show that  $0 < 0$  or  $(a, b) = (b, a)$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
2. Verify that  $b = ec$ .
3. Verify that  $a = dc$ .
4. Verify that  $gb + fa = c$ .
5. Execute **procedure 1.14** on  $\langle b, a, c, e, d, g, f \rangle$ .
6. Therefore verify that  $(b, a) = c = (a, b)$ .

### Procedure 1.17

#### Objective

Choose two integers  $a, b$  and a positive integer  $c$  such that  $a \equiv b \pmod{c}$ . The objective of the following instructions is to show that  $0 < 0$  or  $(a, c) = (b, c)$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, c \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. Verify that  $a = ed$ .
3. Verify that  $c = fd$ .
4. Let  $j = b \text{ div } c - a \text{ div } c$ .
5. Therefore verify that  $b = a + jc = ed + jfd = (e + jf)d$ .
6. Verify that  $gb + (h - gj)c = g(a + jc) + (h - gj)c = ga + hc = d$ .
7. Now execute **procedure 1.14** on  $\langle b, c, d, e + jf, f, g, h - gj \rangle$ .
8. Therefore verify that  $(b, c) = d = (a, c)$ .

### Procedure 1.18

#### Objective

Choose an integer  $a$  and two non-negative integers  $b, c$ . The objective of the following instructions is to show that either  $0 < 0$  or  $(ca, cb) = c(a, b)$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. Verify that  $a = ed$ .
3. Therefore verify that  $ca = e(cd)$ .
4. Verify that  $b = df$ .
5. Therefore verify that  $cb = f(cd)$ .
6. Verify that  $ga + hb = d$ .
7. Therefore verify that  $g(ca) + h(cb) = cd$ .
8. Now execute **procedure 1.14** on  $\langle ca, cb, cd, e, f, g, h \rangle$ .
9. Therefore verify that  $(ca, cb) = cd = c(a, b)$ .

### Procedure 1.19

#### Objective

Choose an integer  $a$  and two non-negative integers  $b, c$ . The objective of the following instructions is to show that either  $0 < 0$  or  $(a, (b, c)) = ((a, b), c)$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle d_0, e_0, f_0, g_0, h_0 \rangle$  receive.
2. Execute **procedure 1.09** on  $\langle b, c \rangle$  and let  $\langle d_1, e_1, f_1, g_1, h_1 \rangle$  receive.
3. Execute **procedure 1.09** on  $\langle (a, b), c \rangle$  and let  $\langle d_2, e_2, f_2, g_2, h_2 \rangle$  receive.
4. Verify that  $a = d_0e_0 = e_0(a, b) = e_0d_2e_2 = e_0e_2((a, b), c)$ .
5. Verify that  $(b, c)$   
 $(a) = g_1b + h_1c$

- (b) =  $g_1d_0f_0 + h_1d_2f_2$
- (c) =  $g_1f_0(a, b) + h_1f_2((a, b), c)$
- (d) =  $g_1f_0d_2e_2 + h_1f_2((a, b), c)$
- (e) =  $g_1f_0e_2((a, b), c) + h_1f_2((a, b), c)$
- (f) =  $(g_1f_0e_2 + h_1f_2)((a, b), c)$ .

6. Verify that  $((a, b), c)$

- (a) =  $d_2$
- (b) =  $g_2(a, b) + h_2c$
- (c) =  $g_2d_0 + h_2d_1f_1$
- (d) =  $g_2(g_0a + h_0b) + h_2f_1(b, c)$
- (e) =  $g_2g_0a + g_2h_0d_1e_1 + h_2f_1(b, c)$
- (f) =  $g_2g_0a + g_2h_0e_1(b, c) + h_2f_1(b, c)$
- (g) =  $g_2g_0a + (g_2h_0e_1 + h_2f_1)(b, c)$ .

7. Execute **procedure 1.14** on  $\langle a, (b, c), ((a, b), c), e_0e_2, g_1f_0e_2 + h_1f_2, g_2g_0, g_2h_0e_1 + h_2f_1 \rangle$ .

8. **Therefore verify that**  $((a, b), c) = (a, (b, c))$ .

#### Notation 1.04

Let us use the notation  $(a_1, a_2, \dots, a_n)$  as a shorthand for "either  $(a_1, (a_2, a_3, \dots, a_n))$  or  $((a_1, a_2), (a_3, a_4, \dots, a_n))$  or  $\dots$  or  $((a_1, a_2, \dots, a_{n-1}), a_n)$ ".

#### Procedure 1.20

##### Objective

Choose two integers  $a, b$  and a non-negative integer  $c$  such that  $(a, c) = 1$  and  $(b, c) = 1$ . The objective of the following instructions is to show that either  $0 < 0$  or  $(ab, c) = 1$ .

##### Implementation

1. Execute **procedure 1.09** on  $\langle a, c \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. Verify that  $ga + hc = d = (a, c) = 1$ .
3. Execute **procedure 1.09** on  $\langle b, c \rangle$  and let  $\langle t, u, v, w, x \rangle$  receive.

4. Verify that  $wb + xc = t = (b, c) = 1$ .
5. Therefore verify that  $(gw)(ab) + (gax + wbh + hxc)c = (ga + hc)(wb + xc) = 1$ .
6. Now execute **procedure 1.14** on  $\langle ab, c, 1, ab, c, gw, gax + wbh + hxc \rangle$ .
7. **Therefore verify that**  $(ab, c) = 1$ .

#### Procedure 1.21

##### Objective

Choose an integer  $a$  and two non-negative integers  $b, c$  such that  $(a, bc) = 1$ . The objective of the following instructions is to show that either  $0 < 0$  or  $(a, b) = 1$ .

##### Implementation

1. Execute **procedure 1.09** on  $\langle a, bc \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. Verify that  $d = (a, bc) = 1$ .
3. Verify that  $ga + (hc)b = ga + h(bc) = d = 1$ .
4. Now execute **procedure 1.14** on  $\langle a, b, 1, a, b, g, hc \rangle$ .
5. **Therefore verify that**  $(a, b) = 1$ .

#### Notation 1.05

Let us use the notation " $a$  is prime" as a shorthand for " $a > 1$  and  $a \bmod k \neq 0$  for  $1 < k < a$ ".

#### Procedure 1.22

##### Objective

Choose an integer  $a$  and a prime  $b$  such that  $a \bmod b \neq 0$ . The objective of the following instructions is to show that either  $0 \neq 0$  or  $(a, b) = 1$ .

##### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
2. Verify that  $0 < c \leq b$ .



3. If  $c = b$ , then do the following:
  - (a) Verify that  $a = cd = bd$ .
  - (b) Therefore verify that  $a \bmod b = 0$ .
  - (c) **Therefore using (O) and (3b), verify that  $0 \neq 0$ .**
  - (d) **Abort procedure.**
4. Otherwise if  $1 < c < b$ , then do the following:
  - (a) Verify that  $b = ce$ .
  - (b) Therefore verify that  $b \bmod c = 0$ .
  - (c) **Therefore using (O) and (4b), verify that  $0 \neq 0$ .**
  - (d) **Abort procedure.**
5. Otherwise, do the following:
  - (a) **Verify that  $(a, b) = c = 1$ .**

## Procedure 1.23

### Objective

Choose two integers  $a, b$  and a prime  $c$  such that  $a \bmod c \neq 0$  and  $b \bmod c \neq 0$ . The objective of the following instructions is to show that either  $0 \neq 0$  or  $ab \bmod c \neq 0$ .

### Implementation

1. Execute **procedure 1.22** on  $\langle a, c \rangle$ .
2. Verify that  $(a, c) = 1$ .
3. Execute **procedure 1.22** on  $\langle b, c \rangle$ .
4. Verify that  $(b, c) = 1$ .
5. Execute **procedure 1.20** on  $\langle a, b, c \rangle$ .
6. Now verify that  $0 < (ab, c) = 1 < c$ .
7. Execute **procedure 1.13** on  $\langle ab, c \rangle$ .
8. **Now verify that  $ab \bmod c \neq 0$ .**

### Notation 1.06

Let us use the notation  $|A|$  as a shorthand for "the number of items in the list  $A$ ".

### Notation 1.07

Let us use the notation  $a_*$  as a shorthand for " $1 \cdot a_1 a_2 \cdots a_{|a|}$ ".

### Notation 1.08

Let us use the notation  $A \frown B$  as a shorthand for "the list formed by concatenating  $B$  onto  $A$ ".

## Procedure 1.24

### Objective

Choose a positive integer  $a$ . The objective of the following instructions is to construct a list of prime numbers  $b$  such that  $a = b_*$ .

### Implementation

1. If  $a = 1$ , then do the following:
  - (a) Verify that  $a = 1 = \langle \rangle_*$ .
  - (b) Therefore yield  $\langle \rangle$ .
2. Otherwise, do the following:
  - (a) Verify that  $a > 1$ .
  - (b) For  $c = 2$  up to  $c = a - 1$ , do the following:
    - i. If  $a \bmod c = 0$ , then do the following:
      - A. Verify that  $a = (a \operatorname{div} c)c$ .
      - B. Therefore verify that  $1 < a \operatorname{div} c < a$ .
      - C. Execute **procedure 1.24** on  $\langle a \operatorname{div} c \rangle$  and let  $\langle d \rangle$  receive.
      - D. Using (B) and (C), verify that  $|d| > 0$ .
      - E. Verify that every element of  $d$  is prime.
      - F. Verify that  $a \operatorname{div} c = d_*$ .
      - G. Execute **procedure 1.24** on  $\langle c \rangle$  and let  $\langle e \rangle$  receive.
      - H. Using (b) and (G), verify that  $|e| > 0$ .
      - I. Verify that every element of  $e$  is prime.
      - J. Verify that  $c = e_*$ .
      - K. **Therefore verify that  $|d \frown e| > 0$ .**

- L. **Also verify that every element of  $d \frown e$  is prime.**
- M. **Also verify that  $a = (a \operatorname{div} c)c = d_*e_* = (d \frown e)_*$ .**
- N. **Yield  $\langle d \frown e \rangle$ .**
- (c) Otherwise do the following:
  - i. **Verify that  $a$  is prime.**
  - ii. **Yield  $\langle a \rangle$ .**
- ii. Now verify that  $b_k \bmod a = 0$ .
- iii. **Hence using (O) and (ii), verify that  $0 \neq b_k \bmod a = 0$ .**
- iv. **Abort procedure.**
- (e) Otherwise do the following:
  - i. Verify that  $b_k \operatorname{div} a = 1$ .
  - ii. **Therefore verify that  $b_k = a$ .**
  - iii. **Yield  $\langle k \rangle$ .**

## Procedure 1.25

### Objective

Choose a prime  $a$  and a list of primes  $b$  such that  $b_* \equiv 0 \pmod{a}$ . The objective of the following instructions is to either show that  $0 = 1$  or to construct a  $k$  such that  $a = b_k$ .

### Implementation

1. Using (O), verify that  $a > 1$ .
2. If  $|b| = 0$ , then do the following:
  - (a) Verify that  $1 = b_* \equiv 0 \pmod{a_1}$ .
  - (b) **Therefore using (1) and (a), verify that  $0 = 1$ .**
  - (c) **Abort procedure.**
3. Otherwise if  $b_j \bmod a \neq 0$  for  $j = 1$  to  $n$ , then do the following:
  - (a) Using **procedure 1.23**, verify that  $b_* \not\equiv 0 \pmod{a}$ .
  - (b) **Therefore using (O) and (a), verify that  $0 \neq 0$ .**
  - (c) **Abort procedure.**
4. Otherwise do the following:
  - (a) Let  $k$  be such that  $b_k \bmod a_1 = 0$ .
  - (b) Verify that  $b_k = (b_k \operatorname{div} a)a_1$ .
  - (c) Verify that  $b_k \operatorname{div} a \geq 1$ .
  - (d) If  $b_k \operatorname{div} a > 1$ , then do the following:
    - i. Using (1),(b), and (d), verify that  $1 < a < b_k$ .

## Notation 1.09

Let us use the notation  $[a : b]$  as a shorthand for "the list  $\langle a, a + 1, \dots, b - 1 \rangle$ ".

## Procedure 1.26

### Objective

Choose two lists of primes  $a, b$  such that  $a_* = b_*$ . The objective of the following instructions is to show that either  $1 > 1$  or  $a$  is included in  $b$ .

### Implementation

1. If  $|a| = 0$ , then do the following:
  - (a) **Verify that  $a$  is included in  $b$ .**
2. Otherwise, do the following:
  - (a) Verify that  $|a| > 0$ .
  - (b) Verify that  $b_* \equiv a_* \equiv 0 \pmod{a_1}$ .
  - (c) Execute **procedure 1.25** on  $\langle a_1, b \rangle$  and let  $\langle k \rangle$  receive.
  - (d) Therefore verify that  $b_k = a_1$ .
  - (e) Now verify  $(a_{[2:|a|+1]})_* = (b_{[1:k] \frown [k+1:|b|+1]})_*$ .
  - (f) Now execute **procedure 1.26** on  $\langle a_{[2:|a|+1]}, b_{[1:k] \frown [k+1:|b|+1]} \rangle$ .
  - (g) Now verify that  $a_{[2:|a|+1]}$  is included in  $b_{[1:k] \frown [k+1:|b|+1]}$ .
  - (h) **Therefore verify that  $a$  is included in  $b$ .**

## Procedure 1.27

### Objective

Choose two lists of primes  $a, b$  such that  $a_* = b_*$ . The objective of the following instructions is to show that either  $1 > 1$  or  $a$  is a rearrangement of  $b$ .

### Implementation

1. Execute **procedure 1.26** on  $\langle a, b \rangle$ .
2. Verify that  $a$  is included in  $b$ .
3. Execute **procedure 1.26** on  $\langle b, a \rangle$ .
4. Verify that  $b$  is included in  $a$ .
5. **Therefore verify that  $a$  is a rearrangement of  $b$ .**

## Procedure 1.28

### Objective

Choose a positive integer  $a$ . The objective of the following instructions is to construct a prime  $b$  such that  $b > a$  and  $c$  is not prime for  $c = a + 1$  up to  $c = b - 1$ .

### Implementation

1. Verify that  $a! + 1 > 1$ .
2. Execute **procedure 1.24** on  $\langle a! + 1 \rangle$  and let  $\langle d \rangle$  receive.
3. Therefore using (1) and (2), verify that  $|d| > 0$ .
4. Now verify that  $(a! + 1) \bmod d_1 = 0$ .
5. For  $e = 2$  up to  $e = a$ , do the following:
  - (a) Verify that  $a! + 1 \equiv 1 \pmod{e}$ .
  - (b) If  $e = d_1$ , then do the following:
    - i. Using (4) and (a), verify that  $0 \equiv a! + 1 \equiv 1 \pmod{e}$ .
    - ii. **Therefore verify that  $0 = 1$ .**
    - iii. **Abort procedure.**
6. Otherwise do the following:
  - (a) **Using (2), verify that  $d_1$  is prime.**

- (b) Using (a), verify that  $d_1 > 1$ .
- (c) **Using (a) and (5), verify that  $d_1 > a$ .**
- (d) **Let  $b$  be the least prime between  $a + 1$  and  $d_1$ .**
- (e) **Yield  $\langle b \rangle$ .**

## Procedure 1.29

### Objective

Choose a positive integer  $a$ . The objective of the following instructions is to construct a positive integer  $b$  such that none of  $b + 1, b + 2, \dots, b + (a - 1)$  are prime.

### Implementation

1. Let  $b = a! + 1$ .
2. For  $i = 1$  up to  $i = a - 1$ , do the following:
  - (a) Verify that  $b + i = a! + 1 + i = i!(i + 1)(i + 2) \cdots (a + 1 + i) = (1 + i)(i!(i + 2)(i + 3) \cdots (a + 1))$ .
  - (b) Therefore verify that  $b + i \equiv 0 \pmod{i + 1}$ .
  - (c) Also verify that  $b + i = a! + 1 + i > a! \geq a \geq i + 1 > 1$ .
  - (d) **Therefore verify that  $b + i$  is not prime.**
3. **Yield  $\langle b \rangle$ .**

## Procedure 1.30

### Objective

Choose two lists of primes  $a, b$  in such a way that their intersection is empty. The objective of the following instructions is to show that  $0 = 1$  or  $(a_*, b_*) = 1$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle a_*, b_* \rangle$  and let  $\langle c, d, e, f, g \rangle$ .
2. Verify that  $0 < c \leq b$ .
3. If  $c > 1$ , then do the following:

- (a) Execute **procedure 1.24** on  $\langle c \rangle$  and let  $\langle h \rangle$  receive.
  - (b) Using (3) and (a), verify that  $|h| > 0$ .
  - (c) Now verify that  $a_* = dc = dh_* = dh_1(h_{[2:|h|+1]})_* \equiv 0 \pmod{h_1}$ .
  - (d) Execute **procedure 1.25** on  $\langle h_1, a \rangle$  and let  $\langle k \rangle$  receive.
  - (e) Now verify that  $b_* = ec = eh_* = eh_1(h_{[2:|h|+1]})_* \equiv 0 \pmod{h_1}$ .
  - (f) Execute **procedure 1.25** on  $\langle h_1, b \rangle$  and let  $\langle m \rangle$  receive.
  - (g) **Therefore verify that**  $a_k = b_m$ .
  - (h) **Abort procedure.**
4. Otherwise do the following:
- (a) **Verify that**  $(a_*, b_*) = c = 1$ .

### Procedure 1.31

#### Objective

Choose two lists of primes  $a, b$ . Let  $c$  be the common sublist with multiplicity of  $a$  and  $b$ . The objective of the following instructions is to show that either  $0 < 0$  or  $(a_*, b_*) = c_*$ .

#### Implementation

1. Let  $d$  be the result of removing with multiplicity elements of  $c$  from  $a$ .
2. Verify that  $a_* = c_*d_*$ .
3. Let  $e$  be the result of removing with multiplicity elements of  $c$  from  $b$ .
4. Verify that  $b_* = c_*e_*$ .
5. Verify that  $d$  and  $e$  share no common elements.
6. **Therefore using procedure 1.18 and procedure 1.30, verify that**  $(a_*, b_*) = (c_*d_*, c_*e_*) = c_*(d_*, e_*) = c_*$ .

### Procedure 1.32

#### Objective

Choose an integer  $a$  and a positive integer  $b$ . The objective of the following instructions is to construct integers  $c, f, e$  such that  $c = af$ ,  $c = be$ ,  $c(a, b) = ab$ , and  $|a| \leq c \operatorname{sgn}(a) \leq |a|b$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle d, e, f, g, h \rangle$  receive.
2. **Let**  $c = af$ .
3. **Verify that**  $c(a, b) = cd = afd = ab$ .
4. Verify that  $d > 0$ .
5. Verify that  $b = fd$ .
6. Therefore verify that  $1 \leq f \leq b$ .
7. Therefore verify that  $|a| \leq |a|f \leq |a|b$ .
8. **Therefore verify that**  $|a| \leq c \operatorname{sgn}(a) \leq |a|b$ .
9. **Verify that**  $c = af = def = be$ .
10. **Yield the tuple**  $\langle c, f, e \rangle$ .

#### Notation 1.10

Let us use the notation  $[a, b]$  as a shorthand for "the first part of the triple yielded by executing **procedure 1.32** on  $\langle a, b \rangle$ ".

### Procedure 1.33

#### Objective

Choose two positive integers  $a, b$ . The objective of the following instructions is to show that  $[a, b] = [b, a]$ .

#### Implementation

1. Verify that  $(a, b) > 0$ .
2. Verify that  $[a, b](a, b) = ab = ba = [b, a](b, a) = [b, a](a, b)$ .
3. **Therefore verify that**  $[a, b] = [b, a]$ .

### Procedure 1.34

#### Objective

Choose an integer  $a$  and two positive integers  $b, c$ . The objective of the following instructions is to show that  $[ca, cb] = c[a, b]$ .

#### Implementation

1. Verify that  $(ca, cb) > 0$ .
2. Using **procedure 1.18**, verify that  $[ca, cb](ca, cb) = cacb = c^2ab = c^2[a, b](a, b) = c[a, b](ca, cb)$ .
3. **Therefore verify that**  $[ca, cb] = c[a, b]$ .

### Procedure 1.35

#### Objective

Choose an integer  $a$  and two positive integers  $b, c$ . The objective of the following instructions is to show that  $[[a, b], c] = [a, [b, c]]$ .

#### Implementation

1. Verify that  $(a, b)(ab, (ac, bc))(b, c)[[a, b], c]$ 
  - (a)  $= (ab, (ac, bc))(b, c)[(a, b)[a, b], (a, b)c]$
  - (b)  $= (ab, (ac, bc))(b, c)[ab, (ac, bc)]$
  - (c)  $= ab(ac, bc)(b, c)$
  - (d)  $= abc(a, b)(b, c)$
  - (e)  $= bc(a, b)(ab, ac)$
  - (f)  $= (a, b)((ab, ac), bc)[(ab, ac), bc]$
  - (g)  $= (a, b)(ab, (ac, bc))[(ab, ac), bc]$
  - (h)  $= (a, b)(ab, (ac, bc))[a(b, c), [b, c](b, c)]$
  - (i)  $= (a, b)(ab, (ac, bc))(b, c)[a, [b, c]]$ .
2. Verify that  $(a, b)(ab, (ac, bc))(b, c) > 0$ .
3. **Therefore verify that**  $[[a, b], c] = [a, [b, c]]$ .

### Notation 1.11

Let us use the notation  $[a_1, a_2, \dots, a_n]$  as a shorthand for "either  $[a_1, [a_2, a_3, \dots, a_n]]$  or  $[[a_1, a_2], [a_3, a_4, \dots, a_n]]$  or  $\dots$  or  $[[a_1, a_2, \dots, a_{n-1}], a_n]$ ".

### Procedure 1.36

#### Objective

Choose three positive integers  $a, b, c$ . The objective of the following instructions is to show that  $([a, b], c) = [(a, c), (b, c)]$ .

#### Implementation

1. Using **procedure 1.32**, **procedure 1.18**, **procedure 1.19**, **procedure 1.16**, and **procedure 1.10**, verify that  $(a, b)((a, c), (b, c))([a, b], c)$ 
  - (a)  $= ((a, c), (b, c))((a, b)[a, b], (a, b)c)$
  - (b)  $= ((a, c), (b, c))(ab, (ac, bc))$
  - (c)  $= (a^2b, a^2c, c^2a, c^2b, b^2a, bac, b^2c)$
  - (d)  $= (a, b)(ab, ac, bc, c^2)$
  - (e)  $= (a, b)(a, c)(b, c)$
  - (f)  $= (a, b)((a, c), (b, c))[(a, c), (b, c)]$ .
2. Verify that  $(a, b)((a, c), (b, c)) > 0$ .
3. **Therefore verify that**  $([a, b], c) = [(a, c), (b, c)]$ .

### Procedure 1.37

#### Objective

Choose three positive integers  $a, b, c$ . The objective of the following instructions is to show that  $[(a, b), c] = ([a, c], [b, c])$ .

#### Implementation

1. Using **procedure 1.32**, **procedure 1.18**, **procedure 1.19**, **procedure 1.16**, and **procedure 1.10**, verify that  $((a, b), c)(a, c)(b, c)[(a, b), c]$ 
  - (a)  $= (a, c)(b, c)(a, b)c$

- (b) =  $(ab, ac, cb, c^2)(a, b)c$   
(c) =  $(a^2b, a^2c, ac^2, ab^2, abc, cb^2, bc^2)c$   
(d) =  $(a, b, c)(ab, ac, bc)c$   
(e) =  $((a, b), c)(ac(b, c), bc(a, c))$   
(f) =  $((a, b), c)(a, c)(b, c)([a, c], [b, c])$ .
2. Verify that  $((a, b), c)(a, c)(b, c) > 0$ .
3. **Therefore verify that**  $[(a, b), c] = [[a, c], [b, c]]$ .

### Procedure 1.38

#### Objective

Choose two integers  $a, c$  and two positive integers  $b, d$  in such a way that  $a \equiv c \pmod{(b, d)}$ . The objective of the following instructions is to construct an integer  $0 \leq e < [b, d]$ .

#### Implementation

1. Execute **procedure 1.09** on  $\langle b, d \rangle$  and let  $\langle f, g, h, i, j \rangle$  receive.
2. **Yield the tuple**  $\langle (a + ((c - a) \operatorname{div}(b, d))ib) \bmod [b, d] \rangle$ .

### Notation 1.12

Let us use the notation  $\chi_{b,d}(a, c)$  as a shorthand for "the result yielded by executing **procedure 1.38** on  $\langle a, c, b, d \rangle$ ".

### Procedure 1.39

#### Objective

Choose three integers  $x, a, c$  and two positive integers  $b, d$  such that  $x \equiv a \pmod{b}$  and  $x \equiv c \pmod{d}$ . The objective of the following instructions is to show that  $0 \neq 0$  if  $a \not\equiv c \pmod{(b, d)}$ , otherwise  $x \equiv \chi_{b,d}(a, c) \pmod{[b, d]}$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle b, d \rangle$  and let  $\langle e, f, g, h, i \rangle$  receive.
2. Let  $j = x \operatorname{div} b - a \operatorname{div} b$ .
3. Verify that  $x = a + jb$ .
4. Let  $k = x \operatorname{div} d - c \operatorname{div} d$ .
5. Verify that  $x = c + kd$ .
6. **Therefore verify that**  $c - a = jb - kd$ .
7. If  $a \not\equiv c \pmod{(b, d)}$ , then do the following:
  - (a) Verify that  $0 \neq d - a = jb - kd = jef - keg \equiv 0 \pmod{e}$ .
  - (b) **Therefore verify that**  $0 \neq 0$ .
  - (c) **Abort procedure.**
8. Otherwise do the following:
  - (a) Verify that  $c - a \equiv 0 \pmod{(b, d)}$ .
  - (b) Let  $l = (c - a) \operatorname{div}(b, d)$ .
  - (c) Verify that  $l(b, d) = le = c - a = jb - kd = jef - keg$ .
  - (d) **Therefore verify that**  $l = jf - kg$ .
  - (e) **Therefore verify that**  $l \equiv jf \pmod{g}$ .
  - (f) Also, using (1) verify that  $efh + egi = bh + di = e$ .
  - (g) **Therefore verify that**  $fh + gi = 1$ .
  - (h) **Therefore verify that**  $fh \equiv 1 \pmod{g}$ .
  - (i) **Therefore verify that**  $lh \equiv jfh \equiv j \pmod{g}$ .
  - (j) **Therefore using procedure 1.05**, verify that  $lhb \equiv jb \pmod{bg = [b, d]}$ .
  - (k) **Therefore verify that**  $x \equiv a + jb \equiv a + lhb \equiv \chi_{b,d}(a, c) \pmod{[b, d]}$ .

### Procedure 1.40

#### Objective

Choose two integers  $a, c$  and two positive integers  $b, d$  in such a way that  $a \equiv c \pmod{(b, d)}$ . The objective of the following instructions is to show that  $\chi_{b,d}(a, c) = \chi_{d,b}(c, a)$ .

## Implementation

1. Execute **procedure 1.09** on  $\langle b, d \rangle$  and let  $\langle f, g, h, i, j \rangle$  receive.
2. Verify that  $ib + jd = f = (b, d)$ .
3. Execute **procedure 1.09** on  $\langle d, b \rangle$  and let  $\langle k, l, m, n, p \rangle$  receive.
4. Verify that  $pb + nd = k = (d, b) = (b, d)$ .
5. Execute **procedure 1.12** on  $\langle b, p, n, d \rangle$  and let  $\langle q \rangle$  receive.
6. Therefore verify that  $n = j - qg$ .
7. Verify that  $\chi_{b,d}(a, c)$ 
  - (a)  $= (a + ((c - a) \operatorname{div}(b, d))ib) \bmod [b, d]$
  - (b)  $= (a + ((c - a) \operatorname{div}(b, d))(f - jd)) \bmod [b, d]$
  - (c)  $= (a + ((c - a) \operatorname{div}(b, d))f + ((a - c) \operatorname{div}(b, d))jd) \bmod [b, d]$
  - (d)  $= (a + (c - a) + ((a - c) \operatorname{div}(b, d))jd) \bmod [b, d]$
  - (e)  $= (c + ((a - c) \operatorname{div}(d, b))(n + qg)d) \bmod [b, d]$
  - (f)  $= (c + ((a - c) \operatorname{div}(d, b))dn + ((a - c) \operatorname{div}(d, b))q[b, d]) \bmod [b, d]$
  - (g)  $= (c + ((a - c) \operatorname{div}(d, b))dn) \bmod [b, d]$
  - (h)  $= \chi_{d,b}(c, a)$ .

## Procedure 1.41

### Objective

Choose three integers  $x, a, c$  and two positive integers  $b, d$  such that  $a \equiv c \pmod{(b, d)}$  and  $x \equiv \chi_{b,d}(a, c) \pmod{[b, d]}$ . The objective of the following instructions is to show that  $x \equiv a \pmod{b}$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle b, d \rangle$  and let  $\langle e, f, g, h, i \rangle$  receive.
2. Verify that  $x \bmod [b, d] = \chi_{b,d}(a, c) \bmod [b, d]$ .
3. Therefore verify that  $x \bmod (bg) = \chi_{b,d}(a, c) \bmod (bg)$ .
4. Therefore verify that  $(x \bmod (bg)) \bmod b = (\chi_{b,d}(a, c) \bmod (bg)) \bmod b$ .

5. Therefore using **procedure 1.03**, verify that  $x \bmod b = \chi_{b,d}(a, c) \bmod b = (a + ((c - a) \operatorname{div}(b, d))hb) \bmod b = a \bmod b$ .

## Procedure 1.42

### Objective

Choose three integers  $x, a, c$  and two positive integers  $b, d$  such that  $a \equiv c \pmod{(b, d)}$  and  $x \equiv \chi_{b,d}(a, c) \pmod{[b, d]}$ . The objective of the following instructions is to show that  $x \equiv a \pmod{b}$  and  $x \equiv c \pmod{d}$ .

### Implementation

1. Execute **procedure 1.41** on  $\langle x, a, c, b, d \rangle$ .
2. Therefore verify that  $x \equiv a \pmod{b}$ .
3. Now verify that  $x \equiv \chi_{b,d}(a, c) \equiv \chi_{d,b}(c, a) \pmod{[d, b]}$
4. Execute **procedure 1.41** on  $\langle x, c, a, d, b \rangle$ .
5. Therefore verify that  $x \equiv c \pmod{d}$ .

## Procedure 1.43

### Objective

Choose two integers  $a, c$  and three positive integers  $b, d, e$  such that  $a \equiv c \pmod{(b, d)}$ . The objective of the following instructions is to show that  $\chi_{b,d}(ea, ec) = e\chi_{b,d}(a, c)$ .

### Implementation

1. Verify that  $\chi_{b,d}(a, c) \equiv a \pmod{b}$ .
2. Therefore using **procedure 1.05**, verify that  $e\chi_{b,d}(a, c) \equiv ea \pmod{b}$ .
3. Verify that  $\chi_{b,d}(a, c) \equiv c \pmod{d}$ .
4. Therefore using **procedure 1.05**, verify that  $e\chi_{b,d}(a, c) \equiv ec \pmod{d}$ .
5. Also using **procedure 1.02** and (O), verify that  $ea \equiv ec \pmod{(b, d)}$ .
6. Therefore using **procedure 1.39**, verify that  $e\chi_{b,d}(a, c) \equiv \chi_{b,d}(ea, ec) \pmod{[b, d]}$ .

7. **Therefore verify that**  $e\chi_{b,d}(a, c) = \chi_{b,d}(ea, ec)$ .

## Procedure 1.44

### Objective

Choose two integers  $a, c$  and three positive integers  $b, d, e$  such that  $a \equiv c \pmod{eb, ed}$ . The objective of the following instructions is to show that  $\chi_{eb,fd}(a, c) \pmod{[b, d]} = \chi_{b,d}(a, c)$ .

### Implementation

1. Verify that  $\chi_{eb,ed}(a, c) \equiv a \pmod{eb}$ .
2. Therefore using **procedure 1.03**, verify that  $\chi_{eb,ed}(a, c) \equiv a \pmod{b}$ .
3. Verify that  $\chi_{eb,ed}(a, c) \equiv c \pmod{ed}$ .
4. Therefore using **procedure 1.03**, verify that  $\chi_{eb,ed}(a, c) \equiv c \pmod{d}$ .
5. Now verify that  $a \equiv c \pmod{e(b, d)}$ .
6. Therefore using **procedure 1.03**, verify that  $a \equiv c \pmod{(b, d)}$ .
7. Therefore using **procedure 1.39**, verify that  $\chi_{eb,ed}(a, c) \equiv \chi_{b,d}(a, c) \pmod{[b, d]}$ .
8. **Therefore verify that**  $\chi_{eb,ed}(a, c) \pmod{[b, d]} = \chi_{b,d}(a, c)$ .

## Procedure 1.45

### Objective

Choose three integers  $a, c, e$  and three positive integers  $b, d, f$  such that  $a \equiv e \pmod{(b, f)}$ , and  $c \equiv e \pmod{(d, f)}$ . The objective of the following instructions is to show that  $\chi_{b,d}(a, c) \equiv e \pmod{[b, d, f]}$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle b, f \rangle$  and let  $\langle g_0, h_0, i_0, j_0, k_0 \rangle$  receive.
2. Execute **procedure 1.09** on  $\langle d, f \rangle$  and let  $\langle g_1, h_1, i_1, j_1, k_1 \rangle$  receive.
3. Verify that  $e \equiv a \pmod{(b, f)}$ .

4. Verify that  $e \equiv c \pmod{(d, f)}$ .

5. Therefore using **procedure 1.39** and **procedure 1.44**, verify that  $e$

$$\begin{aligned} (a) &\equiv \chi_{(b,f),(d,f)}(a, c) \\ (b) &\equiv \chi_{(b,f)h_1,(d,f)h_2}(a, c) \\ (c) &= \chi_{b,d}(a, c) \pmod{[(b, f), (d, f)]}. \end{aligned}$$

6. **Therefore using procedure 1.36**, verify that  $e \equiv \chi_{b,d}(a, c) \pmod{[b, d, f]}$ .

## Procedure 1.46

### Objective

Choose three integers  $a, c, e$  and three positive integers  $b, d, f$  such that  $a \equiv c \pmod{(b, d)}$ ,  $a \equiv e \pmod{(b, f)}$ , and  $c \equiv e \pmod{(d, f)}$ . Execute **procedure 1.45** on  $\langle a, c, e, b, d, f \rangle$ . Execute **procedure 1.45** on  $\langle c, e, a, d, f, b \rangle$ . The objective of the following instructions is to show that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) = \chi_{b,[d,f]}(a, \chi_{d,f}(c, e))$ .

### Implementation

1. Verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv e \pmod{f}$ .
2. Verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv \chi_{b,d}(a, c) \pmod{[b, d]} = gb = hd$ .
3. Therefore using **procedure 1.03**, verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv \chi_{b,d}(a, c) \equiv a \pmod{b}$ .
4. Also using **procedure 1.03**, verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv \chi_{b,d}(a, c) \equiv c \pmod{d}$ .
5. Therefore using (1), (4), and **procedure 1.39**, verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv \chi_{d,f}(c, e) \pmod{[d, f]}$ .
6. Therefore using (3), (5), and **procedure 1.39**, verify that  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) \equiv \chi_{b,[d,f]}(a, \chi_{d,f}(c, e)) \pmod{[b, [d, f]]} = [[b, d], f]$ .
7. **Therefore verify that**  $\chi_{[b,d],f}(\chi_{b,d}(a, c), e) = \chi_{b,[d,f]}(a, \chi_{d,f}(c, e))$ .

### Notation 1.13

Let us use the notation  $\chi_{b_1, b_2, \dots, b_n}(a_1, a_2, \dots, a_n)$  as a shorthand for "either  $\chi_{b_1, [b_2, b_3, \dots, b_n]}(a_1, \chi_{b_2, b_3, \dots, b_n}(a_2, a_3, \dots, a_n))$  or



$\chi_{[b_1, b_2], [b_3, b_4, \dots, b_n]}(\chi_{b_1, b_2}(a_1, a_2), \chi_{b_3, b_4, \dots, b_n}(a_3, a_4, \dots, a_n))$   
 or  $\dots$  or  $\chi_{[b_1, b_2, \dots, b_{n-1}], b_n}(\chi_{b_1, b_2, \dots, b_{n-1}}(a_1, a_2, \dots, a_{n-1}), a_n)$ ".

### Notation 1.14

Let us use the notation  $\phi(n)$  as a shorthand for "the list of distinct non-negative integers less than  $n$  where each integer  $x$  is such that  $(x, n) = 1$ ".

### Procedure 1.47

#### Objective

Choose an integer  $a$  and a positive integer  $b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each element of  $a\phi(b) \bmod b$  is in  $\phi(b)$ .

#### Implementation

1. Verify that  $(a, b) = 1$ .
2. For  $i = 1$  to  $i = \phi(b)$ , do the following:
  - (a) Using (O), verify that  $(\phi(b)_i, b) = 1$ .
  - (b) Execute **procedure 1.20** on  $\langle a, \phi(b)_i, b \rangle$ .
  - (c) Therefore verify that  $(a\phi(b)_i, b) = 1$ .
  - (d) Execute **procedure 1.17** on  $\langle a\phi(b)_i \bmod b, a\phi(b)_i, b \rangle$ .
  - (e) Therefore verify that  $(a\phi(b)_i \bmod b, b) = (a\phi(b)_i, b) = 1$ .
  - (f) Also verify that  $0 \leq a\phi(b)_i \bmod b < b$ .
  - (g) Therefore verify that  $a\phi(b)_i \bmod b$  is contained in the list  $\phi(b)$ .
3. **Therefore verify that each element of  $a\phi(b) \bmod b$  is in  $\phi(b)$ .**

### Procedure 1.48

#### Objective

Choose an integer  $a$  and a positive integer  $b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each element of  $a\phi(b) \bmod b$  is distinct.

#### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle r, t, u, v, w \rangle$  receive.
2. Verify that  $va + wb = r = (a, b) = 1$ .
3. Therefore verify that  $va \equiv 1 \pmod{b}$ .
4. Now for  $i = 1$  to  $i = |\phi(b)|$ , do the following:
  - (a) For  $j = i + 1$  to  $j = |\phi(b)|$ , do the following:
    - i. If  $a\phi(b)_i \equiv a\phi(b)_j \pmod{b}$ , then do the following:
      - A. Verify that  $\phi(b)_i \equiv va\phi(b)_i \equiv va\phi(b)_j \equiv \phi(b)_j \pmod{b}$ .
      - B. Therefore verify that  $\phi(b)_i = \phi(b)_j$ .
      - C. Also verify that  $i \neq j$ .
      - D. Therefore using (O), verify that  $\phi(b)_i \neq \phi(b)_j$ .
      - E. **Therefore using (B) and (D), verify that  $\phi(b)_i \neq \phi(b)_j$ .**
      - F. **Abort procedure.**
    - ii. Otherwise, do the following:
      - A. Verify that  $a\phi(b)_i \not\equiv a\phi(b)_j \pmod{b}$ .
5. **Therefore verify that  $a\phi(b) \bmod b$  is composed of distinct elements.**

### Procedure 1.49

#### Objective

Choose an integer  $a$  and a positive integer  $b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that  $a\phi(b) \bmod b$  is a rearrangement of  $\phi(b)$ .

#### Implementation

1. Execute **procedure 1.47** on  $\langle a, b \rangle$ .
2. Therefore verify that each element of  $a\phi(b) \bmod b$  is in  $\phi(b)$ .
3. Verify that  $|a\phi(b) \bmod b| = |\phi(b)|$ .
4. Execute **procedure 1.48** on  $\langle a, b \rangle$ .

5. Therefore verify that  $a\phi(b) \bmod b$  is composed of distinct elements.
6. **Therefore verify that  $a\phi(b) \bmod b$  is a rearrangement of  $\phi(b)$ .**

## Procedure 1.50

### Objective

Choose an integer  $a$  and a positive integer  $b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that  $a^{|\phi(b)|} \equiv 1 \pmod{b}$ .

### Implementation

1. For  $i = 1$  to  $i = |\phi(b)|$ , do the following:
  - (a) Execute **procedure 1.09** on  $\langle \phi(b)_i, b \rangle$  and let  $\langle r_i, t_i, u_i, v_i, w_i \rangle$ .
  - (b) Using (O), verify that  $v_i \phi(b)_i + w_i b = r_i = \phi(b)_i, b = 1$ .
  - (c) Therefore verify that  $v_i \phi(b)_i \equiv 1 \pmod{b}$ .
2. Therefore using **procedure 1.49**, verify that  $\prod_{i=1}^{|\phi(b)|} \phi(b)_i \equiv \prod_{i=1}^{|\phi(b)|} a \phi(b)_i \equiv a^{|\phi(b)|} \prod_{i=1}^{|\phi(b)|} \phi(b)_i \pmod{b}$ .
3. **Therefore verify that**  $1 \equiv \prod_{i=1}^{|\phi(b)|} (v_i \phi(b)_i) = \prod_{i=1}^{|\phi(b)|} v_i \prod_{i=1}^{|\phi(b)|} \phi(b)_i \equiv a^{|\phi(b)|} \prod_{i=1}^{|\phi(b)|} \phi(b)_i \prod_{i=1}^{|\phi(b)|} v_i \equiv a^{|\phi(b)|} \pmod{b}$ .

### Notation 1.15

Let us use the notation  $a \times b$  as a shorthand for "the  $|a| \times |b|$  matrix such that for  $1 \leq i \leq |a|$ , for  $1 \leq j \leq |b|$ ,  $(a \times b)_{i,j} = \langle a_i, b_j \rangle$ ".

## Procedure 1.52

### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each entry of  $\chi_{a,b}([0 : a] \times [0 : b])$  is in  $[0 : ab]$ .

### Implementation

1. Execute **procedure 1.09** on  $\langle a, b \rangle$  and let  $\langle c, d, e, f, g \rangle$  receive.
2. Let  $h = \chi_{a,b}([0 : a] \times [0 : b])$ .
3. Verify that  $c = (a, b) = 1$ .
4. Therefore verify that  $b = ce = (1)e = e$ .
5. Therefore verify that  $0 \leq h_{i,j} < [a, b] = ae = ab$  for  $1 \leq i \leq a$ , for  $1 \leq j \leq b$ .
6. **Therefore verify that each entry of  $h$  is in  $[0 : ab]$ .**

## Procedure 1.53

### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each entry of  $\chi_{a,b}([0 : a] \times [0 : b])$  is distinct.

### Implementation

1. Let  $h = \chi_{a,b}([0 : a] \times [0 : b])$ .
2. For each distinct unordered pair of index pairs  $\langle i, j \rangle$  and  $\langle k, l \rangle$  of  $h$ , do the following:
  - (a) If  $h_{i,j} = h_{k,l}$ , then do the following:
    - i. Verify that  $\chi_{a,b}([0 : a]_i, [0 : b]_j) = h_{i,j} = h_{k,l} = \chi_{a,b}([0 : a]_k, [0 : b]_l)$ .
    - ii. Verify that  $\chi_{a,b}(i-1, j-1) = \chi_{a,b}(k-1, l-1)$ .
    - iii. Therefore verify that  $i-1 \equiv \chi_{a,b}(i-1, j-1) = \chi_{a,b}(k-1, l-1) \equiv k-1 \pmod{a}$ .
    - iv. Therefore verify that  $i = k$ .
    - v. Also verify that  $j-1 \equiv \chi_{a,b}(i-1, j-1) = \chi_{a,b}(k-1, l-1) \equiv l-1 \pmod{b}$ .
    - vi. Therefore verify that  $j = l$ .
    - vii. Therefore verify that  $\langle i, j \rangle = \langle k, l \rangle$ .
    - viii. Using (2), verify that  $\langle i, j \rangle \neq \langle k, l \rangle$ .
    - ix. **Therefore verify that  $\langle i, j \rangle \neq \langle i, j \rangle$ .**
    - x. **Abort procedure.**
  - (b) Otherwise do the following:

- i. Verify that  $h_{i,j} \neq h_{k,l}$ .
- 3. **Therefore verify that each entry of  $h$  is distinct.**

### Procedure 1.54

#### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that  $\chi_{a,b}([0 : a] \times [0 : b])$  is a rearrangement  $[0 : ab]$ .

#### Implementation

1. Let  $h = \chi_{a,b}([0 : a] \times [0 : b])$ .
2. Execute **procedure 1.52** on  $\langle a, b \rangle$ .
3. Therefore verify that each element of  $h$  is in  $[0 : ab]$ .
4. Also verify that  $h$  has the same number of entries as  $[0 : ab]$ .
5. Execute **procedure 1.53** on  $\langle a, b \rangle$ .
6. Therefore verify that  $h$  is composed of distinct elements.
7. **Therefore verify that  $h$  is a rearrangement of  $[0 : ab]$ .**

### Procedure 1.55

#### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each entry of  $\chi_{a,b}(\phi(a) \times \phi(b))$  is in  $\phi(ab)$ .

#### Implementation

1. Let  $h = \chi_{a,b}(\phi(a) \times \phi(b))$ .
2. Now, for each index pair  $\langle i, j \rangle$  of  $h$ , do the following:
  - (a) Verify that  $0 \leq h_{i,j} < [a, b] = [a, b](a, b) = ab$ .
  - (b) Verify that  $h_{i,j} = \chi_{a,b}(\phi(a)_i, \phi(b)_j) \equiv \phi(a)_i \pmod{a}$ .
  - (c) Execute **procedure 1.17** on  $\langle h_{i,j}, \phi(a)_i, a \rangle$ .

- (d) Therefore verify that  $(a, h_{i,j}) = (h_{i,j}, a) = (\phi(a)_i, a) = 1$ .
- (e) Verify that  $h_{i,j} = \chi_{a,b}(\phi(a)_i, \phi(b)_j) \equiv \phi(b)_j \pmod{b}$ .
- (f) Execute **procedure 1.17** on  $\langle h_{i,j}, \phi(b)_j, b \rangle$ .
- (g) Therefore verify that  $(b, h_{i,j}) = (h_{i,j}, b) = (\phi(b)_j, b) = 1$ .
- (h) Therefore verify that  $(h_{i,j}, ab) = (ab, h_{i,j}) = 1$ .
- (i) Therefore verify that  $h_{i,j}$  is in  $\phi(ab)$ .

3. **Therefore verify that each entry of  $\chi_{a,b}(\phi(a) \times \phi(b))$  is in  $\phi(ab)$ .**

### Procedure 1.56

#### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that each entry of  $\phi(ab)$  is in  $\chi_{a,b}(\phi(a) \times \phi(b))$ .

#### Implementation

1. For  $i = 1$  to  $i = |\phi(ab)|$ , do the following:
  - (a) Verify that  $(\phi(ab)_i, ab) = 1$ .
  - (b) Verify that  $\phi(ab)_i \equiv \phi(ab)_i \pmod{a}$ .
  - (c) Therefore verify that  $(\phi(ab)_i \pmod{a}, a) = (\phi(ab)_i, a) = 1$ .
  - (d) Also verify that  $0 \leq \phi(ab)_i \pmod{a} < a$ .
  - (e) Therefore verify that  $\phi(ab)_i \pmod{a}$  is amongst  $\phi(a)$ .
  - (f) Verify that  $\phi(ab)_i \equiv \phi(ab)_i \pmod{b}$ .
  - (g) Also verify that  $(\phi(ab)_i \pmod{b}, b) = (\phi(ab)_i, b) = 1$ .
  - (h) Also verify that  $0 \leq \phi(ab)_i \pmod{b} < b$ .
  - (i) Therefore verify that  $\phi(ab)_i \pmod{b}$  is amongst  $\phi(b)$ .
  - (j) Therefore verify that  $\langle \phi(ab)_i \pmod{a}, \phi(ab)_i \pmod{b} \rangle$  is amongst  $\phi(a) \times \phi(b)$ .

- (k) Also using (b) and (f), verify that  $\phi(ab)_i \equiv \chi_{a,b}(\phi(ab)_i \bmod a, \phi(ab)_i \bmod b) \pmod{[a, b]} = [a, b](a, b) = ab$ .
  - (l) Therefore verify that  $\phi(ab)_i = \chi_{a,b}(\phi(ab)_i \bmod a, \phi(ab)_i \bmod b)$ .
  - (m) Therefore using (j) and (l), verify that  $\phi(ab)_i$  is amongst  $\chi_{a,b}(\phi(a) \times \phi(b))$ .
2. **Therefore verify that each entry of  $\phi(ab)$  is in  $\chi_{a,b}(\phi(a) \times \phi(b))$ .**

## Procedure 1.57

### Objective

Choose two positive integers  $a, b$  such that  $(a, b) = 1$ . The objective of the following instructions is to show that  $\phi(ab)$  is a rearrangement of  $\chi_{a,b}(\phi(a) \times \phi(b))$  and that  $|\phi(ab)| = |\phi(a)||\phi(b)|$ .

### Implementation

1. Execute **procedure 1.54** on  $\langle a, b \rangle$ .
2. Therefore verify that  $\chi_{a,b}([0 : a] \times [0 : b])$  are a rearrangement of  $[0 : ab]$ .
3. Verify that  $\chi_{a,b}(\phi(a) \times \phi(b))$  is a submatrix of  $\chi_{a,b}([0 : a] \times [0 : b])$ .
4. Therefore verify that the entries of  $\chi_{a,b}(\phi(a) \times \phi(b))$  are distinct.
5. Execute **procedure 1.55** on  $\langle a, b \rangle$ .
6. Therefore verify that the entries of  $\chi_{a,b}(\phi(a) \times \phi(b))$  are in  $\phi(ab)$ .
7. Verify that the entries of  $\phi(ab)$  are distinct.
8. Execute **procedure 1.56** on  $\langle a, b \rangle$ .
9. Therefore verify that the entries of  $\phi(ab)$  are in  $\chi_{a,b}(\phi(a) \times \phi(b))$ .
10. **Therefore verify that  $\phi(ab)$  is a rearrangement of  $\chi_{a,b}(\phi(a) \times \phi(b))$ .**
11. **Therefore verify that  $|\phi(ab)| = |\chi_{a,b}(\phi(a) \times \phi(b))| = |\phi(a) \times \phi(b)| = |\phi(a)||\phi(b)|$ .**

## Notation 1.16

Let us use the notation  $[P]$  as a shorthand for "1 if  $P$ , 0 if otherwise".

## Procedure 1.58

### Objective

Choose a positive integer  $a$  and a prime  $b$ . The objective of the following instructions is to show that  $|\phi(b^a)| = b^a - b^{a-1}$ .

### Implementation

1. Using **procedure 1.21**, verify that  $\sum_{r=0}^{b^a-1} [(r, b^a) = 1] \leq \sum_{r=0}^{b^a-1} [(r, b) = 1]$ .
2. Using **procedure 1.20**, verify that  $\sum_{r=0}^{b^a-1} [(r, b) = 1] \leq \sum_{r=0}^{b^a-1} [(r, b^a) = 1]$ .
3. Therefore verify that  $\sum_{r=0}^{b^a-1} [(r, b^a) = 1] = \sum_{r=0}^{b^a-1} [(r, b) = 1]$ .
4. Using **procedure 1.13**, verify that  $\sum_{r=0}^{b^a-1} [(r, b) = 1] \leq \sum_{r=0}^{b^a-1} [r \bmod b \neq 0]$ .
5. Using **procedure 1.22**, verify that  $\sum_{r=0}^{b^a-1} [r \bmod b \neq 0] \leq \sum_{r=0}^{b^a-1} [(r, b) = 1]$ .
6. Therefore verify that  $\sum_{r=0}^{b^a-1} [(r, b) = 1] = \sum_{r=0}^{b^a-1} [r \bmod b \neq 0]$ .
7. **Therefore using (3) and (6), verify that  $|\phi(b^a)| = \sum_{r=0}^{b^a-1} [(r, b^a) = 1] = \sum_{r=0}^{b^a-1} [(r, b) = 1] = \sum_{r=0}^{b^a-1} [r \bmod b \neq 0] = \sum_{r=0}^{b^a-1} (1 - [r \bmod b = 0]) = b^a - b^{a-1}$ .**

## Procedure 1.59

### Objective

Choose a list of primes  $a$ . Let  $b$  be the list of distinct primes in  $a$ . Let  $c$  be a list such that  $c_i$  is the multiplicity of  $b_i$  in  $a$  for  $i = 1$  to  $i = |b|$ . The objective of the following instructions is to show that  $|\phi(a_*)| = \prod_{i=1}^{|b|} (b_i^{c_i} - b_i^{c_i-1})$ .

## Implementation

1. If  $a = \langle \rangle$ , then do the following:
  - (a) Verify that  $|b| = |a| = 0$ .
  - (b) **Therefore verify that**  $\phi(a_*) = \phi(1) = 1 = \prod_{i=1}^{|b|} (b_i^{c_i} - b_i^{c_i-1})$ .
2. Otherwise, do the following:
  - (a) Verify that  $a_* = \prod_{i=1}^{|b|} b_i^{c_i}$ .
  - (b) Verify that  $|a| > 0$ .
  - (c) Therefore verify that  $|c| = |b| > 0$ .
  - (d) Therefore using **procedure 1.30**, verify that  $(b_1^{c_1}, \prod_{i=2}^{|b|} b_i^{c_i}) = 1$ .
  - (e) Let  $d$  be the list  $a$  with all instances of  $a_1$  removed.
  - (f) Verify that  $|d| < |a|$ .
  - (g) Now execute **procedure 1.59** on  $\langle d \rangle$ .
  - (h) Hence verify that  $\phi(d_*) = \phi(\prod_{i=2}^{|b|} b_i^{c_i}) = \prod_{i=2}^{|b|} (b_i^{c_i} - b_i^{c_i-1})$ .
  - (i) **Therefore using (d), (h), procedure 1.57 and procedure 1.58, verify that**  $|\phi(a_*)| = |\phi(\prod_{i=1}^{|b|} b_i^{c_i})| = |\phi(b_1^{c_1} \prod_{i=2}^{|b|} b_i^{c_i})| = |\phi(b_1^{c_1})| |\phi(\prod_{i=2}^{|b|} b_i^{c_i})| = (b_1^{c_1} - b_1^{c_1-1}) |\phi(\prod_{i=2}^{|b|} b_i^{c_i})| = (b_1^{c_1} - b_1^{c_1-1}) \prod_{i=2}^{|b|} (b_i^{c_i} - b_i^{c_i-1}) = \prod_{i=1}^{|b|} (b_i^{c_i} - b_i^{c_i-1})$ .

## Notation 1.17

Let us use the notation  $a^b$  as a shorthand for " $a(a-1) \cdots (a-(b-1))$ ".

## Procedure 1.60

### Objective

Choose a list of distinct elements  $a$  and a non-negative integer  $b$  such that  $b \leq |a|$ . Let  $c$  be a list of length- $b$  permutations of  $a$ . The objective of the following instructions is to show that  $|c| = |a|^b$ .

## Implementation

1. If  $|b| > 0$ , then do the following:
  - (a) For each entry  $d$  in  $a$ , do the following:
    - i. Let  $e$  be the list formed by removing  $d$  from  $a$ .
    - ii. Verify that the entries of  $e$  are distinct.
    - iii. Verify that  $|e| = |a| - 1$ .
    - iv. Now execute **procedure 1.60** on  $\langle e, b-1 \rangle$ .
    - v. Therefore verify that the number of length- $b-1$  permutations of  $e$  is  $|e|^{b-1}$ .
    - vi. Therefore verify that the number of length- $b$  permutations of  $a$  beginning with  $d$  is  $|e|^{b-1} = (|a| - 1)^{b-1}$ .
  - (b) Therefore verify that the number of length- $b$  permutations of  $a$  beginning with any entry of  $a$  is  $|a|(|a| - 1)^{b-1} = |a|^b$ .
  - (c) Therefore verify that the number of length- $b$  permutations of  $a$  are  $|a|^b$ .
  - (d) **Therefore verify that**  $|c| = |a|^b$ .
2. Otherwise do the following:
  - (a) Verify that  $b = 0$ .
  - (b) Verify that the number of length-0 permutations of  $a$  is 1.
  - (c) **Therefore verify that**  $|c| = 1 = |a|^0 = |a|^b$ .

## Notation 1.18

Let us use the notation  $\binom{n}{r}$  as a shorthand for " $n^{\text{div}(r!)} \text{ mod } r!$ ".

## Procedure 1.61

### Objective

Choose a list of distinct elements  $n$  and a non-negative integer  $r$  such that  $r \leq |n|$ . Let  $b$  be the largest list of length- $r$  sublists of  $n$  such that no two of them are permutations of each other. The objective of the following instructions is to either show that  $1 > 2$  or to show that  $|b| = \binom{|n|}{r}$  and that  $n^{\text{div}(r!)} \text{ mod } r! = 0$ .

## Implementation

1. Let  $a$  and  $f$  be a list of all the permutations of  $n$ .
2. Using **procedure 1.60**, verify that  $|a| = |n|^{|n|}$ .
3. For each list  $c$  in  $b$ , do the following:
  - (a) Using **procedure 1.60**, verify that the number of permutations of  $c$  is  $r!$ .
  - (b) Let  $d$  be the list obtained by removing the elements of  $c$  from  $n$ .
  - (c) Using **procedure 1.60**, verify that the number of permutations of  $d$  is  $(n - r)!$ .
  - (d) Let  $e$  be the list of permutations of  $n$  beginning with a permutations of  $c$ .
  - (e) Verify that  $|e| = |c||d| = r!(n - r)!$ .
  - (f) If  $e$  is not a sublist of  $a$ , then do the following:
    - i. Let  $g$  be a list in  $e$  that is not in  $a$ .
    - ii. Verify that  $e$  is a sublist of  $f$ .
    - iii. Therefore verify that  $g$  was in  $a$  but then was removed.
    - iv. Therefore verify that the variable  $c$  was formerly equal to a permutation of the current  $c$ .
    - v. **Therefore verify that  $b$  contains at least two permutations of  $c$ .**
    - vi. **Abort procedure.**
  - (g) Otherwise, do the following:
    - i. Verify that  $e$  is a sublist of  $a$ .
    - ii. Remove the lists in  $e$  from  $a$ .
4. If  $a \neq \langle \rangle$ , then do the following:
  - (a) Let  $g$  be a list in  $a$ .
  - (b) Let  $h$  be the sublist of  $g$  corresponding to its first  $r$  elements.
  - (c) Therefore verify that the permutations of  $n$  beginning with a permutation of  $h$  were never removed from  $a$ .
  - (d) Therefore verify that the variable  $c$  was never equal to a permutation of  $h$ .
  - (e) Therefore verify that no permutation of  $h$  is in  $b$ .

- (f) **Therefore verify that  $b \cap \langle h \rangle$  is larger than  $b$  and is also a list of length- $r$  sublists of  $n$  such that no two of them are permutations of each other.**
  - (g) **Abort procedure.**
5. Otherwise do the following:
- (a) Verify that  $|n|! \bmod (r!(|n| - r)!) = 0$ .
  - (b) Therefore verify that  $|n|! = (|n|! \div (r!(|n| - r)!))r!(|n| - r)!$ .
  - (c) Therefore verify that  $|n|! \div (|n| - r)! = (|n|! \div (r!(|n| - r)!))r!$ .
  - (d) **Therefore verify that  $n^x \bmod r! = (|n|! \div (|n| - r)!) \bmod r! = 0$ .**
  - (e) Also verify that (3) iterated  $|n|! \div (r!(|n| - r)!)$  times.
  - (f) **Therefore using **procedure 1.08**, verify that  $|b| = |n|! \div (r!(|n| - r)!) = (|n|! \div (|n| - r)!) \div (r!) = n^x \div (r!) = \binom{n}{r}$ .**

## Procedure 1.62

### Objective

Choose two positive integers  $a, b$ . The objective of the following instructions is to show that  $\binom{a}{b} = \binom{a-1}{b-1} + \binom{a-1}{b}$ .

### Implementation

1. Using **procedure 1.05** and **procedure 1.06**, verify that  $\binom{a-1}{b-1} + \binom{a-1}{b}$ 
  - (a)  $= (a - 1)^{\underline{b-1}} \div (b - 1)! + (a - 1)^{\underline{b}} \div b!$
  - (b)  $= ((a - 1)^{\underline{b-1}} b) \div b! + (a - 1)^{\underline{b}} \div b!$
  - (c)  $= ((a - 1)^{\underline{b-1}} b + (a - 1)^{\underline{b}}) \div b!$
  - (d)  $= ((a - 1)^{\underline{b-1}} b + (a - 1)^{\underline{b-1}} (a - b)) \div b!$
  - (e)  $= ((a - 1)^{\underline{b-1}} a) \div b!$
  - (f)  $= a^{\underline{b}} \div b!$
  - (g)  $= \binom{a}{b}$ .

### Notation 1.19

Let us use the notation  $\mathbb{Z}$  as a shorthand for "integer".

### Notation 1.20

Let us use the notation  $A[x_1, x_2, \dots, x_n]$  as a shorthand for "formal polynomial with  $A$  coefficients in the indeterminates  $x_1, x_2, \dots, x_n$ ".

## Procedure 1.63

### Objective

Choose a non-negative integer  $a$ . The objective of the following instructions is to show that the  $\mathbb{Z}[x]$   $(1+x)^a = \sum_{r=0}^a \binom{a}{r} x^r$ .

### Implementation

1. If  $a = 0$ , then do the following:
  - (a) **Verify that**  $(1+x)^a = (1+x)^0 = 1 = \sum_{r=0}^0 \binom{0}{r} x^r = \sum_{r=0}^a \binom{a}{r} x^r$ .
2. Otherwise, do the following:
  - (a) Verify that  $a > 0$ .
  - (b) Therefore verify that  $a - 1 \geq 0$ .
  - (c) Execute **procedure 1.63** on  $\langle a - 1 \rangle$ .
  - (d) Therefore verify that  $(1+x)^{a-1} = \sum_{r=0}^{a-1} \binom{a-1}{r} x^r$ .
  - (e) Therefore using **procedure 1.62**, verify that  $(1+x)^a$ 
    - i.  $= (1+x)(1+x)^{a-1}$
    - ii.  $= (1+x) \sum_{r=0}^{a-1} \binom{a-1}{r} x^r$
    - iii.  $= \sum_{r=0}^{a-1} \binom{a-1}{r} x^r + \sum_{r=0}^{a-1} \binom{a-1}{r} x^{r+1}$
    - iv.  $= \sum_{r=0}^a \binom{a-1}{r} x^r + \sum_{r=1}^a \binom{a-1}{r-1} x^r$
    - v.  $= 1 + \sum_{r=1}^a \left( \binom{a-1}{r} + \binom{a-1}{r-1} \right) x^r$
    - vi.  $= 1 + \sum_{r=1}^a \binom{a}{r} x^r$
    - vii.  $= \sum_{r=0}^a \binom{a}{r} x^r$ .

## Procedure 1.64

### Objective

Choose an integer  $r$  and a prime  $n$  such that  $0 < r < n$ . The objective of the following instructions is to show that either  $0 \neq 0$  or  $\binom{n}{r} \bmod n = 0$ .

### Implementation

1. Using **procedure 1.61**, verify that  $\binom{n}{r} r! = n^r \equiv 0 \pmod{n}$ .
2. If  $\binom{n}{r} \bmod n \neq 0$ , then do the following:
  - (a) Verify that  $i \bmod n \neq 0$  for  $i = 1$  to  $i = r$ .
  - (b) Therefore using **procedure 1.23**, verify that  $r! \bmod n \neq 0$ .
  - (c) Therefore using (2) and (b), verify that  $\binom{n}{r} r! \bmod n \neq 0$ .
  - (d) **Therefore using (1) and (c), verify that**  $0 \neq 0$ .
  - (e) **Abort procedure.**
3. Otherwise, do the following:
  - (a) **Verify that**  $\binom{n}{r} \bmod n = 0$ .

## Procedure 1.65

### Objective

Choose a non-negative integer  $a$  and a prime  $b$ . The objective of the following instructions is to show that the  $\mathbb{Z}[x]$   $\sum_{r=0}^a \binom{a}{r} x^r \equiv \sum_{r=0}^a \binom{a \bmod b}{r \bmod b} \binom{a \bmod b}{r \bmod b} x^r \pmod{b}$ .

### Implementation

1. Using **procedure 1.02**, **procedure 1.63**, and **procedure 1.64**, verify that  $\sum_{r=0}^a \binom{a}{r} x^r$ 
  - (a)  $= (1+x)^a$
  - (b)  $= (1+x)^{(a \bmod b)b + a \bmod b}$
  - (c)  $= (1+x)^{(a \bmod b)b} (1+x)^{a \bmod b}$
  - (d)  $= ((1+x)^b)^{a \bmod b} (1+x)^{a \bmod b}$
  - (e)  $= \left( \sum_{u=0}^b \binom{b}{u} x^u \right)^{a \bmod b} \left( \sum_{t=0}^{a \bmod b} \binom{a \bmod b}{t} x^t \right)$

$$\begin{aligned}
(f) &\equiv (1 + x^b)^{a \operatorname{div} b} \left( \sum_{t=0}^{a \bmod b} \binom{a \bmod b}{t} x^t \right) \\
(g) &= \left( \sum_{u=0}^{a \operatorname{div} b} (x^b)^u \binom{a \operatorname{div} b}{u} \right) \left( \sum_{t=0}^{a \bmod b} \binom{a \bmod b}{t} x^t \right) \\
(h) &= \sum_{u=0}^{a \operatorname{div} b} \sum_{t=0}^{a \bmod b} \binom{a \operatorname{div} b}{u} \binom{a \bmod b}{t} x^{ub+t} \\
(i) &= \sum_{u=0}^{a \operatorname{div} b} \sum_{t=0}^{a \bmod b} \binom{a \operatorname{div} b}{(ub+t) \operatorname{div} b} \binom{a \bmod b}{(ub+t) \bmod b} x^{ub+t} \\
(j) &= \sum_{r=0}^a \binom{a \operatorname{div} b}{r \operatorname{div} b} \binom{a \bmod b}{r \bmod b} x^r \pmod{b}.
\end{aligned}$$

## 2 Rational Arithmetic

### Notation 2.00

Let us use the notation  $\mathbb{Q}$  as a shorthand for "rational".

### Procedure 2.0

#### Objective

Choose an integer  $n \geq 0$  and a  $\mathbb{Q}[x]$   $p = p_0x^n + p_1x^{n-1} + \dots + p_n$ . Let  $y, z$  be indeterminates. The objective of the following instructions is to construct a  $\mathbb{Q}[y, z]$   $G$  such that  $p(z) - p(y) = (z - y)G(y, z)$ .

#### Implementation

1. Let the  $\mathbb{Q}[y, z]$   $G = \sum_{r=1}^n p_{n-r}(z^{r-1} + z^{r-2}y + \dots + zy^{r-2} + y^{r-1})$ .
2. Verify that  $p(z) - p(y)$ 
  - (a)  $= (p_0z^n + p_1z^{n-1} + \dots + p_n) - (p_0y^n + p_1y^{n-1} + \dots + p_n)$
  - (b)  $= (\sum_{r=0}^n p_{n-r}z^r) - (\sum_{r=0}^n p_{n-r}y^r)$
  - (c)  $= \sum_{r=1}^n p_{n-r}(z^r - y^r)$
  - (d)  $= \sum_{r=1}^n p_{n-r}(z - y)(z^{r-1} + z^{r-2}y + \dots + zy^{r-2} + y^{r-1})$
  - (e)  $= (z - y) \sum_{r=1}^n p_{n-r}(z^{r-1} + z^{r-2}y + \dots + zy^{r-2} + y^{r-1})$
  - (f)  $= (z - y)G(y, z)$ .
3. Yield the tuple  $\langle G \rangle$ .

### Procedure 2.01

#### Objective

Choose a  $\mathbb{Q}[x]$   $p = x^n + p_1x^{n-1} + \dots + p_n$  and  $\mathbb{Q}$ s  $a_1 < a_2 < \dots < a_n < a_{n+1}$  in such a way that for  $i = 1$  to  $i = n + 1$ ,  $p(a_i) = 0$ . The objective of the following instructions is to show that  $0 \neq 0$ .

#### Implementation

1. Write  $p$  as  $1 * p$ , so that it has two factors.
2. For  $i = 1$  up to  $i = n$ , do the following:
  - (a) Let  $g$  be the rightmost factor of  $p$ .
  - (b) If  $g(a_i) \neq 0$ , do the following:
    - i. For  $k = 1$  to  $k = i - 1$ , verify that  $(a_i - a_k) \neq 0$ .
    - ii. Verify that  $p(a_i) \neq 0$ .
    - iii. Therefore using (O) and (ii), verify that  $0 \neq 0$ .
    - iv. **Abort procedure.**
  - (c) Otherwise  $g(a_i) = 0$ . Now do the following:
    - i. Execute **procedure 2.0** on  $g$  and let the tuple  $\langle G \rangle$  receive the result.
    - ii. Let  $x$  be an indeterminate.
    - iii. Let the  $\mathbb{Q}[x]$   $q = q(x) = G(a_i, x)$ .
    - iv. Verify that the  $\mathbb{Q}[x]$   $g = g(x) = g(x) - g(a_i) = (x - a_i)G(a_i, x) = (x - a_i)q(x) = (x - a_i)q$ .
    - v. Verify that  $p = (x - a_1)(x - a_2) \dots (x - a_i)q$ .
3. Now verify that  $p = (x - a_1)(x - a_2) \dots (x - a_n)1$ .
4. Using (3), verify that  $p(a_{n+1}) \neq 0$ .
5. Therefore using (O) and (4), verify that  $0 \neq 0$ .
6. **Abort procedure.**

### Procedure 2.02

#### Objective

Choose a  $\mathbb{Q}[x]$   $f$ . Choose  $\mathbb{Q}$ s  $a < b$  such that  $\operatorname{sgn}(f(a)) = -\operatorname{sgn}(f(b))$ . Choose a rational number



target  $B > 0$ . The objective of the following instructions is to construct a  $\mathbb{Q}$   $d$  such that  $a \leq d \leq b$  and  $|f(d)| < B$ .

### Implementation

1. Execute **procedure 2.0** on  $f$  and let the tuple  $\langle G \rangle$  receive the result.
2. Let  $x, y$  be indeterminates.
3. Verify that the  $\mathbb{Q}[x, y]$   $f(y) - f(x) = (y - x)G(x, y)$ .
4. Let  $c = a$  and  $d = b$ .
5. Until  $|d - c||G|(|a|, |b|) < B$ 
  - (a) Let  $e = \frac{c+d}{2}$ .
  - (b) If  $\text{sgn}(f(c)) = -\text{sgn}(f(e))$ , then:
    - i. Let  $d = e$ .
  - (c) Otherwise if  $\text{sgn}(f(e)) = -\text{sgn}(f(d))$ , then:
    - i. Let  $c = e$ .
  - (d) Otherwise if  $f(e) = 0$ , then do the following:
    - i. **Verify that**  $|f(e)| = 0 < B$ .
    - ii. Yield the tuple  $\langle e \rangle$ .
6. **Verify that**  $|f(c)|, |f(d)| < |f(d) - f(c)| = |(d - c)G(c, d)| = |d - c||G(c, d)| \leq |d - c||G|(|c|, |d|) \leq |d - c||G|(|a|, |b|) < B$ .
7. **Yield the tuple**  $\langle c \rangle$ .

### Procedure 2.03

#### Objective

Choose a  $\mathbb{Q}[x]$   $f = x^n + p_1x^{n-1} + \dots + p_n$  and pairs of  $\mathbb{Q}$ s  $(a_n, b_n), (a_{n-1}, b_{n-1}), \dots, (a_0, b_0)$  in such a way that:

1.  $a_n < b_n \leq a_{n-1} < b_{n-1} \leq \dots \leq a_1 < b_1 \leq a_0 < b_0$ .
2.  $\text{sgn}(f(a_i)) = -\text{sgn}(f(b_i))$  for  $i = 0$  to  $i = n$ .

The objective of the following instructions is to show that  $1 = -1$ .

### Implementation

1. If  $n > 0$ :
  - (a) Let  $B = \min_{k=0}^{n-1} \min(|f(a_k)|, |f(b_k)|)$ .
  - (b) For  $k = 0$  to  $k = n - 1$ , verify that  $|f(a_k)| \geq B$ .
  - (c) Execute **procedure 2.02** on the formal polynomial  $f$ , interval  $(a_n, b_n)$ , and target of  $B$ . Let the tuple  $\langle d \rangle$  receive the result.
  - (d) Verify that  $|f(d)| < B$ .
  - (e) Execute **procedure 2.0** on the formal polynomial  $f$  and let the tuple  $\langle G \rangle$  receive the result.
  - (f) Let  $x$  be an indeterminate.
  - (g) Let the formal polynomial  $h = G(d, x)$ .
  - (h) Verify that  $h$  is a monic  $(n - 1)^{th}$  degree formal polynomial.
  - (i) Verify that the formal polynomial  $f = f(x) = f(x) - f(d) + f(d) = (x - d)G(d, x) + f(d) = (x - d)h(x) + f(d) = (x - d)h + f(d)$ .
  - (j) For  $k = 0$  to  $k = n - 1$ , do the following:
    - i. If  $f(a_k) \geq B$ , in-order verify that:
      - A.  $f(a_k) \geq B > |f(d)| \geq f(d)$ .
      - B.  $f(a_k) - f(d) > 0$ .
      - C.  $(a_k - d)h(a_k) > 0$ .
      - D.  $h(a_k) > 0$ .
      - E.  $f(b_k) \leq -B < -|f(d)| \leq f(d)$ .
      - F.  $f(b_k) - f(d) < 0$ .
      - G.  $(b_k - d)h(b_k) < 0$ .
      - H.  $h(b_k) < 0$ .
    - ii. Otherwise, if  $f(a_k) \leq -B$ , do the following:
      - A. **Using steps analogous to (ji), verify that**  $h(a_k) < 0$ .
      - B. **Using steps analogous to (ji), verify that**  $h(b_k) > 0$ .
  - (k) Execute **procedure 2.03** on  $h$  and  $a_{n-1} < b_{n-1} \leq a_{n-2} < b_{n-2} \leq \dots \leq a_1 < b_1 \leq a_0 < b_0$ .
2. Otherwise, do the following:
  - (a) Verify that  $n = 0$ .

- (b) Therefore verify that  $h = 1$ .
- (c) **Therefore verify that**  $1 = \text{sgn}(1) = \text{sgn}(f_0(a_0)) = -\text{sgn}(f_0(b_0)) = -\text{sgn}(1) = -1$ .
- (d) **Abort procedure.**

#### Notation 2.01

Let us use the notation  $p \circ q$  as a shorthand for "the sum of products where each product is the coefficient of a monomial in  $p$  times the coefficient of the same monomial in  $q$ ".

#### Procedure 2.04

##### Objective

Choose two lists of  $\mathbb{Q}[x]$ s  $s, q$  in such a way that, letting  $m = |s| - 1$ ,

1. For  $i = 0$  to  $i = m$ ,  $\deg(s_i) = i$ .
2. For  $i = 0$  to  $i = m$ ,  $\text{sgn}(x^i \circ s_i) = \text{sgn}(x^m \circ s_m)$ .
3. For  $i = 1$  to  $i = m - 1$ ,  $s_{i-1} + s_{i+1} = q_i s_i$ .

Let  $x, y$  be indeterminates. The objective of the following instructions is to construct lists of  $\mathbb{Q}[x]$ s  $g, h$  such that  $g_i s_{i+1} + h_i s_i = 1$  for  $i = 1$  to  $i = m - 1$ .

##### Implementation

1. Let  $g = h = \langle \rangle$ .
2. If  $m > 2$ , do the following:
  - (a) Verify that  $q_{m-1} s_{m-1} - s_m = s_{m-2}$ .
  - (b) Execute **procedure 2.04** on  $s_{[0:m]}$  and  $q_{[1:m-1]}$  and let the tuple  $\langle \cdot, g, h \rangle$  receive.
  - (c) Verify that  $g_{m-2} s_{m-1} + h_{m-2} s_{m-2} = 1$ .
  - (d) Let  $g_{m-1} = -h_{m-2}$ .
  - (e) Let  $h_{m-1} = g_{m-2} + h_{m-2} q_{m-1}$ .
  - (f) **Therefore verify that**  $g_{m-1} s_m + h_{m-1} s_{m-1} = g_{m-2} s_{m-1} + h_{m-2} (q_{m-1} s_{m-1} - s_m) = g_{m-2} s_{m-1} + h_{m-2} s_{m-2} = 1$ .
3. Otherwise, if  $m = 2$  do the following:
  - (a) Verify that  $s_0 + s_2 = q_1 s_1$ .

- (b) Let  $g_1 = -\frac{1}{s_0}$ .
  - (c) Let  $h_1 = \frac{q_1}{s_0}$ .
  - (d) **Therefore verify that**  $g_1 s_2 + h_1 s_1 = 1$ .
4. **Yield the tuple**  $\langle s, q, g, h \rangle$ .

#### Notation 2.02

Let us use the notation  $J_s(x)$  as a shorthand for "the number of sign changes in the list  $s_0(x), s_1(x), \dots, s_{|s|-1}(x)$ ".

#### Procedure 2.05

##### Objective

Execute **procedure 2.04** and let  $\langle s, q, g, h \rangle$  receive. Execute **procedure 2.0** on  $s$  and let  $\langle G \rangle$  receive the result. Choose  $\mathbb{Q}$ s  $c$  and  $d$  in such a way that:

1.  $J_m(c)$  and  $J_m(d)$  are well defined.
2. Letting  $B = \max_{i=1}^m |G_i(c, d)|$ .
3. Letting  $C = \max_{i=1}^{m-1} \max(|g_i(c)|, |h_i(c)|, |g_i(d)|, |h_i(d)|)$ .
4. Letting  $D = \max_{i=1}^{m-1} \max(|q_i(c)|, |q_i(d)|, 2)$ .
5.  $|d - c| \leq \frac{1}{BCD}$ .

The objective of the following instructions is to show that either  $0 < 0$  or  $|J_m(d) - J_m(c)| = [\text{sgn}(s_m(c)) \neq \text{sgn}(s_m(d))]$ .

##### Implementation

1. Let  $i = 0$ .
2. Do the following:
  - (a) Verify that  $\text{sgn}(s_i(c)) = \text{sgn}(s_i(d))$ .
  - (b) Verify that  $J_i(c) = J_i(d)$ .
  - (c) If  $\text{sgn}(s_{i+1}(c)) = \text{sgn}(s_{i+1}(d))$ , do the following:
    - i. Verify that  $J_{i+1}(c) = J_{i+1}(d)$ .
    - ii. Set  $i$  to  $i + 1$  and go to (2) if the new  $i < m$ .
  - (d) Otherwise, if  $\text{sgn}(s_{i+1}(c)) \neq \text{sgn}(s_{i+1}(d))$  and  $i + 2 \leq m$ , do the following:
    - i. Execute **procedure 2.5 auxilliary procedure** on  $i$ .

ii. If  $\text{sgn}(s_{i+2}(c)) \neq \text{sgn}(s_{i+2}(d))$ , do the following:

- A. Verify that  $|s_{i+2}(c)| < |s_{i+2}(d) - s_{i+2}(c)| = |(d-c)G_{i+2}(c, d)| \leq \frac{1}{BCD} \cdot B = \frac{1}{CD} = \frac{1}{C} \cdot \frac{1}{D} \leq \frac{1}{C}(1 - \frac{1}{D})$ .
- B. Using (A) and (i), verify that  $\frac{1}{C}(1 - \frac{1}{D}) < |s_{i+2}(c)| < \frac{1}{C}(1 - \frac{1}{D})$ .

C. **Abort procedure.**

iii. Otherwise if  $\text{sgn}(s_i(c)) = \text{sgn}(s_{i+2}(c))$ , do the following:

- A. Verify that  $2\frac{1}{C}(1 - \frac{1}{D}) < |s_i(c)| + |s_{i+2}(c)| = |s_i(c) + s_{i+2}(c)| = |q_{i+1}(c)s_{i+1}(c)| < D\frac{1}{CD}$ .
- B. Verify that  $2(1 - \frac{1}{D}) < 1$ .
- C. Using (B) and the construction of  $D$ , verify that  $2 \leq D < 2$ .

D. **Abort procedure.**

iv. Otherwise, do the following:

- A. Verify that  $\text{sgn}(s_i(d)) = \text{sgn}(s_i(c)) \neq \text{sgn}(s_{i+2}(c)) = \text{sgn}(s_{i+2}(d))$ .
- B. Therefore verify that  $1 = J_{i+2}(c) - J_i(c) = J_{i+2}(d) - J_i(d)$ .
- C. Therefore verify that  $J_i(c) + 1 = J_{i+2}(c) = J_{i+2}(d) = J_i(d) + 1$ .

D. Set  $i$  to  $i + 2$  and go to (2).

(e) Otherwise, verify the following:

- i.  $\text{sgn}(s_{i+1}(c)) \neq \text{sgn}(s_{i+1}(d))$ .
- ii.  $|J_{i+1}(c) - J_{i+1}(d)| = 1$ .
- iii.  $i + 1 = m$ .

3. If  $\text{sgn}(s_m(c)) = \text{sgn}(s_m(d))$ , then do the following:

(a) Verify that  $J_m(c) = J_m(d)$ .

4. Otherwise do the following:

(a) Verify that  $|J_m(d) - J_m(c)| = 1$ .

### Auxilliary Procedure

**Objective** Choose a non-negative integer  $i < m$  such that  $\text{sgn}(s_{i+1}(c)) \neq \text{sgn}(s_{i+1}(d))$  and  $i+2 \leq m$ . The objective of the following instructions is to show

that  $|s_{i+1}(c)| < \frac{1}{CD}$ ,  $|s_{i+1}(d)| < \frac{1}{CD}$ ,  $\frac{1}{C}(1 - \frac{1}{D}) < |s_i(c)|$ ,  $\frac{1}{C}(1 - \frac{1}{D}) < |s_i(d)|$ ,  $\frac{1}{C}(1 - \frac{1}{D}) < |s_{i+2}(c)|$ , and  $\frac{1}{C}(1 - \frac{1}{D}) < |s_{i+2}(d)|$ .

### Implementation

1. Verify the following in order:

- (a)  $|s_{i+1}(c)| < |s_{i+1}(c) - s_{i+1}(d)| = |c - d||G_{i+1}(c, d)| \leq |c - d|B \leq lB = \frac{1}{CD}$
- (b)  $|s_{i+1}(d)| < |s_{i+1}(c) - s_{i+1}(d)| \leq \frac{1}{CD}$
- (c)  $1 = g_i(c)s_{i+1}(c) + h_i(c)s_i(c) = |g_i(c)s_{i+1}(c) + h_i(c)s_i(c)| \leq |g_i(c)||s_{i+1}(c)| + |h_i(c)||s_i(c)| < C(\frac{1}{CD} + |s_i(c)|)$
- (d)  $\frac{1}{C}(1 - \frac{1}{D}) < |s_i(c)|$
- (e)  $1 < C(\frac{1}{CD} + |s_i(d)|)$
- (f)  $\frac{1}{C}(1 - \frac{1}{D}) < |s_i(d)|$
- (g)  $1 = g_{i+1}(c)s_{i+2}(c) + h_{i+1}(c)s_{i+1}(c) = |g_{i+1}(c)s_{i+2}(c) + h_{i+1}(c)s_{i+1}(c)| \leq |g_{i+1}(c)||s_{i+2}(c)| + |h_{i+1}(c)||s_{i+1}(c)| < C(|s_{i+2}(c)| + \frac{1}{CD})$
- (h)  $\frac{1}{C}(1 - \frac{1}{D}) < |s_{i+2}(c)|$
- (i)  $1 < C(|s_{i+2}(d)| + \frac{1}{CD})$
- (j)  $\frac{1}{C}(1 - \frac{1}{D}) < |s_{i+2}(d)|$

### Procedure 2.06

#### Objective

Choose a  $\mathbb{Q}[x]$   $p = p_0x^t + p_1x^{t-1} + p_2x^{t-2} + \dots + p_tx^0$ , where  $p_0 \neq 0$ . Choose a  $\mathbb{Q}$   $k > 1 + \max_{i=1}^t |\frac{p_i}{p_0}|$ . The objective of the following instructions is to show that  $\text{sgn}(p(k)) = \text{sgn}(p_0)$ .

#### Implementation

1. In reverse order verify the following:

- (a)  $\text{sgn}(p_0k^n + p_1k^{n-1} + \dots + p_nk^0) = \text{sgn}(p_0)$
- (b)  $\text{sgn}(k^n + \frac{p_1}{p_0}k^{n-1} + \dots + \frac{p_n}{p_0}k^0) = 1$
- (c)  $k^n + \frac{p_1}{p_0}k^{n-1} + \dots + \frac{p_n}{p_0}k^0 > 0$
- (d)  $k^n > -(\frac{p_1}{p_0}k^{n-1} + \dots + \frac{p_n}{p_0}k^0)$
- (e)  $k^n > |\frac{p_1}{p_0}k^{n-1} + \dots + \frac{p_n}{p_0}k^0|$

- (f)  $k^n > |\max_{i=1}^t \frac{p_i}{p_0}|(k^{n-1} + \dots + k^0)|$
- (g)  $k^n > \max_{i=1}^t \frac{p_i}{p_0} \frac{k^n - 1}{k - 1}$
- (h)  $k^{n+1} - k^n > \max_{i=1}^t \frac{p_i}{p_0} (k^n - 1)$
- (i)  $k^{n+1} - (1 + \max_{i=1}^t \frac{p_i}{p_0})k^n + \max_{i=1}^t \frac{p_i}{p_0} > 0$
- (j)  $k > 1 + \max_{i=1}^t \frac{p_i}{p_0}$

## Procedure 2.07

### Objective

Choose a  $\mathbb{Q}[x]$   $p = p_0x^t + p_1x^{t-1} + p_2x^{t-2} + \dots + p_tx^0$ , where  $p_0 \neq 0$ . Choose a  $\mathbb{Q}$   $k < -(1 + \max_{i=1}^t \frac{p_i}{p_0})$ . The objective of the following instructions is to show that  $\text{sgn}(p(k)) = (-1)^t \text{sgn}(p_0)$ .

### Implementation

1. Let  $q = q_0x^t + q_1x^{t-1} + q_2x^{t-2} + \dots + q_tx^0$ , where  $q_i = (-1)^i p_i$ .
2. Verify that  $k < -(1 + \max_{i=1}^t \frac{q_i}{q_0})$ .
3. Therefore verify that  $-k > 1 + \max_{i=1}^t \frac{q_i}{q_0}$ .
4. Execute **procedure 2.06** on  $q$  and  $-k$ .
5. Hence verify that  $(-1)^t \text{sgn}(p(k))$ 
  - (a)  $= \text{sgn}((-1)^t p(k))$
  - (b)  $= \text{sgn}((-1)^t \sum_{i=0}^t p_i k^{t-i})$
  - (c)  $= \text{sgn}(\sum_{i=0}^t (-1)^i (-1)^{t-i} p_i k^{t-i})$
  - (d)  $= \text{sgn}(\sum_{i=0}^t q_i (-k)^{t-i})$
  - (e)  $= \text{sgn}(q(-k))$
  - (f)  $= \text{sgn}(q_0)$
  - (g)  $= \text{sgn}(p_0)$ .
6. **Therefore verify that**  $\text{sgn}(p(k)) = (-1)^t (-1)^t \text{sgn}(p(k)) = (-1)^t \text{sgn}(p_0)$ .

## Procedure 2.08

### Objective

Choose a list of  $\mathbb{Q}[x]$ s,  $s$ , and  $\mathbb{Q}$ s  $a, l, c$  such that  $a < c$  and  $l > 0$ . The objective of the following instructions is to either show that  $0 < 0$  or to construct a

list of  $\mathbb{Q}$ s,  $b$ , such that  $a = b_1 < b_2 < \dots < b_{|b|} = c$ ,  $b_{i+1} - b_i \leq l$  for  $i = 1$  to  $i = |b| - 1$ , and  $J_s(b)$  is defined for  $i = 1$  to  $i = |b| - 1$ .

### Implementation

1. Let  $e = \langle \langle \rangle, \langle \rangle, \dots, \langle \rangle \rangle$ .
2. Let  $f = \sum_{r=1}^{|s|} \deg(s_r)$ .
3. Let  $b = \langle a \rangle$ .
4. Let  $d = b_1$ .
5. While  $d + l < c$ , do the following:
  - (a) Let  $m = l$ .
  - (b) While  $J_s(d + m)$  is not defined and  $|e| \leq f$ , do the following:
    - i. Let  $1 \leq i \leq |s|$  be an integer such that  $s_i(d + m) = 0$ .
    - ii. Append  $d + m$  onto  $e_i$ .
    - iii. Set  $m = \frac{m}{2}$ .
  - (c) If  $\sum |e| > f$ , then do the following:
    - A. Verify that  $\sum |e| \leq f$ .
    - B. Therefore using (c), verify that  $\sum |e| \leq f < \sum |e|$ .
    - C. **Abort procedure.**
  - ii. Otherwise, do the following:
    - A. Let  $1 \leq i \leq |s|$  be an integer such that  $|e_i| > \deg(s_i)$ .
    - B. Execute **procedure 2.01** on  $s_i$  and a sorted  $e_i$ .
    - C. **Abort procedure.**
- (d) Otherwise, do the following:
  - i. **Verify that  $J_s(d + m)$  is defined.**
  - ii. Append  $d + m$  onto  $b$ .
  - iii. **Verify that  $0 < b_{|b|} - b_{|b|-1} = m \leq l$ .**
  - iv. Set  $d$  to  $d + m$ .
  - v. Using (5), verify that  $d < c$ .
6. Verify that  $d < c$ .

7. Verify that  $d + l \geq c$ .
8. **Therefore verify that**  $0 < c - d \leq l$ .
9. Append  $c$  onto  $b$ .
10. **Yield**  $\langle b \rangle$ .

## Procedure 2.09

### Objective

Execute [procedure 2.04](#) and let  $\langle s, q, g, h \rangle$  receive. Let  $m = |s| - 1$ . The objective of the following instructions is to either show that  $0 < 0$  or to construct two lists of rational numbers  $c, d$  such that  $c_1 < d_1 \leq c_2 < d_2 \leq \dots \leq c_m < d_m$  and  $\text{sgn}(s_m(c_i)) = -\text{sgn}(s_m(d_i))$  for  $i = 1$  to  $i = m$ .

### Implementation

1. Let  $U = 1 + \max_{i=0}^m \left( 1 + \max_{j=1}^i \left| \frac{x^{i-j} \circ s_i}{x^i \circ s_i} \right| \right)$
2. Using [procedure 2.06](#), verify that  $J(U) = 0$ .
3. Using [procedure 2.07](#), verify that  $J(-U) = m$ .
4. Execute [procedure 2.0](#) on  $s$  and let  $\langle G \rangle$  receive the result.
5. Let the rational  $B = \max_{i=1}^m |G_i|(U, U)$ .
6. Let  $C = \max_{i=1}^m \max(|g_i|(U), |h_i|(U))$ .
7. Let  $D = \max(3, \max_{i=1}^m |q_i|(U))$
8. Let  $l = \frac{1}{BCD}$ .
9. Execute [procedure 2.08](#) on  $s$  with endpoints  $-U, U$  and a step size of  $l$  and let  $\langle e \rangle$  receive the result.
10. Let  $c = d = \langle \rangle$ .
11. For  $i = 1$  to  $i = |e| - 1$ :
  - (a) Execute [procedure 2.05](#) on the tuple  $\langle e_i, e_{i+1} \rangle$ .
  - (b) If  $J_m(c) \neq J_m(d)$ , then do the following:
    - i. Append  $e_i$  to  $c$ .
    - ii. Append  $e_{i+1}$  to  $d$ .
    - iii. Cognizant of [procedure 2.05](#), verify that  $|J_m(d) - J_m(c)| = 1$ .
- iv. Therefore verify that  $\text{sgn}(s_m(c_{|c|})) = -\text{sgn}(s_m(d_{|d|}))$ .
- v. Also verify that  $d_{|d|-1} \leq c_{|c|} < d_{|d|}$ .
12. If less than  $m$  pairs of rational numbers were recorded, then do the following:
  - (a) Verify that each change of  $J_m(x)$  over the course of (12) was by 1.
  - (b) Verify that  $J_m(x)$  changed less than  $m$  times over the course of (12).
  - (c) Therefore verify that  $|J_m(U) - J_m(-U)| < m$ .
  - (d) Therefore using (2) and (3), verify that  $m = |J_m(U) - J_m(-U)| < m$ .
  - (e) **Abort procedure.**
13. Otherwise, do the following:
  - (a) Verify that  $m \leq |c| = |d|$ .
  - (b) **Yield the tuple**  $\langle c, d \rangle$ .

## 3 Matrix Arithmetic

### Notation 3.00

Let us use the notation  $\mathcal{M}_{m,n}(A)$  as a shorthand for " $m \times n$  matrix of  $As$ ".

### Procedure 3.0

#### Objective

Choose a  $\mathcal{M}_{m,2}(\mathbb{Q}[x])$ ,  $A$ . Let  $\deg(0) = \infty$ . Let  $k = \min(\deg(A_{1,1}), \deg(A_{1,2}))$  and  $q = \deg(A_{1,1})$ . The objective of the following instructions is to make  $A_{1,2} = 0$ ,  $\deg(A_{1,1}) \leq k$ , and either leave  $A_{*,1}$  unchanged or make  $\deg(A_{1,1}) < q$  by a sequence of operations whereby, in each step a  $\mathbb{Q}[x]$  times either of the columns is added to the other.

#### Implementation

1. Let  $A$  be our working matrix.
2. While  $A_{1,2} \neq 0$ , do the following:
  - (a) If  $\deg(A_{1,1}) \leq \deg(A_{1,2})$ , then:

- i. Subtract  $\frac{x^{\deg(A_{1,2}) \circ A_{1,2}}}{x^{\deg(A_{1,1}) \circ A_{1,1}}} x^{\deg(A_{1,2}) - \deg(A_{1,1})}$  times  $A_{1,1}$  from  $A_{1,2}$ .
- ii. Now verify that either  $A_{1,2}$ 's degree has decreased or  $A_{1,2} = 0$ .
- (b) Otherwise, do the following:
  - i. Let  $p = \frac{x^{\deg(A_{1,1}) \circ A_{1,1}}}{x^{\deg(A_{1,2}) \circ A_{1,2}}} x^{\deg(A_{1,1}) - \deg(A_{1,2})}$ .
  - ii. If  $A_{1,1} = pA_{1,2}$ , then do the following:
    - A. Add  $1 - p$  times  $A_{1,2}$  to  $A_{1,1}$ .
    - B. Verify that now  $A_{1,1} = A_{1,2}$ .
  - iii. Otherwise, do the following:
    - A. Verify that  $A_{1,1} \neq pA_{1,2}$ .
    - B. Add  $-p$  times  $A_{1,2}$  to  $A_{1,1}$ .
  - iv. Therefore verify that  $A_{1,1} \neq 0$ .
  - v. Also verify that  $A_{1,1}$ 's degree has decreased.
3. **Verify that**  $A_{1,2} = 0$ .
4. Verify that the changes to  $A_{1,1}$ , if any, have decreased its degree.
5. If sensical, do the following:
  - (a) Verify that all changes to  $A_{1,2}$  but the last have decreased its degree.
  - (b) Verify that  $\deg(A_{1,1}) \leq$  the degree of the penultimate value of  $A_{1,2}$ .
6. **Therefore verify that**  $\deg(A_{1,1}) \leq k$ .
7. If  $A_{*,1}$  was changed, then do the following:
  - (a) Verify that  $A_{1,1}$  was also changed.
  - (b) **Therefore verify that**  $\deg(A_{1,1}) < q$ .
8. **Yield the tuple**  $\langle A \rangle$ .

### Notation 3.01

Let us use the notation "diagonal" as a shorthand for "matrix positions such that the row index equals the column index".

### Notation 3.02

Let us use the notation  $\mathcal{D}_{m,n}(A)$  as a shorthand for " $\mathcal{M}_{m,n}(A)$  with 0s in all the off-diagonal positions".

## Procedure 3.01

### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to transform  $A$  into a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$  by a sequence of operations whereby either a  $\mathbb{Q}[x]$  times any of the columns is added to a different column, or a  $\mathbb{Q}[x]$  times any of the rows is added to a different row.

### Implementation

1. If  $m = 0$  or  $n = 0$ , then do the following:
    - (a) **Verify that**  $A$  is a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ .
    - (b) **Yield the tuple**  $\langle A \rangle$ .
  2. Otherwise do the following:
  3. Verify that  $m > 0$  and  $n > 0$ .
  4. Let  $A$  be our working matrix.
  5. Now do the following:
    - (a) While there are non-zero entries in the top row less its first entry, do the following:
      - i. In the first row, select the  $\mathcal{M}_{m,2}(\mathbb{Q}[x])$  whose top-right entry coincides with the last non-zero entry of the first row
      - ii. Apply **procedure 3.0** on this submatrix.
    - iii. Verify that the top-left and top-right entries of the submatrix are now non-zero and zero respectively.
    - iv. If the first column of  $A$  was modified by (5aii), then do the following:
      - A. Verify that  $\deg(A_{1,1})$  decreased.
      - B. Go back to (5).
  - (b) Now do the same operations as in (a), but this time with the operations themselves reflected across the matrix's diagonal.
6. Verify that, except for the top-left entry, the first row and the first column are zero.
7. Apply **procedure 3.01** on the submatrix  $A_{[2:m+1],[2:n+1]}$ .
8. Verify that (7)'s execution leaves the first row and column unchanged.

9. Verify that  $A$  is now a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ .

10. Yield the tuple  $\langle A \rangle$ .

### Procedure 3.02

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ , a  $\mathcal{M}_{n,p}(\mathbb{Q}[x])$ ,  $B$ , and a  $\mathcal{M}_{p,q}(\mathbb{Q}[x])$ ,  $C$ . The objective of the following instructions is to show that  $(AB)C = A(BC)$ .

#### Implementation

1. Verify that  $(AB)_{i,l} = \sum_{k=1}^n (A_{i,k} * B_{k,l})$  for  $1 \leq i \leq m$ , for  $1 \leq l \leq p$ .
2. Verify that  $((AB)C)_{i,r} = \sum_{l=1}^p ((AB)_{i,l} * C_{l,r}) = \sum_{l=1}^p (\sum_{k=1}^n (A_{i,k} * B_{k,l}) * C_{l,r})$  for  $1 \leq i \leq m$ , for  $1 \leq r \leq q$ .
3. Verify that  $(BC)_{k,r} = \sum_{l=1}^p (B_{k,l} * C_{l,r})$  for  $1 \leq k \leq n$ , for  $1 \leq r \leq q$ .
4. Verify that  $(A(BC))_{i,r} = \sum_{k=1}^n (A_{i,k} * (BC)_{k,r}) = \sum_{k=1}^n (A_{i,k} * \sum_{l=1}^p (B_{k,l} * C_{l,r}))$  for  $1 \leq i \leq m$ , for  $1 \leq r \leq q$ .
5. Therefore Verify that  $(2) = \sum_{l=1}^p (\sum_{k=1}^n (A_{i,k} * B_{k,l} * C_{l,r})) = \sum_{k=1}^n (\sum_{l=1}^p (A_{i,k} * B_{k,l} * C_{l,r})) = \sum_{k=1}^n (A_{i,k} * \sum_{l=1}^p (B_{k,l} * C_{l,r})) = (4)$  for  $1 \leq i \leq m$ , for  $1 \leq r \leq q$ .
6. Therefore verify that  $(AB)C = A(BC)$ .

#### Notation 3.03

Let us use the notation  $I_n$  as a shorthand for "the  $\mathcal{M}_{n,n}(\mathbb{Q})$  with only 1s on the diagonal and 0s everywhere else".

#### Notation 3.04

Let us use the notation  $\mathcal{T}_m(\mathbb{Q}[x])$  as a shorthand for " $\mathcal{M}_{m,m}(\mathbb{Q}[x])$  with only 1s on the diagonal, a single  $\mathbb{Q}[x]$  off the diagonal, and 0s everywhere else".

### Procedure 3.03

#### Objective

Choose a procedure,  $A$ , and two non-negative integers  $m, n$ . The objective of the following instructions is to construct a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ s,  $M$ , and a list of  $\mathcal{T}_n(\mathbb{Q}[x])$ s,  $N$  such that  $M_{|M|+1-i}$  equals  $I_m$  after applying the  $i^{th}$  row operation carried out by  $A$  also on it, and  $N_i$  equals  $I_n$  after applying the  $i^{th}$  row operation carried out by  $A$  also on it.

#### Implementation

1. Make an empty list,  $N$ .
2. Augment procedure  $A$  so that each time a polynomial  $x$  times a column  $i$  is added onto column  $j$ , an  $n \times n$  matrix that only has 1s on its diagonal, and such that the only non-zero entry off its diagonal is  $x$  at position  $(i, j)$ , is appended onto  $N$ .
3. Make an empty list,  $M$ .
4. Also augment procedure  $A$  so that each time a polynomial  $x$  times a row  $i$  is added onto row  $j$ , an  $n \times n$  matrix that only has 1s on its diagonal, and such that the only non-zero entry off its diagonal is  $x$  at position  $(j, i)$ , is prepended onto  $M$ .
5. Now run procedure  $A$ .
6. Yield the tuple  $\langle M, N \rangle$ .

### Procedure 3.04

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to show that  $I_m A = A = A I_n$ .

#### Implementation

1. For  $1 \leq r \leq m$ , do the following:
  - (a) For  $1 \leq t \leq n$ , do the following:
    - i. Verify that  $(I_m A)_{r,t} = \sum_{u=1}^m (I_m)_{r,u} A_{u,t} = (I_m)_{r,r} A_{r,t} = 1 * A_{r,t} = A_{r,t}$ .
2. Therefore verify that  $I_m A = A$ .

3. For  $1 \leq r \leq m$ , do the following:
  - (a) For  $1 \leq t \leq n$ , do the following:
    - i. Verify that  $(AI_n)_{r,t} = \sum_{u=1}^m A_{r,u}(I_n)_{u,t} = A_{r,t}(I_n)_{t,t} = A_{r,t} * 1 = A_{r,t}$ .
4. Therefore verify that  $AI_n = A$ .

### Notation 3.05

Let us use the notation  $A^{-1}$  as a shorthand for the result yielded by executing [procedure 3.05](#) on  $A$ .

### Procedure 3.05

#### Objective

Choose a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to construct a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $A^{-1}$ , such that  $A * A^{-1} = I_m$ .

#### Implementation

1. Let  $A^{-1}$  be  $\langle \rangle$ .
2. For  $i = 1$  to  $i = |A|$ , do the following:
  - (a) Let  $(j, k)$  be the position of the off diagonal entry of  $A_i$ .
  - (b) Let  $B$  equal  $A_i$  but with entry  $(j, k)$  negated.
  - (c) For  $1 \leq r \leq m$  and  $r \neq j$ , do the following:
    - i. For  $1 \leq t \leq m$ , do the following:
      - A. Verify that  $(A_i B)_{r,t} = \sum_{u=1}^m (A_i)_{r,u} B_{u,t} = (A_i)_{r,r} B_{r,t} = 1 * B_{r,t} = [r = t]$ .
  - (d) For  $1 \leq t \leq m$  and  $t \neq k$ , do the following:
    - i. Verify that  $(A_i B)_{j,t} = \sum_{u=1}^m (A_i)_{j,u} B_{u,t} = (A_i)_{j,t} B_{t,t} = (A_i)_{j,t} * 1 = [j = t]$ .
  - (e) Verify that  $(A_i B)_{j,k} = \sum_{u=1}^m (A_i)_{j,u} B_{u,k} = (A_i)_{j,j} B_{j,k} + (A_i)_{j,k} B_{k,k} = 1 * B_{j,k} + (A_i)_{j,k} * 1 = B_{j,k} + (A_i)_{j,k} = 0$ .
  - (f) Therefore verify that  $A_i B = I_m$ .
  - (g) Now prepend  $B$  onto  $A^{-1}$ .
3. Verify that  $|A| = |A^{-1}|$ .

4. Therefore using [procedure 3.02](#) and [procedure 3.04](#), verify that  $A * A^{-1} = I_m$ .

- (a)  $= A_1 \cdots A_{|A|-1} A_{|A|} A^{-1}_1 A^{-1}_2 \cdots A^{-1}_{|A|}$
- (b)  $= A_1 \cdots A_{|A|-2} A_{|A|-1} I_m A^{-1}_2 A^{-1}_3 \cdots A^{-1}_{|A|}$
- (c)  $= A_1 \cdots A_{|A|-2} A_{|A|-1} A^{-1}_2 A^{-1}_3 \cdots A^{-1}_{|A|}$
- (d)  $\vdots$
- (e)  $= A_1 I_m A^{-1}_{|A|}$
- (f)  $= A_1 A^{-1}_{|A|}$
- (g)  $= I_m$ .

### Procedure 3.06

#### Objective

Choose a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to show that  $(A^{-1})^{-1} = A$  and  $A^{-1} * A = I_m$ .

#### Implementation

1. Verify that  $(A^{-1})^{-1} = A$ .
2. Therefore using [procedure 3.05](#), verify that  $A^{-1} * A = A^{-1} * (A^{-1})^{-1} = I_m$ .

### Procedure 3.07

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to define the polynomials  $u_1, u_2, \dots, u_{\min(m,n)}$  and transform  $A$  into a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$  such that  $A_{k,k} = u_k A_{1,1}$  for  $1 \leq k \leq \min(m,n)$  by a sequence of operations whereby either a  $\mathbb{Q}[x]$  times any of the columns is added to a different column, or a  $\mathbb{Q}[x]$  times any of the rows is added to a different row.

#### Implementation

1. Let  $u = \langle 1 \rangle$ .
2. For  $j$  going from 2 to  $\min(m,n)$ , do the following:



- (a) Verify that  $A_{k,k} = u_k A_{1,1}$  for  $k = 1$  to  $k = |u|$ .
  - (b) Add row  $j$  to row 1.
  - (c) Now verify that  $A_{1,j} = A_{j,j}$ .
  - (d) Set  $A' = A$  and let  $A'$  be our working matrix.
  - (e) Let  $\langle M, N \rangle$  receive the results of executing **procedure 3.03** on the pair  $\langle m, n \rangle$  and the following procedure:
    - i. Execute **procedure 3.0** on the submatrix of  $A'$  formed by selecting row 1 and columns 1 and  $j$  as if there were nothing in between.
  - (f) Now verify that:
    - i.  $M$  is empty.
    - ii.  $AN_* = M_*AN_* = A'$ .
    - iii.  $A = AI_n = AN_*N^{-1}_* = A'N^{-1}_*$ .
    - iv.  $A'_{1,j} = 0$ .
    - v.  $A_{1,1} = A'_{1,1}N^{-1}_{*1,1} + A'_{1,j}N^{-1}_{*j,1} = A'_{1,1}N^{-1}_{*1,1}$ .
    - vi.  $A_{j,j} = A_{1,j} = A'_{1,1}N^{-1}_{*1,j} + A'_{1,j}N^{-1}_{*j,j} = A'_{1,1}N^{-1}_{*1,j}$ .
    - vii.  $A_{j,1} = 0$ .
    - viii.  $A'_{j,1} = A_{j,1}N_{*1,1} + A_{j,j}N_{*j,1} = A_{j,j}N_{*j,1} = A'_{1,1}N^{-1}_{*1,j}N_{*j,1}$ .
    - ix.  $A'_{j,j} = A_{j,1}N_{*1,j} + A_{j,j}N_{*j,j} = A_{j,j}N_{*j,j} = A'_{1,1}N^{-1}_{*1,j}N_{*j,j}$ .
  - (g) Subtract  $N^{-1}_{*1,j}N_{*j,1}$  times row 1 from row  $j$ .
  - (h) Now verify that  $A'_{j,1} = 0$ .
  - (i) For  $k = 2$  to  $k = |u|$ , do the following:
    - i. Verify that  $A'_{k,k} = A_{k,k} = u_k A_{1,1} = u_k A'_{1,1}N^{-1}_{*1,1}$ .
    - ii. Set  $u_k = u_k N^{-1}_{*1,1}$ .
    - iii. Hence verify that  $A'_{k,k} = u_k A'_{1,1}$ .
  - (j) Let  $u_j = N^{-1}_{*1,j}N_{*j,j}$ .
  - (k) Hence verify that  $A'_{j,j} = u_j A'_{1,1}$ .
  - (l) Now let  $A = A'$ .
- 3. Hence verify that  $A'_{k,k} = u_k A'_{1,1}$  for  $k = 1$  to  $k = \min(m, n)$ .**

4. **Yield  $\langle u \rangle$ .**

### Procedure 3.08

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ , and a  $\mathcal{M}_{n,k}(\mathbb{Q}[x])$ ,  $B$ . Choose integers  $1 \leq a \leq m$ ,  $1 \leq b \leq n$ , and  $1 \leq c \leq k$ . The objective of the following instructions is to show that  $(AB)_{[1:a],[1:c]} = A_{[1:a],[1:b]}B_{[1:b],[1:c]} + A_{[1:a],[b:n+1]}B_{[b:n+1],[1:c]}$ .

#### Implementation

1. Multiply matrix  $A$  by matrix  $B$ .
2. For each  $1 \leq i \leq a - 1$ , do the following:
  - (a) For each  $1 \leq j \leq c - 1$ , do the following:
    - i. Verify that  $(AB)_{i,j} = \sum_{p=1}^n A_{i,p}B_{p,j} = \sum_{p=1}^{b-1} A_{i,p}B_{p,j} + \sum_{p=b}^n A_{i,p}B_{p,j} = \sum_{p=1}^{b-1} (A_{[1:a],[1:b]})_{i,p} (B_{[1:b],[1:c]})_{p,j} + \sum_{p=1}^{1+n-b} (A_{[1:a],[b:n+1]})_{i,p} (B_{[b:n+1],[1:c]})_{p,j} = (A_{[1:a],[1:b]}B_{[1:b],[1:c]})_{i,j} + (A_{[1:a],[b:n+1]}B_{[b:n+1],[1:c]})_{i,j}$ .
3. **Therefore verify that  $(AB)_{[1:a],[1:c]} = A_{[1:a],[1:b]}B_{[1:b],[1:c]} + A_{[1:a],[b:n+1]}B_{[b:n+1],[1:c]}$ .**
4. **Do similar computations to verify that the other three blocks of  $AB$  are computed in an analogous way to multiplying two  $2 \times 2$  matrices.**

### Procedure 3.09

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Let  $A_{0,0} = 1$ . The objective of the following instructions is to define the polynomials  $v_1, v_2, \dots, v_{\min(m,n)}$  and transform  $A$  into a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$  such that  $A_{k,k} = v_k A_{k-1,k-1}$  for  $1 \leq k \leq \min(m, n)$  by a sequence of operations whereby either a  $\mathbb{Q}[x]$  times any of the columns is added to a different column, or a  $\mathbb{Q}[x]$  times any of the rows is added to a different row.

## Implementation

1. Apply **procedure 3.01** on matrix  $A$ .
2. Let  $v = \langle \rangle$ .
3. Let  $p = \langle A_{1,1}, A_{2,2}, \dots, A_{\min(m,n), \min(m,n)} \rangle$ .
4. For  $j$  going from 1 to  $\min(m, n)$ , do the following:
  - (a) Set  $A' = A$ .
  - (b) Let  $\langle M, N \rangle$  receive the results of executing **procedure 3.03** on the pair  $\langle m, n \rangle$  and the following procedure:
    - i. Apply **procedure 3.07** on the submatrix of  $A'$  containing rows  $j$  to  $m$  and columns  $j$  to  $n$ , and let  $\langle u \rangle$  receive.
  - (c) Verify that  $A'_{k,k} = u_{k+1-j} A'_{j,j}$  for  $k = j$  to  $k = \min(m, n)$ .
  - (d) **Verify that  $A'$  is the same as  $A$  modulo the submatrix spanning rows  $j$  to  $m$  and columns  $j$  to  $n$ .**
  - (e) Verify that  $M_i$  is the same as  $I_m$  modulo the submatrix spanning rows  $j$  to  $m$  and columns  $j$  to  $m$ , for  $i = 1$  to  $|M|$ .
  - (f) Therefore verify that  $M_*$  is the same as  $I_m$  modulo the submatrix spanning rows  $j$  to  $m$  and columns  $j$  to  $m$ .
  - (g) Verify that  $N_i$  is the same as  $I_n$  modulo the submatrix spanning rows  $j$  to  $n$  and columns  $j$  to  $n$ , for  $i = 1$  to  $|N|$ .
  - (h) Therefore verify that  $N_*$  is the same as  $I_n$  modulo the submatrix spanning rows  $j$  to  $n$  and columns  $j$  to  $n$ .
  - (i) Verify that  $A' = M_* A N_*$ .
  - (j) Let  $v_j = \sum_{r=j}^{\min(m,n)} (M_*)_{j,r} p_{r+1-j} (N_*)_{r,j}$ .
  - (k) Hence using (f), (h), and (i), verify that  $A'_{j,j}$ 
    - i.  $= (M_* A N_*)_{j,j}$
    - ii.  $= \sum_{r=1}^m (M_*)_{j,r} (A N_*)_{r,j}$
    - iii.  $= \sum_{r=1}^{\min(m,n)} (M_*)_{j,r} (A N_*)_{r,j}$
    - iv.  $= \sum_{r=1}^{\min(m,n)} (M_*)_{j,r} A_{r,r} (N_*)_{r,j}$
    - v.  $= \sum_{r=j}^{\min(m,n)} (M_*)_{j,r} A_{r,r} (N_*)_{r,j}$
    - vi.  $= \sum_{r=j}^{\min(m,n)} (M_*)_{j,r} A_{j-1,j-1} p_{r+1-j} (N_*)_{r,j}$

$$\text{vii.} = A_{j-1,j-1} \sum_{r=j}^{\min(m,n)} (M_*)_{j,r} p_{r+1-j} (N_*)_{r,j}$$

$$\text{viii.} = A'_{j-1,j-1} \sum_{r=j}^{\min(m,n)} (M_*)_{j,r} p_{r+1-j} (N_*)_{r,j}$$

$$\text{ix.} = A'_{j-1,j-1} v_j.$$

(l) Set  $A$  to  $A'$ .

(m) Set  $p$  to  $u_{2:|u|}$ .

5. **Yield the tuple  $\langle v \rangle$ .**

## Notation 3.06

Let us use the notation  $\det(A)$  as a shorthand for the result yielded by executing **procedure 3.10** on  $A$ .

## Procedure 3.10

### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to construct a  $\mathbb{Q}[x]$ ,  $\det(A)$ .

## Implementation

1. If  $m = 0$ , then do the following:

(a) **Yield the tuple  $\langle 1 \rangle$ .**

2. Otherwise, do the following:

(a) **Yield the tuple**  
 $\langle \sum_{r=1}^m (-1)^{r-1} A_{r,1} \det(A_{[1:r] \setminus [r+1, m+1], [2:m+1]}) \rangle$ .

## Procedure 3.11

### Objective

Choose a  $\mathbb{Q}[x]$   $p$ . Choose two  $\mathcal{M}_{m,1}(\mathbb{Q}[x])$ s,  $B$  and  $C$ . Choose an integer  $0 < i \leq m$ . Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ , such that its  $i^{\text{th}}$  column is  $B + pC$ . Let  $A'$  be  $A$  but with the  $i^{\text{th}}$  column replaced by  $B$  and let  $A'''$  be  $A$  but with the  $i^{\text{th}}$  column replaced by  $C$ . The objective of the following instructions is to show that  $\det(A) = \det(A') + p \det(A''')$ .

## Implementation

1. If  $i = 1$ , then verify that  $\det(A)$

$$\begin{aligned}
 (a) &= \frac{\sum_{r=1}^m (-1)^{r-1} A_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 (b) &= \frac{\sum_{r=1}^m (-1)^{r-1} (B + pC)_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 (c) &= \frac{\sum_{r=1}^m (-1)^{r-1} (B)_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} + \frac{\sum_{r=1}^m (-1)^{r-1} (pC)_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 (d) &= \frac{\sum_{r=1}^m (-1)^{r-1} (B)_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} + \frac{p \sum_{r=1}^m (-1)^{r-1} (C)_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 (e) &= \frac{\sum_{r=1}^m (-1)^{r-1} (A')_{r,1}}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})} + \frac{p \sum_{r=1}^m (-1)^{r-1} (A''')_{r,1}}{\det(A'''_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 (f) &= \det(A') + p \det(A''')
 \end{aligned}$$

2. Otherwise, do the following:

$$\begin{aligned}
 (a) \text{ For } r = 1 \text{ to } r = m, \text{ do the following:} \\
 \text{i. Execute } \text{procedure } 3.11 \text{ on } \langle p, B_{[1:r] \cap [r+1:m+1], 1}, C_{[1:r] \cap [r+1:m+1], 1}, i - 1, A_{[1:r] \cap [r+1:m+1], [2:m+1]} \rangle. \\
 \text{ii. Therefore verify that} \\
 \frac{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})} = \frac{p \det(A'''_{[1:r] \cap [r+1:m+1], [2:m+1]})}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})}.
 \end{aligned}$$

(b) Therefore using (a), verify that  $\det(A)$

$$\begin{aligned}
 \text{i.} &= \frac{\sum_{r=1}^m (-1)^{r-1} A_{r,1}}{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})} \cdot \\
 \text{ii.} &= \frac{\sum_{r=1}^m (-1)^{r-1} A_{r,1}}{(\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]}) + p \det(A'''_{[1:r] \cap [r+1:m+1], [2:m+1]}))} \cdot \\
 \text{iii.} &= \frac{\sum_{r=1}^m (-1)^{r-1} A'_{r,1}}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]}) + \sum_{r=1}^m (-1)^{r-1} A'''_{r,1}} \cdot \\
 \text{iv.} &= \det(A') + p \det(A''').
 \end{aligned}$$

**Make an analogous procedure for cases when a given row is the sum of two  $1 \times m$  matrices.**

## Procedure 3.12

### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ . Choose a row  $1 < i \leq m$ . Let  $A'$  be  $A$  with columns  $i-1$  and  $i$  swapped. The objective of the following instructions is to show that  $\det(A') = -\det(A)$ .

### Implementation

1. If  $i = 2$ , then verify that  $\det(A)$

$$\begin{aligned}
 (a) &= \sum_{r=1}^m (-1)^{r-1} A_{r,1} \det(A_{[1:r] \cap [r+1:m+1], [2:m+1]}) \\
 (b) &= \sum_{r=1}^m (-1)^{r-1} A_{r,1} \sum_{t=r+1}^m (-1)^{t-2} A_{t,2} \cdot \frac{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\sum_{t=1}^m (-1)^{t-1} A_{t,1} \sum_{r=1}^{t-1} (-1)^{r-1} A_{r,2}} \cdot \frac{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})} \\
 (c) &= \sum_{t=1}^m (-1)^{t-2} A_{t,2} \sum_{r=1}^{t-1} (-1)^{r-1} A_{r,1} \cdot \frac{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\sum_{r=1}^m (-1)^{r-1} A_{r,2} \sum_{t=r+1}^m (-1)^{t-1} A_{t,1}} \cdot \frac{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\det(A_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})} \\
 (d) &= \sum_{t=1}^m (-1)^{t-2} A'_{t,1} \sum_{r=1}^{t-1} (-1)^{r-1} A'_{r,2} \cdot \frac{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\sum_{r=1}^m (-1)^{r-1} A'_{r,1} \sum_{t=r+1}^m (-1)^{t-1} A'_{t,2}} \cdot \frac{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})} \\
 (e) &= -(\sum_{r=1}^m (-1)^{r-1} A'_{r,1} \sum_{t=r+1}^m (-1)^{t-2} A'_{t,2} \cdot \frac{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\sum_{t=1}^m (-1)^{t-1} A'_{t,1} \sum_{r=1}^{t-1} (-1)^{r-1} A'_{r,2}} \cdot \frac{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}{\det(A'_{[1:r] \cap [r+1:t] \cap [t+1:m+1], [3:m+1]})}) \\
 (f) &= -\det(A').
 \end{aligned}$$

2. Otherwise do the following:

$$\begin{aligned}
 (a) \text{ Verify that } i > 2. \\
 (b) \text{ For } r = 1 \text{ to } r = m, \text{ do the following:} \\
 \text{i. Execute } \text{procedure } 3.12 \text{ on } \langle i - 1, A_{[1:r] \cap [r+1:m+1], [2:m+1]} \rangle. \\
 \text{ii. Therefore verify that} \\
 \frac{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})} = \frac{-\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})}{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})}. \\
 (c) \text{ Therefore using (b), verify that} \\
 \det(A) = \sum_{r=1}^m (-1)^{r-1} A_{r,1} \cdot \frac{\det(A_{[1:r] \cap [r+1:m+1], [2:m+1]})}{\sum_{r=1}^m (-1)^{r-1} A'_{r,1}} = \sum_{r=1}^m (-1)^{r-1} A'_{r,1} \cdot \frac{\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})}{(-\det(A'_{[1:r] \cap [r+1:m+1], [2:m+1]})} = -\det(A').
 \end{aligned}$$

Make an analogous procedure to verify that row swaps cause sign alternations.

### Procedure 3.13

#### Objective

Choose integers  $1 < i \leq m$ . Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ , such that columns  $i-1$  and  $i$  are the same. The objective of the following instructions is to show that  $\det(A) = 0$ .

#### Implementation

1. If  $i = 2$ , then verify that  $\det(A)$ 
  - (a)  $= \sum_{r=1}^m (-1)^{r-1} A_{r,1} \det(A_{[1:r] \setminus [r+1:m+1], [2:m+1]})$
  - (b)  $= \sum_{r=1}^m (-1)^{r-1} A_{r,1} \sum_{t=r+1}^m (-1)^{t-2} A_{t,2} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]}) + \sum_{t=1}^m (-1)^{t-1} A_{t,1} \sum_{r=1}^{t-1} (-1)^{r-1} A_{r,2} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]})$
  - (c)  $= \sum_{r=1}^m (-1)^{r-1} A_{r,1} \sum_{t=r+1}^m (-1)^{t-2} A_{t,2} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]}) + \sum_{r=1}^m (-1)^{r-1} A_{r,2} \sum_{t=r+1}^m (-1)^{t-1} A_{t,1} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]})$
  - (d)  $= - \sum_{r=1}^m (-1)^{r-1} A_{r,1} \sum_{t=r+1}^m (-1)^{t-1} A_{t,2} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]}) + \sum_{r=1}^m (-1)^{r-1} A_{r,1} \sum_{t=r+1}^m (-1)^{t-1} A_{t,2} \cdot \det(A_{[1:r] \setminus [r+1:t] \setminus [t+1:m+1], [3:m+1]})$
  - (e)  $= 0$ .
2. Otherwise do the following:
  - (a) Verify that  $i > 2$ .
  - (b) For  $r = 1$  to  $r = m$ , do the following:
    - i. Execute **procedure 3.13** on  $\langle i-1, A_{[1:r] \setminus [r+1:m+1], [2:m+1]} \rangle$ .
    - ii. Therefore verify that  $\det(A_{[1:r] \setminus [r+1:m+1], [2:m+1]}) = 0$ .
  - (c) Therefore using (b), verify that  $\det(A) = \sum_{r=1}^m (-1)^{r-1} A_{r,1} \det(A_{[1:r] \setminus [r+1:m+1], [2:m+1]}) = \sum_{r=1}^m (-1)^{r-1} A_{r,1} * 0 = 0$ .

Make an analogous procedure to verify that matrix choices with repeated rows yield determinants equal to zero.

### Procedure 3.14

#### Objective

Choose integers  $1 \leq i \leq m$ . Choose an integer  $0 < j \leq m-i$ . Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ . Let  $A'$  be  $A$  but with column  $i$  moved  $j$  places. The objective of the following instructions is to show that  $\det(A') = (-1)^j \det(A)$ .

#### Implementation

1. Let  $A_i = A$ .
2. For  $k = i+1$  to  $k = i+j$ , do the following:
  - (a) Let  $A_k$  be obtained by swapping columns  $k-1$  and  $k$  of  $A_{k-1}$ .
  - (b) Using **procedure 3.12**, verify that  $\det(A_k) = -\det(A_{k-1})$ .
3. Verify that  $A' = A_{i+j}$ .
4. **Therefore verify that**  $\det(A') = \det(A_{i+j}) = (-1)^1 \det(A_{i+j-1}) = \dots = (-1)^j \det(A_i) = (-1)^j \det(A)$ .

Make an analogous procedure that verifies that  $\det(A') = (-1)^j \det(A)$  when a non-positive integer,  $j$ , is chosen.

Also make an analogous procedure that does the verification for moved rows.

### Procedure 3.15

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ , and choose an integer  $0 \leq k \leq \min(m, n)$ . The objective of the following instructions is to construct a  $\mathcal{M}_{\binom{m}{k}, \binom{n}{k}}(\mathbb{Q}[x])$ ,  $C_k(A)$ .

#### Implementation

1. Yield a tuple comprising the  $\binom{m}{k} \times \binom{n}{k}$  matrix constructed as follows:
  - (a) The rows are labeled by the colexicographically sorted list of increasing length- $k$  sequences whose elements are picked from the first  $m$  positive integers.

- (b) The columns are labeled by the colexicographically sorted list of increasing length- $k$  sequences whose elements are picked from the first  $n$  positive integers.
- (c) For each row label  $I$ : For each column label  $J$ : Let the entry at position  $(I, J)$  be  $\det(A_{I,J})$ .

### Notation 3.07

We will use the notation  $C_k(A)$  to refer to the result yielded by executing **procedure 3.15** on the matrix  $A$  and integer  $k$ .

### Notation 3.08

We will use the notation  $A_{I,J}$  to refer to the entry of  $A$  with row label  $I$  and column label  $J$ .

## Procedure 3.16

### Objective

Choose two integers  $0 \leq k \leq m$ . The objective of the following instructions is to show that  $C_k(I_m) = I_{\binom{m}{k}}$ .

### Implementation

1. For each row label  $I$  of  $C_k(I_m)$ , for each column label  $J$  of  $C_k(I_m)$ , do the following:
  - (a) If the  $I = J$ , then do the following:
    - i. Verify that  $((I_m)_{I,J})_{i,j} = ((I_m)_{J,J})_{i,j} = (I_m)_{J_i,J_j} = [J_i = J_j] = [i = j]$  for  $1 \leq i \leq k$ , for  $1 \leq j \leq k$ .
    - ii. Therefore verify that  $(C_k(I_m))_{I,J} = I_k$ .
    - iii. **Therefore using procedure 3.10, verify that  $(C_k(I_m))_{I,J} = \det((I_m)_{I,J}) = \det(I_k) = 1$ .**
  - (b) Otherwise, do the following:
    - i. Verify that  $I \neq J$ .
    - ii. Let  $i$  be the index of an element of  $I$  that is not an element of  $J$ .
    - iii. Now verify that  $(I_m)_{I_i,j} = [I_i = j] = 0$ , for each  $j$  in  $J$ .

iv. Therefore verify that  $((I_m)_{I,J})_{i,*} = 0_{1 \times k}$ .

v. **Therefore using procedure 3.10, verify that  $(C_k(I_m))_{I,J} = \det((I_m)_{I,J}) = 0$ .**

2. **Therefore verify that  $C_k(I_m) = I_{\binom{m}{k}}$ .**

## Procedure 3.17

### Objective

Choose an integer  $1 \leq k \leq \min(m, n)$ . Choose a  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $A$ , such that the off diagonal entry is the  $\mathbb{Q}[x]$   $p$  at  $(i, j)$ . Also choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $B$ . The objective of the following instructions is to construct a  $\mathcal{M}_{\binom{m}{k}, \binom{m}{k}}(\mathbb{Q}[x])$   $D$  such that  $C_k(AB) = DC_k(B)$ .

### Implementation

1. Let  $D = C_k(I_m) = I_{\binom{m}{k}}$ .
2. Verify that  $AB$  equals  $B$ , but with its row  $i$  having  $p$  times  $B$ 's row  $j$  added to it.
3. Go through the row labels,  $I$ , of  $C_k(AB)$  and do the following:
  - (a) If  $i \notin I$ , then do the following:
    - i. Verify that  $(AB)_{I,[1:n+1]} = B_{I,[1:n+1]}$ .
    - ii. Therefore for each column label  $J$ , verify that  $C_k(AB)_{I,J} = \det((AB)_{I,J}) = \det(B_{I,J}) = C_k(B)_{I,J}$ .
    - iii. **Therefore verify that  $(C_k(AB))_{I,*} = (C_k(B))_{I,*}$ .**
  - (b) Otherwise, if  $i \in I$ , then:
    - i. Let  $I'$  be  $I$  but with an in-place replacement of  $i$  by  $j$ .
    - ii. For each column label  $J$ : Using **procedure 3.11**, verify that  $C_k(AB)_{I,J} = \det((AB)_{I,J}) = \det(B_{I,J}) + p * \det(B_{I',J})$ .
    - iii. If  $j \in I$ , then do the following:
      - A. Verify that the sequence  $I'$  contains two  $j$ s.
      - B. For each column label  $J$ : Using **procedure 3.13** verify that  $\det(B_{I',J}) = 0$ .

- C. Therefore for each column label  $J$ : verify that  $C_k(AB)_{\underline{I}, \underline{J}} = \det(B_{I, J}) = C_k(B)_{\underline{I}, \underline{J}}$ .
- D. **Therefore verify that**  $C_k(AB)_{\underline{I}, *} = C_k(B)_{\underline{I}, *}$ .
- iv. Otherwise if  $j \notin I$ , do the following:
- A. Let  $l$  be the signed number of places that the  $j$  introduced above needs to be moved in order to make  $I'$  an increasing sequence.
- B. Let  $I''$  be obtained from  $I'$  by moving the integer  $j$  in  $I'$  by  $l$  places.
- C. For each column label  $J$ : Using **procedure 3.14**, verify that  $\det(B_{I', J}) = (-1)^l \det(B_{I'', J})$ .
- D. Therefore for each column label  $J$ : Verify that  $C_k(AB)_{\underline{I}, \underline{J}} = \det(B_{I, J}) + p * \det(B_{I', J}) = \det(B_{I, J}) + (-1)^l p * \det(B_{I'', J})$ .
- E. Verify that  $I''$  is a row label of  $C_k(B)$ .
- F. Therefore for each column label  $J$ : Verify that  $C_k(AB)_{\underline{I}, \underline{J}} = \det(B_{I, J}) + (-1)^l p * \det(B_{I'', J}) = C_k(B)_{\underline{I}, \underline{J}} + (-1)^l p * C_k(B)_{\underline{I}'', \underline{J}}$ .
- G. **Therefore verify that**  $(C_k(AB))_{\underline{I}, *} = (C_k(B))_{\underline{I}, *} + (-1)^l p (C_k(B))_{\underline{I}'', *}$ .
- H. **Set**  $D_{\underline{I}, \underline{I}'}$  **to**  $(-1)^l p$ .
- (c) **Therefore verify that**  $C_k(AB)_{\underline{I}, *} = D_{\underline{I}, *} C_k(B)$ .
4. **Therefore verify that**  $C_k(AB) = DC_k(B)$ .
2. Verify that  $AB$  equals  $B_{[1:\min(m,n)+1], [1:n+1]}$  with each row  $i$  multiplied by  $A_{i,i}$ .
3. Go through the row labels,  $I$ , of  $C_k(AB)$  and do the following:
- (a) If  $I_k \leq \min(m, n)$ , then do the following:
- i. Using **procedure 3.15**, verify that every element of  $I$  is less than or equal to  $\min(m, n)$ .
- ii. Let  $A_0 = A$ .
- iii. For  $i = 1$  to  $i = k$ : Let  $A_i$  equal  $A_{i-1}$  but with position  $(I_i, I_i)$  set to 1.
- iv. For each column label  $J$ : Repeatedly using **procedure 3.11**, verify that  $C_k(AB)_{I, J}$
- A.  $= \det((AB)_{I, J})$
- B.  $= \det((A_0 B)_{I, J})$
- C.  $= A_{I_1, I_1} \det((A_1 B)_{I, J})$
- D.  $= A_{I_1, I_1} A_{I_2, I_2} \det((A_2 B)_{I, J})$
- E.  $\vdots$
- F.  $= A_{I_1, I_1} A_{I_2, I_2} \cdots A_{I_k, I_k} \det((A_k B)_{I, J})$
- G.  $= A_{I_1, I_1} A_{I_2, I_2} \cdots A_{I_k, I_k} \det(B_{I, J})$
- H.  $= A_{I_1, I_1} A_{I_2, I_2} \cdots A_{I_k, I_k} C_k(B)_{I, J}$ .
- v. **Therefore verify that**  $(C_k(AB))_{I, *} = A_{I_1, I_1} A_{I_2, I_2} \cdots A_{I_k, I_k} * (C_k(B))_{I, *}$ .
- vi. **Set**  $D_{\underline{I}, \underline{I}}$  **to**  $A_{I_1, I_1} A_{I_2, I_2} \cdots A_{I_k, I_k}$ .
- (b) Otherwise if  $I_k > \min(m, n)$ , then do the following:
- i. Verify that  $A_{I_k, *} = 0_{1 \times n}$ .
- ii. Therefore verify that  $(AB)_{I_k, *} = 0_{1 \times n}$ .
- iii. Therefore verify that  $((AB)_{I, *})_{k, *} = 0_{1 \times n}$ .
- iv. Therefore using **procedure 3.10**, for each column label  $J$ : verify that  $C_k(AB)_{I, J} = \det((AB)_{I, J}) = 0$ .
- v. **Therefore verify that**  $(C_k(AB))_{I, *}$  **is zero**.
- (c) **Therefore verify that**  $C_k(AB)_{\underline{I}, *} = D_{\underline{I}, *} C_k(B)$ .
4. **Verify that**  $D$  **is diagonal**.
5. **Verify that**  $C_k(AB) = DC_k(B)$ .

## Procedure 3.18

### Objective

Choose a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Also choose an  $\mathcal{M}_{n,n}(\mathbb{Q}[x])$ ,  $B$ . Also choose an integer  $0 \leq k \leq \min(m, n)$ . The objective of the following instructions is to construct a  $\mathcal{D}_{\binom{m}{k}, \binom{n}{k}}(\mathbb{Q}[x])$   $D$  such that  $C_k(AB) = DC_k(B)$ .

### Implementation

1. Let  $D = C_k(0_{m \times n}) = 0_{\binom{m}{k} \times \binom{n}{k}}$ .

### Procedure 3.19

#### Objective

Choose an integer  $1 \leq k \leq \min(m, n)$ . Choose a  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $A$ , such that the off diagonal entry is the  $\mathbb{Q}[x]$   $p$  at  $(i, j)$ . Also choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $B$ . The objective of the following instructions is to show that  $C_k(AB) = C_k(A)C_k(B)$ .

#### Implementation

1. Execute **procedure 3.17** on matrices  $A$  and  $I_m$ . Let  $D$  be the matrix constructed.
2. Using **procedure 3.16**, verify that  $C_k(A) = C_k(AI_m) = DC_k(I_m) = DI_{\binom{m}{k}} = D$ .
3. Execute **procedure 3.17** on matrices  $A$  and  $B$ . Let  $D'$  be the matrix constructed.
4. Verify that  $C_k(AB) = D'C_k(B)$ .
5. Verify that  $D' = D = C_k(A)$ .
6. **Therefore verify that  $C_k(AB) = C_k(A)C_k(B)$ .**

Make an analogous procedure to show that  $C_k(BA) = C_k(B)C_k(A)$ .

Using **procedure 3.18**, make a procedure similar to the above but that only instead allows for a diagonal matrix of  $\mathbb{Q}[x]$ s,  $A$ , to be chosen.

### Procedure 3.20

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Let  $D_{0,0} = 1$ . The objective of the following instructions is to construct a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ s,  $M$ , a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ ,  $D$ , a list of  $\mathbb{Q}[x]$ s,  $v$ , and a list of  $\mathcal{T}_n(\mathbb{Q}[x])$ s,  $N$ , such that  $MAN = D$ ,  $A = M^{-1}DN^{-1}$ , and  $D_{i,i} = v_i D_{i-1,i-1}$  for  $i = 1$  to  $i = \min(m, n)$ .

#### Implementation

1. Let  $D$  be a copy of  $A$ .
2. Let  $\langle M, N \rangle$  receive the results of executing **procedure 3.03** on the pair  $\langle m, n \rangle$  and the following procedure:

- (a) Execute **procedure 3.09** on the matrix  $D$  and let  $\langle v \rangle$  receive.

3. **Verify that  $D_{i,i} = v_i D_{i-1,i-1}$  for  $i = 1$  to  $i = \min(m, n)$ .**

4. **Verify that  $M_*AN_* = D$ .**

5. Hence verify that  $A = I_m A I_n = M^{-1}_* M_* A N_* N^{-1}_* = M^{-1}_* D N^{-1}_*$ .

6. **Yield the tuple  $\langle M, D, v, N \rangle$ .**

### Procedure 3.21

#### Objective

Choose integers  $0 \leq k \leq \min(m, n, p)$ . Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Also choose a  $\mathcal{M}_{n,p}(\mathbb{Q}[x])$ ,  $B$ . The objective of the following instructions is to show that  $C_k(AB) = C_k(A)C_k(B)$ .

#### Implementation

1. Execute **procedure 3.20** on  $A$  and let  $\langle M, D, , N \rangle$  receive.

2. Using repeated applications of **procedure 3.19**, verify that  $C_k(AB)$

$$(a) = C_k(M^{-1}_1 \cdots M^{-1}_{|M|} D N^{-1}_1 \cdots N^{-1}_{|N|} B)$$

$$(b) = C_k(M^{-1}_1) \cdots C_k(M^{-1}_{|M|}) * C_k(D) * C_k(N^{-1}_1) \cdots C_k(N^{-1}_{|N|}) C_k(B)$$

$$(c) = C_k(M^{-1}_1 \cdots M^{-1}_{|M|} D N^{-1}_1 \cdots N^{-1}_{|N|}) C_k(B)$$

$$(d) = C_k(A) C_k(B).$$

### Procedure 3.22

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ . Let  $D$  be a copy of  $A$ . Execute **procedure 3.09** on  $D$ . The objective of the following instructions is to show that  $\det(A)$  is the product of the diagonal entries of  $D$ .

#### Implementation

1. Execute **procedure 3.20** on  $A$  and let  $\langle M, D, , N \rangle$  receive.

2. Using **procedure 3.10** and **procedure 3.21**, verify that  $\det(A)$

$$(a) = C_m(A)$$

$$(b) = C_m(M^{-1}_1 \cdots M^{-1}_{|M|} D N^{-1}_1 \cdots N^{-1}_{|N|})$$

$$(c) = C_m(M^{-1}_1) \cdots C_m(M^{-1}_{|M|}) C_m(D) C_m(N^{-1}_1) \cdots C_m(N^{-1}_{|N|})$$

$$(d) = 1 \cdots 1 C_m(D) 1 \cdots 1 = C_m(D)$$

$$(e) = \det(D).$$

3. Using **procedure 3.10**, verify that  $\det(D)$  is the product of the diagonal entries of  $D$ .

### Procedure 3.23

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to construct a  $\mathcal{M}_{n,m}(\mathbb{Q}[x])$ ,  $A^T$ .

#### Implementation

1. Make an  $n \times m$  matrix,  $A^T$ .
2. For  $i = 1$  to  $i = n$ :
  - (a) For  $j = 1$  to  $j = m$ :
    - i. Let  $A^T_{i,j} = A_{j,i}$ .
3. Yield the tuple  $\langle A^T \rangle$ .

#### Notation 3.09

Let us use the notation  $A^T$  for the result yielded by executing **procedure 3.23** on  $A$ .

### Procedure 3.24

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ , and a  $\mathcal{M}_{n,k}(\mathbb{Q}[x])$ ,  $B$ . The objective of the following instructions is to show that  $B^T A^T = (AB)^T$ .

#### Implementation

1. Verify that  $B^T A^T$  and  $(AB)^T$  have dimensions  $k \times m$ .
2. For  $i = 1$  to  $i = k$ :
  - (a) For  $j = 1$  to  $j = m$ :
    - i. Using **procedure 3.23**, verify that
 
$$(B^T A^T)_{i,j} = \sum_{l=0}^n B_{l,i} A_{j,l} = \sum_{l=0}^n A_{j,l} B_{l,i} = (AB)_{j,i} = ((AB)^T)_{i,j}.$$
3. Therefore verify that  $B^T A^T = (AB)^T$ .

### Procedure 3.25

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $A$ . The objective of the following instructions is to show that  $\det(A^T) = \det(A)$ .

#### Implementation

1. Execute **procedure 3.20** on  $A$  and let  $\langle M, D, , N \rangle$  receive.
2. Therefore using procedures **procedure 3.22** and **procedure 3.24**, verify that  $\det(A^T)$ 
  - (a)  $= \det((M^{-1}_1 \cdots M^{-1}_{|M|} D N^{-1}_1 \cdots N^{-1}_{|N|})^T)$
  - (b)  $= \det((N^{-1}_{|N|})^T \cdots (N^{-1}_1)^T D^T (M^{-1}_{|M|})^T \cdots (M^{-1}_1)^T)$
  - (c)  $= \det(D^T)$
  - (d)  $= \det(D)$
  - (e)  $= \det(M^{-1}_1 \cdots M^{-1}_{|M|} D N^{-1}_1 \cdots N^{-1}_{|N|})$
  - (f)  $= \det(A)$ .

### Procedure 3.26

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ , and an integer  $0 \leq k \leq \min(m, n)$ . The objective of the following instructions is to show that  $C_k(A)^T = C_k(A^T)$ .



### Implementation

1. For each row label  $I$  of  $C_k(A^T)$ , do the following:
  - (a) For each column label  $J$  of  $C_k(A^T)$ , do the following:
    - i. Using **procedure 3.25**, verify that  $(C_k(A^T))_{I,J} = \det((A^T)_{I,J}) = \det(A_{J,I}) = (C_k(A))_{J,I}$ .
2. **Therefore verify that  $(C_k(A))^T = (C_k(A^T))$ .**

### Procedure 3.27

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q})$ ,  $A$ , and a  $\mathcal{M}_{m,p}(\mathbb{Q})$ ,  $B$ . Execute **procedure 3.20** on  $A$  and let  $\langle M, D, N \rangle$  receive the result. If the indices of the rows of  $D$  that are entirely zero are also the indices of the rows of  $MB$  that are entirely zero, then the objective of the following instructions is to construct a  $\mathcal{M}_{n,p}(\mathbb{Q})$   $E$  such that  $AE = B$ .

### Implementation

1. Verify that  $A = M^{-1}DN^{-1}$ .
2. Verify that  $M^{-1}$ ,  $D$ , and  $N^{-1}$  are  $\mathcal{M}_{*,*}(\mathbb{Q})$ s.
3. Let  $C$  be an  $n \times p$  matrix with its  $i^{th}$  row given as follows:
  - (a) If  $D_{i,i} \neq 0$ , then do the following:
    - i. Let row  $i$  be row  $i$  of  $MB$  divided by  $D_{i,i}$ .
  - (b) Otherwise, do the following:
    - i. **Choose  $p$  rational numbers to fill up the row.**
4. Verify that  $DC = MB$ .
5. Let  $E$  be  $NC$ .
6. **Therefore using procedure 3.05, verify that  $AE = M^{-1}DN^{-1}E = M^{-1}DN^{-1}NC = M^{-1}DI_nC = M^{-1}DC = M^{-1}MB = I_mB = B$ .**
7. **Yield the tuple  $\langle E \rangle$ .**

### Notation 3.10

The notation  $A \setminus B$  shall be used to refer to the result yielded by executing **procedure 3.27** on matrices  $A$  and  $B$ .

### Notation 3.11

Make an analogous procedure to yield an  $F$  such that  $FA = B$ . The notation  $B/A$  shall be used to refer to the  $F$  yielded by invoking this procedure.

### Procedure 3.28

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q})$ ,  $A$ , a  $\mathcal{M}_{n,p}(\mathbb{Q})$ ,  $E$ , and a  $\mathcal{M}_{m,p}(\mathbb{Q})$ ,  $B$  such that  $AE = B$ . Execute **procedure 3.20** on  $A$  and let  $\langle M, D, N \rangle$  receive the result. If the indices of the rows of  $D$  that are entirely zero are not also the indices of the rows of  $M_*B$  that are entirely zero, then the objective of the following instructions is to show that  $0 \neq 0$ .

### Implementation

1. Verify that  $M^{-1}DN^{-1}E = AE = B$ .
2. Therefore verify that  $DN^{-1}E = M_*B$ .
3. Let  $i$  be an integer such that  $D_{i,*}$  is zero and yet  $(M_*B)_{i,*}$  is not zero.
4. Verify that  $D_{i,*} = D_{i,*}N^{-1}E = (DN^{-1}E)_{i,*} = (M_*B)_{i,*}$ .
5. Let  $j$  be an integer such that  $(M_*B)_{i,j} \neq 0$ .
6. **Now verify that  $0 = D_{i,j} = (M_*B)_{i,j} \neq 0$ .**

### Procedure 3.29

#### Objective

Choose two  $\mathcal{M}_{m,m}(\mathbb{Q})$ s,  $A$  and  $B$ , such that  $AB = I_m$ . The objective of the following instructions is to show that either  $0 = 1$  or  $BA = I_m$ .

## Implementation

1. Execute **procedure 3.20** on  $B$  and let  $\langle M, D, N \rangle$  receive the result.
2. Verify that  $B = M^{-1} * D N^{-1} *$ .
3. If  $D$  has a zero on its diagonal, then do the following:
  - (a) Using **procedure 3.22**, verify that  $\det(I_m) = \det(AB) = \det(A) \det(B) = \det(A) \det(D) = \det(A) * 0 = 0$ .
  - (b) Using **procedure 3.10**, verify that  $\det(I_m) = 1^m = 1$ .
  - (c) Therefore verify that  $0 = 1$ .
  - (d) **Abort procedure.**
4. Otherwise do the following:
  - (a) Verify that  $D$  does not have a zero on its diagonal.
  - (b) Verify that  $B \setminus I_m = I_m(B \setminus I_m) = AB(B \setminus I_m) = A(B(B \setminus I_m)) = AI_m = A$ .
  - (c) **Therefore verify that**  $BA = B(B \setminus I_m) = I_m$ .

## Procedure 3.30

### Objective

Choose an  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $M$ , and an  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $B$ . The objective of the following instructions is to construct a  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ ,  $Q$ , and a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $R$ , such that  $M = (xI_m - B)Q + R$ .

### Implementation

1. Let  $M_0x^b + M_1x^{b-1} + \dots + M_bx^0 = M$ , where the  $M_i$  are  $\mathcal{M}_{m,m}(\mathbb{Q})$ s.
2. Now let  $R = B^bM_0 + B^{b-1}M_1 + \dots + B^0M_b$ .
3. Let  $Q = \sum_{k=1}^b (x^{k-1}I_mB^0 + x^{k-2}I_mB^1 + \dots + x^0I_mB^{k-1})M_k$ .
4. Verify that  $M - R = (xI_m - B) \sum_{k=1}^b (x^{k-1}I_mB^0 + x^{k-2}I_mB^1 + \dots + x^0I_mB^{k-1})M_k = (xI_m - B)Q$ .
5. **Verify that**  $M = (xI_m - B)Q + R$ .

6. Yield the tuple  $\langle Q, R \rangle$ .

**Make an analogous procedure that instead has the objective of constructing a  $Q$  and  $R$  such that  $M = Q(xI_m - B) + R$ .**

## Procedure 3.31

### Objective

Choose two  $\mathcal{M}_{m,m}(\mathbb{Q})$ s,  $B, A$ , and two lists of  $\mathcal{T}_m(\mathbb{Q}[x])$ s such that  $xI_m - B = M(xI_m - A)N$ . The objective of the following instructions is to either show that  $0 = 1$  or to construct  $\mathcal{M}_{m,m}(\mathbb{Q})$ s  $R_1$  and  $R_3$  such that  $I_m = R_1R_3$  and  $B = R_1AR_3$ .

### Implementation

1. Verify that  $(xI_m - B)N^{-1} = M(xI_m - A)NN^{-1} = M(xI_m - A)I_m = M(xI_m - A)$ .
2. Execute **procedure 3.30** on  $\langle M, B \rangle$  and let  $\langle Q_1, R_1 \rangle$  receive.
3. Verify that  $M = (xI_m - B)Q_1 + R_1$ .
4. Execute **procedure 3.30** on  $\langle N^{-1}, A \rangle$  and let  $\langle Q_2, R_2 \rangle$  receive.
5. Verify that  $N^{-1} = Q_2(xI_m - A) + R_2$ .
6. By substituting  $M$  and  $N^{-1}$  into (2), verify that  $(xI_m - B)(Q_2(xI_m - A) + R_2) = ((xI_m - B)Q_1 + R_1)(xI_m - A)$ .
7. By rearranging both sides, verify that  $(xI_m - B)(Q_2 - Q_1)(xI_m - A) = R_1(xI_m - A) - (xI_m - B)R_2$ .
8. By equating the coefficients of different powers of  $x$  both sides, verify that  $Q_2 - Q_1 = 0_{m \times m}$ .
9. Verify that  $R_1(xI_m - A) - (xI_m - B)R_2 = (xI_m - B)(Q_2 - Q_1)(xI_m - A) = (xI_m - B)0_{m \times m}(xI_m - A) = 0_{m \times m}$ .
10. Therefore by adding  $(xI_m - B)R_2$  to both sides, verify that  $xR_1 - R_1A = R_1(xI_m - A) = (xI_m - B)R_2 = xR_2 - BR_2$ .
11. By equating the coefficients of  $x$  on both sides, verify that  $R_1 = R_2$ .
12. Therefore verify that  $R_1A = BR_1$ .

13. Execute **procedure 3.30** on  $\langle M^{-1}, A \rangle$  and let  $\langle Q_3, R_3 \rangle$  receive.
14. Verify that  $M^{-1} = (xI_m - A)Q_3 + R_3$ .
15. Verify that  $I_m = MM^{-1} = ((xI_m - B)Q_1 + R_1)M^{-1} = (xI_m - B)Q_1M^{-1} + R_1M^{-1} = (xI_m - B)Q_1M^{-1} + R_1(xI - A)Q_3 + R_1R_3 = (xI_m - B)Q_1M^{-1} + (xI - B)R_1Q_3 + R_1R_3 = (xI_m - B)(Q_1M^{-1} + R_1Q_3) + R_1R_3$ .
16. By equating the powers of  $x$  on both sides, verify that  $Q_1M^{-1} + R_1Q_3 = 0$ .
17. By substituting zero for  $Q_1M^{-1} + R_1Q_3$ , **verify that**  $I_m = (xI_m - B)0_{m \times m} + R_1R_3 = R_1R_3$ .
18. **Therefore using procedure 3.29, verify that**  $R_3R_1 = I_m$ .
19. **Also, verify that**  $B = BI_m = BR_1R_3 = R_1AR_3$ .
20. Yield the pair  $(R_1, R_3)$ .

### Procedure 3.32

#### Objective

Choose a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Choose two integers  $1 \leq i, j \leq m$  such that  $i \neq j$ . The objective of the following instructions is to negate row  $i$  and swap it with row  $j$  using only elementary row and column operations.

#### Implementation

1. Let  $A$  be our working matrix.
2. Subtract row  $j$  from row  $i$ .
3. Add row  $i$  to row  $j$ .
4. Subtract row  $j$  from row  $i$ .
5. **Verify that the  $i^{th}$  row has been negated and swapped with the  $j^{th}$  row.**

Make an analogous procedure to negate column  $i$  and swap it with column  $j$ .

### Procedure 3.33

#### Objective

Choose a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Choose two integers  $1 \leq i, j \leq \min(m, n)$  such that  $i \neq j$ . The objective of the following instructions is to swap  $B_{i,i}$  and  $B_{j,j}$  using only elementary row and column operations.

#### Implementation

1. Let  $A$  be our working matrix.
2. Use **procedure 3.32** to negate the  $i^{th}$  row and swap it with the  $j^{th}$  row.
3. Use **procedure 3.32** to negate the  $i^{th}$  column and swap it with the  $j^{th}$  column.
4. **Therefore, overall verify that  $B_{i,i}$  and  $B_{j,j}$  have been swapped.**

### Procedure 3.34

#### Objective

Choose a  $\mathcal{D}_{m,n}(\mathbb{Q}[x])$ ,  $A$ . Choose two integers  $1 \leq i, j \leq \min(m, n)$  such that  $i \neq j$ . Choose a rational  $k \neq 0$ . The objective of the following instructions is to multiply  $B_{i,i}$  by  $k$  and  $B_{j,j}$  by  $\frac{1}{k}$  using only elementary row and column operations.

#### Implementation

1. Let  $A$  be our working matrix.
2. Add  $k$  times row  $i$  to row  $j$ .
3. Subtract  $\frac{1}{k}$  times row  $j$  from row  $i$ .
4. Add  $k$  times row  $i$  to row  $j$ .
5. Verify that the  $i^{th}$  row has been scaled by  $k$ , the  $j^{th}$  row by  $-\frac{1}{k}$ , and that both these rows are swapped.
6. Use **procedure 3.32** to negate the  $i^{th}$  row and swap it with the  $j^{th}$  row.
7. **Therefore, overall verify that  $B_{i,i}$  has been multiplied by  $k$ , and  $B_{j,j}$  by  $\frac{1}{k}$ .**

### Notation 3.12

Let us use the notation " $p$  is monic" as a shorthand for " $x^{\deg(p)} \circ p = 1$ ".

### Procedure 3.35

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute [procedure 3.09](#) on the polynomial matrix  $xI - A$  and let  $\langle B \rangle$  be the result. The objective of the following instructions is to show that either none of the diagonal entries of  $B$  are equal to zero, or  $1 = 0$ .

#### Implementation

1. Using [procedure 3.10](#), verify that  $\det(xI - A)$  is a monic polynomial of degree  $m$ .
2. Therefore using [procedure 3.22](#), verify that  $\det(B) = \det(xI - A)$ .
3. Therefore verify that  $\det(B)$  is a monic polynomial of degree  $m$ .
4. If any of the diagonal entries of  $B$  equal zero, then do the following:
  - (a) Using [procedure 3.10](#), verify that  $\det(B) = B_{1,1}B_{2,2} \cdots B_{m,m} = 0$ .
  - (b) Therefore using (3) and (4a), verify that  $1 = 0$ .
  - (c) **Abort procedure.**
5. Otherwise do the following:
  - (a) **Verify that none of the diagonal entries of  $B$  equal zero.**

### Notation 3.13

Let us use the notation  $\text{cols}(A)$  as a shorthand for "the number of columns of  $A$ ".

### Notation 3.14

Let us use the notation  $\text{rows}(A)$  as a shorthand for "the number of rows of  $A$ ".

### Procedure 3.36

#### Objective

Choose a list of  $\mathcal{M}_*(\mathbb{Q})$ ,  $C$ . Let  $m = \sum_{i=1}^{|C|} \text{cols}(C_i)$ . The objective of the following instructions is to construct a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $\text{bdiag}(C)$ .

#### Implementation

1. Let  $E$  be a  $0 \times 0$  matrices.
2. **Now for**  $i = 1$  **to**  $i = |C|$ :
  - (a) Add  $\text{cols}(C_i)$  columns filled with zeros to the right end of  $E$ .
  - (b) Add  $\text{cols}(C_i)$  rows filled with zeros to the bottom end of  $E$ .
  - (c) Set the bottom-right corner of  $E$  equal to  $C_i$ .
3. Verify that  $\text{cols}(E) = \sum_{i=1}^{|C|} \text{cols}(C_i) = m$ .
4. **Yield the tuple**  $\langle E \rangle$ .

### Notation 3.15

Let us use the notation  $\text{bdiag}(C)$  as a shorthand for the result yielded by executing [procedure 3.36](#) on  $C$ .

### Procedure 3.37

#### Objective

Choose a positive integer  $m$  and an  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute [procedure 3.20](#) on the polynomial matrix  $xI_m - A$  and let  $\langle B, v, \rangle$  be the result. The objective of the following instructions is to either show that  $0 < 0$  or to construct an integer  $a$  such that  $\sum_{i=a}^m \deg(B_{i,i}) = m$ ,  $\deg(B_{i,i}) > 0$  for  $i = a$  to  $i = m$ , and  $\deg(B_{i,i}) = 0$  for  $i = 1$  to  $i = a - 1$ .

#### Implementation

1. Execute [procedure 3.35](#) on  $A$ .
2. If  $\deg(B_{i,i}) = 0$  for  $i = 1$  to  $i = m$ , then do the following:
  - (a) Verify that  $\det(xI_m - A) = \det(B) = B_{1,1}B_{2,2} \cdots B_{m,m}$ .

(b) Therefore verify that  $0 < m = \deg(\det(xI_m - A)) = \deg(B_{1,1}B_{2,2} \cdots B_{m,m}) = 0 + 0 + \cdots + 0 = 0$ .

(c) **Abort procedure.**

3. Otherwise do the following:

(a) Let  $1 \leq a \leq m$  be the least integer such that  $\deg(B_{a,a}) > 0$ .

(b) **Verify that**  $\deg(B_{i,i}) = 0$  **for**  $i = 1$  **to**  $i = a - 1$ .

(c) **Verify that**  $\sum_{i=a}^m \deg(B_{i,i}) = \sum_{i=1}^m \deg(B_{i,i}) = \deg(B_{1,1}B_{2,2} \cdots B_{m,m}) = \deg(\det(B)) = \deg(xI_m - A) = m$ .

(d) For  $i = a + 1$  to  $i = m$ , do the following:

i. Verify that  $B_{i,i} = u_i B_{i-1,i-1}$ .

ii. Verify that  $B_{i,i} \neq 0$ .

iii. Therefore verify that  $u_i \neq 0$ .

iv. **Therefore verify that**  $\deg(B_{i,i}) = \deg(u_i B_{i-1,i-1}) \geq \deg(B_{i-1,i-1}) > 0$ .

(e) **Yield the tuple**  $\langle a \rangle$ .

### Procedure 3.38

#### Objective

Choose a  $\mathbb{Q}[x]$ ,  $p = x^k + p_1x^{k-1} + p_2x^{k-2} + \cdots + p_kx^0$  such that  $k > 0$ . The objective of the following instructions is to construct a  $\mathcal{M}_{k,k}(\mathbb{Q})$ ,  $\text{rcan}(p)$ .

#### Implementation

1. Make a  $k \times k$  matrix  $C$ .
2. Let  $C$ 's first  $k - 1$  columns be filled with the last  $k - 1$  columns of  $I_k$ .
3. Let  $C$ 's last column from top to bottom be  $-p_k, -p_{k-1}, \dots, -p_1$ .
4. **Yield the tuple**  $\langle C \rangle$ .

#### Notation 3.16

Let us use  $\text{rcan}(p)$  as a shorthand for the result yielded by executing [procedure 3.38](#) on  $p$ .

### Procedure 3.39

#### Objective

Choose a monic  $\mathbb{Q}[x]$ ,  $p$  such that  $\deg(p) > 0$ . Let  $k = \deg(p)$ . Choose a  $\mathcal{M}_{k,k}(\mathbb{Q}[x])$ ,  $D$ , such that  $D = xI_k - \text{rcan}(p)$ . The objective of the following instructions is to transform  $D$  into  $\text{bdiag}(1, \dots, 1, p)$  by a sequence of elementary operations.

#### Implementation

1. Let the matrix  $D$  be our working matrix.
2. For  $i = k$  going down to  $i = 2$ , add  $x$  times row  $i$  to row  $i - 1$ .
3. Verify that  $D$ 's first  $k - 1$  columns are now the last  $k - 1$  columns of  $-I_k$ .
4. Verify that  $D$ 's last column is  $p$  followed by some other polynomials.
5. For  $i = 2$  going up to  $i = k$ , subtract  $D_{i,k}$  times column  $i - 1$  from column  $k$ .
6. Verify that  $D$ 's last column is now  $p$  followed by zeros.
7. For  $i = 2$  going up to  $i = k$ , negate row  $i - 1$  and exchange it with row  $i$  using [procedure 3.32](#).
8. **Therefore verify that**  $D = \text{bdiag}(1, \dots, 1, p)$ .

#### Notation 3.17

Let us use the notation  $\text{mon}(p)$  as a shorthand for  $\frac{p}{x^{\deg(p)} \circ p}$ .

### Procedure 3.40

#### Objective

Choose a positive integer  $m$  and an  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute [procedure 3.03](#) on the polynomial matrix  $xI_m - A$  and let  $\langle B, \rangle$  receive the result. Execute [procedure 3.37](#) on  $A$  and let  $\langle a \rangle$  receive the result. Let  $E_i = \text{rcan}(\text{mon}(B_{a-1+i, a-1+i}))$  for  $i = 1$  to  $i = m + 1 - a$ . The objective of the following instructions is to first show that  $\text{cols}(\text{bdiag}(E)) = m$ ,

and second to apply a sequence of elementary operations on  $xI_m - \text{bdiag}(E)$  to obtain the matrix  $B$ .

## Implementation

1. Verify that the diagonal of  $B$  comprises  $x - 1$  rationals followed by  $B_{a,a}, B_{a+1,a+1}, \dots, B_{m,m}$ .
2. Using **procedure 3.39**, verify that
$$\begin{aligned} \text{cols}(\text{bdiag}(E)) &= \sum_{i=1}^{|E|} \text{cols}(E_i) = \\ \sum_{i=1}^{|E|} \text{cols}(\text{rcan}(\text{mon}(B_{a-1+i,a-1+i}))) &= \\ \sum_{i=1}^{|E|} \text{deg}(\text{mon}(B_{a-1+i,a-1+i})) &= \\ \sum_{i=1}^{m+1-a} \text{deg}(B_{a-1+i,a-1+i}) &= \\ \sum_{i=a}^m \text{deg}(B_{i,i}) = m. \end{aligned}$$
3. Let  $F = xI_m - \text{bdiag}(E)$ .
4. Now for  $i = 1$  to  $i = |E|$ :
  - (a) Let  $j = 1 + \sum_{r=1}^{i-1} \text{cols}(E_r)$ .
  - (b) Let  $k = j + \text{cols}(E_i)$ .
  - (c) Apply **procedure 3.39** on the tuple  $\langle \text{mon}(B_{a-1+i,a-1+i}), F_{[j:k],[j:k]} \rangle$ .
5. Now verify that  $F$  is a  $\mathcal{D}_{m,m}(\mathbb{Q})$ .
6. Also verify that the diagonal of  $F$  comprises  $\text{mon}(B_{a,a}), \text{mon}(B_{a+1,a+1}), \dots, \text{mon}(B_{m,m})$  and  $a - 1$  1s.
7. Rearrange the diagonal of  $F$  so that  $\text{mon}(B_{i,i})$  is at the  $i^{\text{th}}$  position on the diagonal for  $i = a$  to  $i = m$  by doing pairwise swaps. In general, swap the  $i^{\text{th}}$  and  $j^{\text{th}}$  diagonal entries using **procedure 3.33**.
8. For  $i = 1$  to  $i = m - 1$ , do the following:
  - (a) Let  $k = \frac{x^{\text{deg}(B_{i,i})} \circ B_{i,i}}{x^{\text{deg}(F_{i,i})} \circ F_{i,i}}$ .
  - (b) Scale  $B_{i,i}$  by  $k$  and  $B_{i+1,i+1}$  by  $\frac{1}{k}$  using **procedure 3.34**.
  - (c) Now verify that  $F_{i,i} = B_{i,i}$ .
9. Now verify that  $x^m \circ \det(F) = x^m \circ \det(xI_m - \text{bdiag}(E)) = 1 = x^m \circ \det(xI_m - A) = x^m \circ \det(B)$ .
10. Therefore verify that
$$\begin{aligned} F_{m,m} &= \frac{x^{\text{deg}(F_{m,m})} \circ \det(F)}{x^{m-\text{deg}(F_{m,m})} \circ (\det(F_{[1:m],[1:m]}))} = \\ \frac{x^{\text{deg}(B_{m,m})} \circ \det(B)}{x^{m-\text{deg}(B_{m,m})} \circ (\det(B_{[1:m],[1:m]}))} &= x^{\text{deg}(B_{m,m})} \circ B_{m,m}. \end{aligned}$$

11. Therefore verify that  $F_{m,m} = B_{m,m}$ .

12. **Therefore verify that  $F = B$ .**

## Procedure 3.41

### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute **procedure 3.37** on  $A$  and let  $\langle a \rangle$  receive the result. Let  $E_i = \text{rcan}(\text{mon}(B_{a-1+i,a-1+i}))$  for  $i = 1$  to  $i = m + 1 - a$ . The objective of the following instructions is to either show that  $0 = 1$  or to construct  $\mathcal{M}_{m,m}(\mathbb{Q})$ s  $R, T$  such that  $A = R \text{bdiag}(E)T$ ,  $RT = I_m$ , and  $TR = I_m$ .

### Implementation

1. Execute **procedure 3.20** on the polynomial matrix  $xI_m - A$  and let  $\langle P, B, , Q \rangle$  be the result.
2. Verify that  $P_*(xI_m - A)Q_* = B$ .
3. Verify that  $xI_m - A = P^{-1} *_* BQ^{-1} *_*$ .
4. Let  $Z$  be a variant of **procedure 3.20** where every occurrence of **procedure 3.09** in its instructions is replaced with **procedure 3.40**, and where every mention of  $v$  is ignored.
5. Execute procedure  $Z$  on the matrix  $xI_m - \text{bdiag}(E)$  and let  $\langle M, , , N \rangle$  receive the result.
6. Verify that  $M_*(xI_m - \text{bdiag}(E))N_* = B$ .
7. Verify that  $xI_m - A = P^{-1} *_* BQ^{-1} *_* = P^{-1} *_* M(xI_m - \text{bdiag}(E))NQ^{-1} *_*$ .
8. Execute **procedure 3.31** on the matrices  $\langle A, P^{-1}M, \text{bdiag}(E), NQ^{-1} \rangle$ . Let the tuple  $\langle R, T \rangle$  be the result.
9. **Verify that  $A = R \text{bdiag}(E)T$ .**
10. **Verify that  $RT = I_m$ .**
11. **Verify that  $TR = I_m$ .**
12. **Yield the tuple  $\langle R, E, T \rangle$ .**

### Procedure 3.42

#### Objective

Choose a  $\mathbb{Q}[x]$ ,  $r = r_0x^t + r_1x^{t-1} + \dots + r_tx^0$ , and  $\mathcal{M}_{m,m}(\mathbb{Q})$ s,  $R, A, S$  such that  $SR = I_m$ . The objective of the following instructions is to show that  $r(RAS) = Rr(A)S$ .

#### Implementation

1. **Verify that**  $r(RAS) = r_0(RAS)^t + r_1(RAS)^{t-1} + \dots + r_t(RAS)^0 = r_0RA^tS + r_1RA^{t-1}S + \dots + r_tRA^0S = R(r_0A^t + r_1A^{t-1} + \dots + r_tA^0)S = Rr(A)S$ .

### Procedure 3.43

#### Objective

Choose a list of  $\mathcal{M}_{m,m}(\mathbb{Q})$ s,  $A$ , and a  $\mathbb{Q}[x]$ ,  $r = r_0x^t + r_1x^{t-1} + \dots + r_tx^0$ . The objective of the following instructions is to show that  $r(\text{bdiag}(A)) = \text{bdiag}(r(A))$ .

#### Implementation

1. For  $i = 0$  up to  $i = t$ , by repeated applications of [procedure 3.08](#), verify that  $\text{bdiag}(A)^i$  evaluates to  $\text{bdiag}(A^i)$  (where the exponentiation is done element-wise).
2. Therefore verify that  $r(\text{bdiag}(A))$ 
  - (a)  $= r_0 \text{bdiag}(A)^t + r_1 \text{bdiag}(A)^{t-1} + \dots + r_t \text{bdiag}(A)^0$
  - (b)  $= r_0 \text{bdiag}(A^t) + r_1 \text{bdiag}(A^{t-1}) + \dots + r_t \text{bdiag}(A^0)$
  - (c)  $= \text{bdiag}(r_0A^t) + \text{bdiag}(r_1A^{t-1}) + \dots + \text{bdiag}(r_tA^0)$
  - (d)  $= \text{bdiag}(r(A))$  (**where  $r$  is applied element-wise**).

### Procedure 3.44

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ , and a  $\mathbb{Q}[x]$ ,  $r$ . Execute [procedure 3.41](#) on the matrix  $A$  and let the tuple  $\langle R_1, E, R_3 \rangle$  receive the result. The objective of the following instructions is to show that  $r(A) = R_1 \text{bdiag}(r(E))R_3$  (where  $r$  is applied element-wise).

#### Implementation

1. Verify that  $R_3R_1 = I_m$ .
2. Using [procedure 3.42](#), verify that  $r(A) = r(R_1 \text{bdiag}(E)R_3) = R_1r(\text{bdiag}(E))R_3$ .
3. Using [procedure 3.43](#), verify that  $r(\text{bdiag}(E)) = \text{bdiag}(r(E))$  (where  $r$  is applied element-wise).
4. **Therefore verify that**  $r(A) = R_1 \text{bdiag}(r(E))R_3$  (**where  $r$  is applied element-wise**).

### Notation 3.18

Let us use the notation  $e_i$  as a shorthand for "the  $\mathcal{M}_{k,1}(\mathbb{Q})$  that is 0, except for its  $i^{\text{th}}$  entry which is 1".

### Notation 3.19

Let us use the notation  $0_{m \times n}$  as a shorthand for "the  $\mathcal{M}_{m,m}(\mathbb{Q})$  such that every entry is 0".

### Procedure 3.45

#### Objective

Choose a  $\mathbb{Q}[x]$   $p = x^k + p_1x^{k-1} + p_2x^{k-2} + \dots + p_kx^0$  such that  $k > 0$ . The objective of the following instructions is to show that  $p(\text{rcan}(p)) = 0_{k \times k}$ .

#### Implementation

1. Let  $G = \text{rcan}(p)$ .
2. Then by  $G$ 's construction, for  $i = 1$  up to  $i = k$ , verify that  $G^{i-1}e_1 = G^{i-2}e_2 = \dots = G^0e_i = e_i$ .

3. Therefore, for  $i = 1$  up to  $i = k$ : Cognizant of the construction of  $G$ 's last column, verify that
  - (a)  $= (G^k + p_1 G^{k-1} + p_2 G^{k-2} + \dots + p_k G^0) e_i$
  - (b)  $= (G^k + p_1 G^{k-1} + p_2 G^{k-2} + \dots + p_k G^0) G^{i-1} e_1$
  - (c)  $= G^{i-1} (G G^{k-1} + p_1 G^{k-1} + p_2 G^{k-2} + \dots + p_k G^0) e_1$
  - (d)  $= G^{i-1} (G e_k + p_1 e_k + p_2 e_{k-1} + \dots + p_k e_1)$
  - (e)  $= G^{i-1} 0_{k \times 1}$
  - (f)  $= 0_{k \times 1}$ .
4. Therefore verify that  $p(G) = 0_{k \times k}$ .

### Procedure 3.46

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to define the  $\mathbb{Q}[x]$   $\text{last}_A$  and show that either  $1 = 0$  or  $\text{last}_A \neq 0$ .

#### Implementation

1. Execute [procedure 3.20](#) on the polynomial matrix  $xI_m - A$  and let the tuple  $\langle B, \rangle$  receive the result.
2. Execute [procedure 3.35](#) on  $A$ .
3. Verify that  $B_{m,m} \neq 0$ .
4. Yield  $\langle B_{m,m} \rangle$ .

#### Notation 3.20

Let us use the notation  $\text{last}_A$  as a shorthand for the result of executing [procedure 3.46](#) on  $A$ .

### Procedure 3.47

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to either show that  $0 < 0$  or to show that  $\text{last}_A(A) = 0_{m \times m}$ .

#### Implementation

1. Execute [procedure 3.20](#) on the matrix  $A$  and let the tuple  $\langle M, B, v, N \rangle$  receive the result.
2. Execute [procedure 3.37](#) on  $A$  and let  $\langle a \rangle$  receive.
3. Execute [procedure 3.41](#) on  $A$  and let  $\langle R, E, T \rangle$  receive.
4. For  $j = 1$  to  $j = |E|$ :
  - (a) Verify that  $E_j = \text{rcan}(\text{mon}(B_{a-1+j, a-1+j}))$ .
  - (b) Verify that  $\text{last}_A = B_{m,m} = B_{a-1+j, a-1+j} v_{a+j} v_{a+j+1} \dots v_m$ .
  - (c) Let  $k = \deg(\text{mon}(B_{a-1+j, a-1+j}))$ .
  - (d) Therefore using [procedure 3.45](#) verify that  $\text{last}_A(E_j) = B_{m,m}(E_j) = B_{a-1+j, a-1+j}(\text{rcan}(\text{mon}(B_{a-1+j, a-1+j}))) \cdot v_{a+j}(E_j) v_{a+j+1}(E_j) \dots v_m(E_j) = 0_{k \times k} v_{a+j}(E_j) v_{a+j+1}(E_j) \dots v_m(E_j) = 0_{k \times k}$ .
5. Therefore using [procedure 3.44](#) verify that  $\text{last}_A(A) = R \text{bdiag}(\text{last}_A(E))T = R \text{bdiag}(B_{m,m}(E))T = R 0_{m \times m} T = 0_{m \times m}$ .

### Procedure 3.48

#### Objective

Choose a monic  $\mathbb{Q}[x]$   $p$  such that  $\deg(p) > 0$ . Choose a  $\mathbb{Q}[x]$   $g = g_0 x^k + g_1 x^{k-1} + \dots + g_k x^0$  such that  $g_0 \neq 0$  and  $k < \deg(p)$ . The objective of the following instructions is to show that  $g(\text{rcan}(p)) \neq 0_{\deg(p) \times \deg(p)}$ .

#### Implementation

1. Let  $G = \text{rcan}(p)$ .
2. Therefore cognizant of  $G$ 's construction, verify that  $g(G) e_1 = (g_0 G^k + g_1 G^{k-1} + \dots + g_k G^0) e_1 = g_0 e_{k+1} + g_1 e_k + \dots + g_k e_1 \neq 0_{\deg(p) \times 1}$ .
3. Therefore verify that  $g(G) \neq 0_{\deg(p) \times \deg(p)}$ .



### Procedure 3.49

#### Objective

Choose two  $\mathbb{Q}[x]$ s  $g = g_0x^k + g_1x^{k-1} + \dots + g_kx^0$ ,  $p = x^k + p_1x^{k-1} + p_2x^{k-2} + \dots + p_kx^0$  such that  $\deg(p) = \deg(g) > 0$  and  $g(\text{rcan}(p)) = 0_{\deg(p) \times \deg(p)}$ . The objective of the following instructions is to show that  $g = g_0p$ .

#### Implementation

1. Let  $G = \text{rcan}(p)$ .
2. Let  $u = \deg(g)$ .
3. Cognizant of  $G$ 's construction, verify that  $0_{u \times 1} = g(G)e_1 = (g_0G^u + g_1G^{u-1} + g_2G^{u-2} + \dots + g_uG^0)e_1 = g_0Ge_u + g_1e_u + g_2e_{u-1} + \dots + g_ue_1$ .
4. Therefore for  $i = 1$  to  $i = u$ , do the following:
  - (a) Verify that  $0 = (g_0Ge_u + g_1e_u + g_2e_{u-1} + \dots + g_ue_1)_{i,1}$ .
  - (b) Therefore cognizant of  $G$ 's construction, verify that  $-g_0p_{u+1-i} + g_{u+1-i} = 0$ .
  - (c) Therefore verify that  $g_{u+1-i} = g_0p_{u+1-i}$ .
5. **Therefore verify that**  $g = g_0p$ .

### Procedure 3.50

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Choose a  $\mathbb{Q}[x]$   $p = p_0x^t + p_1x^{t-1} + p_2x^{t-2} + \dots + p_tx^0$  where  $p_0 \neq 0$ , such that  $p(A) = 0_{m \times m}$ . The objective of the following instructions is to either show that  $0 \neq 0$  or to construct a  $\mathbb{Q}[x]$   $f$  such that  $p = f \text{last}_A$ .

#### Implementation

1. Let  $F$  be a  $1 \times 2$  matrix consisting in-order of  $p$  and  $\text{last}_A$ .
2. Execute **procedure 3.20** on  $F$  and let  $\langle M, D, , N \rangle$  receive the result.
3. Verify that  $D_{1,1} \neq 0$ .

4. Let  $g = g_0x^w + g_1x^{w-1} + g_2x^{w-2} + \dots + g_wx^0 = D_{1,1}$  in such a way that  $g_0 \neq 0$ .
5. Verify that  $F = M^{-1}DN^{-1} = DN^{-1}$ .
6. Verify that  $\text{last}_A = F_{1,2} = D_{1,1}N^{-1}_{1,2} + D_{1,2}N^{-1}_{2,2} = D_{1,1}N^{-1}_{1,2} = gN^{-1}_{1,2}$ .
7. Let  $u = \text{last}_A$ .
8. Therefore verify that  $N^{-1}_{1,2} \neq 0$ .
9. Therefore verify that  $u = \deg(\text{last}_A) = \deg(D_{1,1}N^{-1}_{1,2}) \geq \deg(D_{1,1}) = \deg(g) = w$ .
10. Verify that  $D = MFN = FN$ .
11. Therefore verify that  $g = D_{1,1} = N_{1,1}p + N_{2,1}\text{last}_A$ .
12. Therefore using **procedure 3.45**, verify that  $g(A) = N_{1,1}(A)p(A) + N_{2,1}(A)\text{last}_A(A) = N_{1,1}(A)0_{m \times m} + N_{2,1}(A)0_{m \times m} = 0_{m \times m}$ .
13. Execute **procedure 3.41** on the matrix  $A$  and let the tuple  $\langle R_1, E, R_3 \rangle$  receive the result.
14. Using **procedure 3.44**, and **procedure 3.41**, verify that  $\text{bdiag}(g(E)) = I_m \text{bdiag}(g(E))I_m = R_3R_1 \text{bdiag}(g(E))R_3R_1 = R_3g(A)R_1 = R_30_{m \times m}R_1 = 0_{m \times m}$ .
15. Let  $G = \text{rcan}(\text{mon}(\text{last}_A))$ .
16. Verify that  $g(G) = g(E_{|E|}) = \text{bdiag}(g(E))_{[m-u+1:m+1], [m-u+1:m+1]} = 0_{u \times u}$ .
17. If  $w < u$ , then:
  - (a) Using **procedure 3.48**, verify that  $g(G) \neq 0_{u \times u}$ .
  - (b) **Abort procedure.**
18. Otherwise, do the following:
  - (a) Verify that  $w = u$ .
  - (b) Using **procedure 3.49**, verify that  $g = g_0\text{last}_A$ .
  - (c) **Therefore verify that**  $p = F_{1,1} = D_{1,1}N^{-1}_{1,1} + D_{1,2}N^{-1}_{2,1} = N^{-1}_{1,1}g + N^{-1}_{2,1} * 0 = N^{-1}_{1,1}g = N^{-1}_{1,1}g_0\text{last}_A$ .
  - (d) **Yield the tuple**  $\langle N^{-1}_{1,1}g_0 \rangle$ .

### Procedure 3.51

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to construct a  $\mathcal{M}_{m^2,*}(\mathbb{Q})$ ,  $\text{pows}(A)$ .

#### Implementation

1. Let  $t = \deg(\text{last}_A)$ .
2. Make an  $m^2 \times t$  matrix,  $\text{pows}(A)$ , whose  $i^{\text{th}}$  column is the sequential concatenation of the columns of  $A^{t-i}$ .
3. Yield  $\langle \text{pows}(A) \rangle$ .

#### Notation 3.21

Let us use the notation  $\text{pows}(A)$  as a shorthand for the result yielded by executing **procedure 3.51** on  $A$ .

### Procedure 3.52

#### Objective

Choose an  $\mathcal{M}_{m,n}(\mathbb{Q})$ ,  $A$ , and an  $\mathcal{M}_{n,m}(\mathbb{Q})$ ,  $B$ , such that  $AB = I_m$ . The objective of the following instructions is to show that either  $0 = 1$  or every column of  $B$  is non-zero.

#### Implementation

1. If any column  $i$  of  $B$ ,  $Be_i$ , is equal to zero, then:
  - (a) Verify that  $0_{n \times 1} = A0_{n \times 1} = A(Be_i) = (AB)e_i = I_me_i = e_i$ .
  - (b) Therefore verify that  $0=1$ .
  - (c) **Abort procedure.**

### Procedure 3.53

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Choose a  $\mathbb{Q}[x]$   $p$  such that  $p \neq 0$ ,  $p(A) = 0$ , and  $\deg(p) < \deg(\text{last}_A)$ . The

objective of the following instructions is to show that  $0 < 0$ .

#### Implementation

1. Execute **procedure 3.50** on  $A$  and  $p$  and let  $f$  receive.
2. Now verify that  $p = f \text{last}_A$ .
3. Verify that  $f \neq 0$  and  $\text{last}_A \neq 0$ .
4. **Therefore using (O), (2), and (3), verify that  $\deg(\text{last}_A) > \deg(p) = \deg(f \text{last}_A) \geq \deg(\text{last}_A)$ .**
5. **Abort procedure.**

### Procedure 3.54

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute **procedure 3.20** on  $\text{pows}(A)$  and let the tuple  $\langle M, D, , N \rangle$  receive the result. Let  $t = \text{cols}(\text{pows}(A))$ . The objective of the following instructions is to show that either  $0 < 0$  or to show that  $C_t(D) = C_t(D)_{1,1}e_1 \neq 0$ .

#### Implementation

1. Execute **procedure 3.20** on  $\text{pows}(A)$  and let the tuple  $\langle M, D, , N \rangle$  receive the result.
2. Verify that  $M_* \text{pows}(A) N_* = D$ .
3. Using **procedure 3.05**, verify that  $M^{-1}MFN = I_{m^2}FN = FN = M^{-1}D$ .
4. If  $C_t(D)_{1,1} = 0$ , then:
  - (a) Verify that for some  $1 \leq i \leq t$ ,  $D_{i,i} = 0$ .
  - (b) Therefore verify that  $De_i = 0_{m^2 \times 1}$ .
  - (c) Therefore verify that  $F(Ne_i) = (FN)e_i = (M^{-1}D)e_i = M^{-1}(De_i) = 0_{m^2 \times 1}$ .
  - (d) Let  $p = N_{1,i}x^{t-1} + N_{2,i}x^{t-2} + \dots + N_{t,i}x^0$ .
  - (e) Therefore verify that  $p(A) = 0_{m \times m}$ .
  - (f) Execute **procedure 3.52** on  $N^{-1}_*$  and  $N_*$ .
  - (g) Therefore verify that  $p \neq 0$ .
  - (h) Execute **procedure 3.53** on  $A$  and  $p$ .

(i) **Abort procedure.**

5. Otherwise, do the following:

- (a) Execute **procedure 3.18** on  $D, I_t, t$  and let  $E$  receive.
- (b) Verify that  $C_t(D) = C_t(DI_t) = EC_t(I_t) = E * 1 = E$ .
- (c) Verify that  $E$  is a  $\mathcal{D}_{\binom{m^2}{t}, \binom{t}{t}}(\mathbb{Q}[x])$ .
- (d) Therefore verify that  $C_t(D)$  is a  $\mathcal{D}_{\binom{m^2}{t}, 1}(\mathbb{Q}[x])$ .
- (e) **Therefore verify that**  $C_t(D) = C_t(D)_{1,1}e_1 \neq 0$ .

### Procedure 3.55

#### Objective

Choose a  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \text{cols}(\text{pows}(A))$ . The objective of the following instructions is to show that either  $0 < 0$  or to show that  $C_t(\text{pows}(A)) \neq 0$ .

#### Implementation

1. Execute **procedure 3.20** on  $\text{pows}(A)$  and let the tuple  $\langle M, D, N \rangle$  receive the result.
2. Verify that  $\text{pows}(A) = M^{-1} * DN^{-1} *$ .
3. Execute **procedure 3.52** on  $C_t(M_*)$ ,  $C_t(M^{-1} *)$ .
4. Verify that all columns of  $C_t(M^{-1} *)$  are non-zero.
5. Let  $t = \text{cols}(\text{pows}(A))$ .
6. Execute **procedure 3.54** on  $A$ .
7. Verify that  $C_t(D) = C_t(D)_{1,1}e_1 \neq 0$ .
8. Therefore verify that  $C_t(D)_{1,1} \neq 0$ .
9. Execute **procedure 3.52** on  $C_t(N_*)$ ,  $C_t(N^{-1} *)$ .
10. Verify that  $C_t(N^{-1} *) \neq 0$ .
11. **Verify that**  $C_t(\text{pows}(A)) = C_t(M^{-1}DN^{-1}) = C_t(M^{-1})C_t(D)C_t(N^{-1}) = C_t(M^{-1})C_t(D)_{1,1}e_1C_t(N^{-1}) = C_t(D)_{1,1}C_t(N^{-1})C_t(M^{-1})e_1 \neq 0_{\binom{m^2}{t} \times 1}$ .

### Notation 3.22

Let us use the notation  $\text{mat}_t(p)$  as a shorthand for " $(x^{t-1} \circ p)e_1 + (x^{t-2} \circ p)e_2 + \dots + (x^0 \circ p)e_t$ ".

### Notation 3.23

Let us use the notation  $\text{pol}(P)$  as a shorthand for " $P_{1,1}x^{t-1} + P_{2,1}x^{t-2} + \dots + P_{t,1}$  where  $t = \text{rows}(P)$ ".

### Notation 3.24

Let us use the notation  $\|A\|^2$  as a shorthand for " $\sum_{i=1}^{\text{rows}(A)} \sum_{j=1}^{\text{cols}(A)} A_{i,j}^2$ ".

### Procedure 3.56

#### Objective

Choose an  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to either show that  $0 < 0$  or to construct a  $\mathbb{Q}[x]$ ,  $\text{sel}_A$ .

#### Implementation

1. Using **procedure 3.26** and **procedure 3.55**, verify that  $C_t(\text{pows}(A)^T \text{pows}(A)) = C_t(\text{pows}(A)^T)C_t(\text{pows}(A)) = C_t(\text{pows}(A))^T C_t(\text{pows}(A)) = \|C_t(\text{pows}(A))\|^2 > 0$ .
2. Let  $H = (\text{pows}(A)^T \text{pows}(A)) \setminus e_1$ .
3. Let  $t = \text{deg}(\text{last}_A)$ .
4. Let  $\text{sel}_A = \frac{\text{pol}(H)}{x^t \circ \text{last}_A}$ .
5. Yield  $\langle \text{sel}_A \rangle$ .

### Notation 3.25

Let us use the notation  $\text{sel}_A$  as a shorthand for the result yielded by executing **procedure 3.56** on  $A$ .

### Notation 3.26

Let us use the notation  $\text{tr}(X)$  as a shorthand for "the sum of the diagonal entries of the square matrix  $X$ ".

### Notation 3.27

Let us use the notation "A is symmetric" as a shorthand for " $A^T = A$ ".

### Procedure 3.57

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \deg(\text{last}_A)$ . Choose two  $\mathbb{Q}[x]$ s  $u = u_1x^{t-1} + u_2x^{t-2} + \dots + u_tx^0$ ,  $w = w_1x^{t-1} + w_2x^{t-2} + \dots + w_tx^0$ . The objective of the following instructions is to show that  $\text{tr}(u(A)w(A)) = \text{mat}(u)^T \text{pows}(A)^T \text{pows}(A) \text{mat}_t(w)$ .

#### Implementation

1. Verify that  $\text{tr}(u(A)w(A))$ 
  - (a)  $= \text{tr}((\sum_{p=1}^t u_p A^{t-p})(\sum_{q=1}^t w_q A^{t-q}))$
  - (b)  $= \text{tr}(\sum_{p=1}^t \sum_{q=1}^t u_p w_q A^{t-p} A^{t-q})$
  - (c)  $= \sum_{p=1}^t \sum_{q=1}^t u_p w_q \text{tr}(A^{t-p} A^{t-q})$
  - (d)  $= \sum_{p=1}^t \sum_{q=1}^t u_p w_q \sum_{e=1}^m \sum_{f=1}^m A^{t-p}_{e,f} A^{t-q}_{f,e}$
  - (e)  $= \sum_{p=1}^t \sum_{q=1}^t u_p w_q \sum_{e=1}^m \sum_{f=1}^m A^{t-p}_{f,e} A^{t-q}_{f,e}$
  - (f)  $= \sum_{p=1}^t \sum_{q=1}^t u_p w_q \sum_{g=1}^{m^2} \text{pows}(A)_{g,p} \text{pows}(A)_{g,q}$
  - (g)  $= \sum_{p=1}^t \sum_{q=1}^t u_p w_q (\text{pows}(A)^T \text{pows}(A))_{p,q}$
  - (h)  $= \sum_{p=1}^t u_p (\text{pows}(A)^T \text{pows}(A) \text{mat}_t(w))_p$
  - (i)  $= \text{mat}_t(u)^T \text{pows}(A)^T \text{pows}(A) \text{mat}_t(w)$

### Procedure 3.58

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \deg(\text{last}_A)$ . Choose a  $\mathbb{Q}[x]$   $u$  such that  $\deg(u) < t$ . The objective of the following instructions is to show that  $\text{tr}(u(A) \text{sel}_A(A)) = \frac{x^{t-1} \circ u}{x^t \circ \text{last}_A}$ .

### Implementation

1. Using [procedure 3.57](#) and [procedure 3.56](#), verify that  $\text{tr}(u(A) \text{sel}_A(A))$ 
  - (a)  $= \text{mat}(u)^T \text{pows}(A)^T \text{pows}(A) \text{mat}_t(\text{sel}_A)$
  - (b)  $= \frac{\text{mat}(u)^T \text{pows}(A)^T \text{pows}(A) ((\text{pows}(A)^T \text{pows}(A)) \setminus e_1)}{x^t \circ \text{last}_A}$
  - (c)  $= \frac{\text{mat}(u)^T e_1}{x^t \circ \text{last}_A}$
  - (d)  $= \frac{\text{mat}(u)_{1,1}}{x^t \circ \text{last}_A}$
  - (e)  $= \frac{x^{t-1} \circ u}{x^t \circ \text{last}_A}$ .

### Procedure 3.59

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to either show that  $0 \neq 0$  or construct  $\mathbb{Q}[x]$ s  $u, v$  such that  $u \text{last}_A + v \text{sel}_A = 1$ .

#### Implementation

1. Let  $t = \deg(\text{last}_A)$ .
2. Let  $G$  be a  $\mathcal{M}_{1,2}(\mathbb{Q}[x])$  where  $G_{1,1} = \text{last}_A$  and  $G_{1,2} = \text{sel}_A$ .
3. Execute [procedure 3.20](#) on  $G$  and let the tuple  $\langle M, D, \cdot, N \rangle$  receive.
4. Verify that  $G = M^{-1} \cdot D N^{-1} \cdot$ .
5. Verify that  $\text{last}_A \neq 0$ .
6. Therefore verify that  $D_{1,1} \neq 0$ .
7. If  $\deg(D_{1,1}) > 0$ , then do the following:
  - (a) Let  $b = N^{-1} \cdot_{1,1}$ .
  - (b) Verify that  $\text{last}_A = b D_{1,1}$ .
  - (c) Let  $z = \deg(b)$ .
  - (d) Verify that  $t = \deg(\text{last}_A) = \deg(b D_{1,1}) = \deg(b) + \deg(D_{1,1}) > \deg(b) = z$ .
  - (e) Let  $c = N^{-1} \cdot_{1,2}$ .
  - (f) Verify that  $\text{sel}_A = c D_{1,1}$ .
  - (g) Let  $u = x^{t-z-1} b$ .
  - (h) Execute [procedure 3.58](#) on  $A$  and  $u$ .

- (i) Hence verify that  $\text{tr}(u(A)\text{sel}_A(A)) = x^{t-1} \circ u = x^z \circ b \neq 0$ .
- (j) Also verify that  $\begin{aligned} \text{tr}(u(A)\text{sel}_A(A)) &= \\ \text{tr}(A^{z-1}b(A)c(A)D_{1,1}(A)) &= \\ \text{tr}(A^{z-1}c(A)b(A)D_{1,1}(A)) &= \\ \text{tr}(A^{z-1}c(A)\text{last}_A(A)) &= \\ \text{tr}(A^{z-1}c(A)0_{m \times m}) &= \text{tr}(0_{m \times m}) = 0. \end{aligned}$
- (k) Therefore verify that  $0 \neq 0$ .
- (l) **Abort procedure.**
- 8. Otherwise, do the following:
  - (a) Verify that  $\deg(D_{1,1}) = 0$ .
  - (b) Let  $u = \frac{N_{1,1}}{D_{1,1}}$ .
  - (c) Let  $v = \frac{N_{2,1}}{D_{1,1}}$ .
  - (d) **Verify that**  $u\text{last}_A + v\text{sel}_A = 1$ .
  - (e) **Yield the tuple**  $\langle u, v \rangle$ .

### Procedure 3.60

#### Objective

Choose two  $\mathbb{Q}[x]$ s  $\langle a, b \rangle$ . The objective of the following instructions is to construct two  $\mathbb{Q}[x]$ s  $u, w$  such that  $a = ub + w$  and  $\deg(w) < \deg(b)$ .

#### Implementation

1. If  $\deg(a) \geq \deg(b)$ :
  - (a) Let  $y = \frac{x^{\deg(a)} \circ a}{x^{\deg(b)} \circ b} x^{\deg(a) - \deg(b)}$
  - (b) Let  $e = a - yb$ .
  - (c) Verify that  $\deg(e) < \deg(a)$ .
  - (d) Execute **procedure 3.60** on the tuple  $\langle e, b \rangle$ . Let the tuple  $\langle c, d \rangle$  receive the result.
  - (e) Verify that  $cb + d = e$ .
  - (f) Verify that  $\deg(d) < \deg(b)$ .
  - (g) Therefore verify that  $cb + d = a - yb$
  - (h) **Therefore verify that**  $(y + c)b + d = a$ .
  - (i) **Also verify that**  $\deg(d) < \deg(b)$ .
  - (j) **Now yield the tuple**  $\langle y + c, d \rangle$ .
2. Otherwise:

- (a) **Verify that**  $0 * b + a = a$ .
- (b) **Verify that**  $\deg(a) < \deg(b)$ .
- (c) **Yield the tuple**  $\langle 0, a \rangle$ .

### Procedure 3.61

#### Objective

Choose two lists of  $\mathbb{Q}[x]$ s  $s, q$  and a non-negative integer  $k$  in such a way that, letting  $m = |s| - 1$ ,

1.  $k < m$ .
2. For  $k \leq i \leq m$ ,  $\deg(s_i) = i$ .
3. For  $k < i < m$ ,  $s_{i-1} + s_{i+1} = q_i s_i$ .

Let  $\deg(0) = -1$ . The objective of the following instructions is to construct  $\mathbb{Q}[x]$ s  $g, h$  such that  $s_k = gs_{m-1} + hs_m$ ,  $\deg(g) = m - 1 - k$ , and  $\deg(h) = m - 2 - k$ .

#### Implementation

1. If  $k < m - 2$ , do the following:
  - (a) Verify that  $s_k + s_{k+2} = q_{k+1}s_{k+1}$ .
  - (b) Therefore verify that  $s_k = q_{k+1}s_{k+1} - s_{k+2}$ .
  - (c) Execute **procedure 3.61** on  $s, q, k + 1$  and let the tuple  $\langle g_1, h_1 \rangle$  receive.
  - (d) Verify that  $s_{k+1} = g_1s_{m-1} + h_1s_m$ .
  - (e) Verify that  $\deg(g_1) = m - 1 - (k + 1) = m - k - 2$ .
  - (f) Verify that  $\deg(h_1) = m - 2 - (k + 1) = m - k - 3$ .
  - (g) Execute **procedure 3.61** on  $s, q, k + 2$  and let the tuple  $\langle g_2, h_2 \rangle$  receive.
  - (h) Verify that  $s_{k+2} = g_2s_{m-1} + h_2s_m$ .
  - (i) Verify that  $\deg(g_2) = m - 1 - (k + 2) = m - k - 3$ .
  - (j) Verify that  $\deg(h_2) = m - 2 - (k + 2) = m - k - 4$ .
  - (k) Let  $g = q_{k+1}g_1 - g_2$ .
  - (l) **Verify that**  $\deg(g) = \max(1 + (m - k - 2), m - k - 3) = m - 1 - k$ .

- (m) Let  $h = q_{k+1}h_1 - h_2$ .
- (n) **Verify that**  $\deg(h) = \max(1 + (m - k - 3), m - k - 4) = m - 2 - k$ .
- (o) **Verify that**  $s_k = q_{k+1}(g_1s_{m-1} + h_1s_m) - (g_2s_{m-1} + h_2s_m) = (q_{k+1}g_1 - g_2)s_{m-1} + (q_{k+1}h_1 - h_2)s_m = gs_{m-1} + hs_m$ .
- 2. Otherwise, if  $k = m - 2$  do the following:
  - (a) Verify that  $s_{m-2} + s_m = q_{m-1}s_{m-1}$ .
  - (b) Let  $g = q_{m-1}$ .
  - (c) **Verify that**  $\deg(g) = 1 = m - 1 - k$ .
  - (d) Let  $h = -1$ .
  - (e) **Verify that**  $\deg(h) = 0 = m - 2 - k$ .
  - (f) **Therefore verify that**  $s_k = s_{m-2} = q_{m-1}s_{m-1} - s_m = gs_{m-1} + hs_m$ .
- 3. Otherwise, if  $k = m - 1$  do the following:
  - (a) Let  $g = 1$ .
  - (b) **Verify that**  $\deg(g) = 0 = m - 1 - k$ .
  - (c) Let  $h = 0$ .
  - (d) **Verify that**  $\deg(h) = -1 = m - 2 - k$ .
  - (e) **Verify that**  $s_k = s_{m-1} = gs_{m-1} + hs_m$ .
- 4. **Yield the tuple**  $\langle g, h \rangle$ .

## Procedure 3.62

### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \deg(\text{last}_A)$ . The objective of the following instructions is to either show that  $0 \neq 0$  or to construct lists of  $\mathbb{Q}[x]$ s  $s, q$  such that

- 1. For  $i = 0$  to  $i = t$ ,  $\deg(s_i) = i$ .
- 2. For  $i = 0$  to  $i = t$ ,  $\text{sgn}(x^i \circ s_i) = \text{sgn}(x^t \circ s_t)$ .
- 3. For  $i = 1$  to  $i = t - 1$ ,  $s_{i-1} + s_{i+1} = q_i s_i$ .
- 4.  $s_t = \text{last}_A$ .

### Implementation

- 1. Execute **procedure 3.59** on  $A$  and let  $\langle u, s_{t+1} \rangle$  receive the result.

- 2. Verify that  $us_t + s_{t+1} \text{sel}_A = 1$ .
- 3. Execute **procedure 3.60** on the tuple  $\langle s_{t+1}, s_t \rangle$ . Let the tuple  $\langle q_t, s_{t-1} \rangle$  receive the result.
- 4. Verify that  $s_{t+1} = q_t s_t + s_{t-1}$ , where  $\deg(s_{t-1}) < \deg(s_t) = t$ .
- 5. Therefore verify that  $us_t + (q_t s_t + s_{t-1}) \text{sel}_A = 1$ .
- 6. Therefore verify that  $s_{t-1}(A) \text{sel}_A(A) = u(A)s_t(A) + (q_t(A)s_t(A) + s_{t-1}(A)) \text{sel}_A(A) = I_{m,m}$ .
- 7. Therefore using **procedure 3.58**, verify that  $\frac{x^{t-1} \circ s_{t-1}}{x^t \circ s_t} = \text{tr}(s_{t-1}(A) \text{sel}_A(A)) = \text{tr}(I_{m,m}) = m > 0$ .
- 8. For  $i = t - 1$  down to  $i = 1$ , do the following:
  - (a) Execute **procedure 3.60** on the tuple  $\langle -s_{i+1}, -s_i \rangle$ . Let the tuple  $\langle q_i, s_{i-1} \rangle$  receive the result.
  - (b) Verify that  $\deg(q_i) = 1$ .
  - (c) Verify that  $x \circ q_i = \frac{x^{i+1} \circ s_{i+1}}{x^i \circ s_i}$ .
  - (d) Also verify that  $-s_{i+1} = -q_i s_i + s_{i-1}$ .
  - (e) Therefore verify that  $q_i s_i = s_{i+1} + s_{i-1}$ .
  - (f) Therefore verify that  $q_i s_i - s_{i+1} = s_{i-1}$ .
  - (g) Execute **procedure 3.61** on the tuple  $\langle s, q, i - 1 \rangle$  and let  $\langle p, j \rangle$  receive.
  - (h) Verify that  $s_{i-1} = ps_{t-1} + q_3 s_t$ .
  - (i) Verify that  $\deg(p) = t - 1 - (i - 1) = t - i$ .
  - (j) Verify that  $\deg(q_3) = t - 2 - (i - 1) = t - 1 - i$ .
  - (k) Therefore verify that  $s_{i-1}(A) = p(A)s_{t-1}(A) + j(A)s_t(A) = p(A)s_{t-1}(A) + j(A)0_{m \times m} = p(A)s_{t-1}(A)$ .
  - (l) If  $p(A) = 0$ , then do the following:
    - i. Execute **procedure 3.53** on  $A$  and  $p$ .
    - ii. **Abort procedure.**
- (m) Otherwise, if  $s_{i-1}(A) = 0_{m \times m}$ , then do the following:
  - i. Verify that  $p(A)s_{t-1}(A) \text{sel}_A(A) = s_{i-1}(A) \text{sel}_A(A) = 0_{m \times m} \text{sel}_A(A) = 0_{m \times m}$ .
  - ii. Verify that  $p(A)s_{t-1}(A) \text{sel}_A(A) = p(A)I_{m,m} = p(A) \neq 0_{m \times m}$ .
  - iii. Therefore verify that  $0 \neq 0$ .

iv. **Abort procedure.**

(n) Otherwise if  $s_{i-1}(A) \text{sel}_A(A) = 0_{m \times m}$ , then do the following:

i. Verify that  $s_{i-1}(A) \text{sel}_A(A) s_{t-1}(A) = 0_{m \times m} s_{t-1}(A) = 0_{m \times m}$ .

ii. Verify that  $s_{i-1}(A) \text{sel}_A(A) s_{t-1}(A) = s_{i-1}(A) I_{m,m} = s_{i-1}(A) \neq 0$ .

iii. Therefore verify that  $0 \neq 0$ .

iv. **Abort procedure.**

(o) Otherwise, do the following:

i. Verify that  $\deg(s_{i-1}) < i$ .

ii. Verify that  $s_{i-1}(A) \text{sel}_A(A) \neq 0_{m \times m}$ .

iii. Execute the **auxilliary procedure** on the tuple  $(i-1, s_{i-1})$ .

iv. Hence verify that  $\frac{x^{i-1} \circ s_{i-1}}{x^i \circ s_i} = \frac{\text{tr}(s_{i-1}(A)^2 \text{sel}_A(A)^2)}{\text{tr}((s_{i-1}(A) \text{sel}_A(A))^2)} = \frac{\|s_{i-1}(A) \text{sel}_A(A)\|^2}{\|s_{i-1}(A) \text{sel}_A(A)\|^2} > 0$ .

v. **Therefore verify that**  $\text{sgn}(x^{i-1} \circ s_{i-1}) = \text{sgn}(x^i \circ s_i)$ .

9. Yield the tuple  $\langle s_{[0:t+1]}, q_{[0:t]} \rangle$ .

### Auxilliary procedure

**Objective** Choose an integer  $0 \leq k \leq t$  such that polynomial  $s_k$  is defined. Choose a  $\mathbb{Q}[x]$   $g$  such that  $\deg(g) \leq \min(k, t-1)$ . The objective of the following instructions is to show that  $\text{tr}(g(A)s_k(A) \text{sel}_A(A)^2) = \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}}$ .

### Implementation

1. If  $k = t$ , then verify that  $\text{tr}(g(A)s_k(A) \text{sel}_A(A)^2)$

(a)  $= \text{tr}(g(A)s_t(A) \text{sel}_A(A)^2)$

(b)  $= \text{tr}(g(A)0_{m \times m} \text{sel}_A(A)^2)$

(c)  $= 0$

(d)  $= \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}}$ .

2. Otherwise if  $k = t-1$ , then verify that  $\text{tr}(g(A)s_k(A) \text{sel}_A(A)^2)$

(a)  $= \text{tr}(g(A)s_{t-1}(A) \text{sel}_A(A)^2)$ .

(b)  $= \text{tr}(g(A)I_{m,m} \text{sel}_A(A))$ .

(c)  $= \text{tr}(g(A) \text{sel}_A(A))$ .

(d)  $= \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}}$ .

3. Otherwise if  $k < t-1$ , then do the following:

(a) Verify that  $\deg(gq_{k+1}) = k+1 \leq t-1$ .

(b) Execute the **auxilliary procedure** on the tuple  $\langle k+1, gq_{k+1} \rangle$ .

(c) Now verify that  $\text{tr}((g(A)q_{k+1}(A)s_{k+1}(A) \text{sel}_A(A)^2) = \frac{\frac{x^{k+2} \circ s_{k+2}}{x^{k+1} \circ s_{k+1}} x^k \circ g}{x^{k+2} \circ s_{k+2}} = \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}}$ .

(d) Verify that  $\deg(g) \leq k \leq t-2$ .

(e) Execute the **auxilliary procedure** on the tuple  $\langle k+2, g \rangle$ .

(f) Now verify that  $\text{tr}(g(A)s_{k+2}(A) \text{sel}_A(A)^2) = \frac{x^{k+2} \circ g}{x^{k+3} \circ s_{k+3}} = \frac{0}{x^{k+3} \circ s_{k+3}} = 0$ .

(g) Therefore verify that  $\text{tr}(g(A)s_k(A) \text{sel}_A(A)^2)$

i.  $= \text{tr}(g(A)(q_{k+1}(A)s_{k+1}(A) + s_{k+2}(A)) \text{sel}_A(A)^2)$

ii.  $= \text{tr}(g(A)q_{k+1}(A)s_{k+1}(A) \text{sel}_A(A)^2) + \text{tr}(g(A)s_{k+2}(A) \text{sel}_A(A)^2)$

iii.  $= \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}} + 0$

iv.  $= \frac{x^k \circ g}{x^{k+1} \circ s_{k+1}}$ .

### Procedure 3.63

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \deg(\text{last}_A)$ . The objective of the following instructions is to either show that  $0 < 0$  or to construct two lists of rational numbers  $c, d$  such that  $c_1 < d_1 \leq c_2 < d_2 \leq \dots \leq c_t < d_t$  and  $\text{sgn}(\text{last}_A(c_i)) = -\text{sgn}(\text{last}_A(d_i))$  for  $i = 1$  to  $i = t$ .

#### Implementation

1. Execute **procedure 3.62** on the matrix  $A$  and let the tuple  $\langle s, q \rangle$  receive the result.

2. Execute **procedure 2.09** supplying the tuple  $\langle s, q \rangle$ . Let the tuple  $\langle c, d \rangle$  receive the result.

3. **Verify that**  $c_1 < d_1 \leq c_2 < d_2 \leq \dots \leq c_t < d_t$ .
4. **Verify that**  $\text{sgn}(\text{last}_A(c_i)) = -\text{sgn}(\text{last}_A(d_i))$  **for**  $i = 1$  **to**  $i = t$ .
5. **Yield**  $\langle c, d \rangle$ .

### Procedure 3.64

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Let  $t = \deg(\text{last}_A)$ . Execute **procedure 3.63** on  $A$  and let the tuple  $\langle c, d \rangle$  receive the result. Execute **procedure 3.20** on  $A$  and let the tuple  $\langle, , u, \rangle$  receive the result. The objective of the following instructions is to either show that  $1 = -1$  or to construct a list of non-negative integers  $k$  such that  $\text{sgn}(u_{k_i}(c_i)) = -\text{sgn}(u_{k_i}(d_i))$  for  $i = 1$  to  $i = t$ .

#### Implementation

1. Verify that  $\text{last}_A = u_1 u_2 \dots u_m$ .
2. For  $i = 1$  to  $i = t$  do the following:
  - (a) If  $\text{sgn}(u_1(c_i)) = \text{sgn}(u_1(d_i)), \text{sgn}(u_2(c_i)) = \text{sgn}(u_2(d_i)), \dots, \text{sgn}(u_m(c_i)) = \text{sgn}(u_m(d_i))$ , then do the following:
    - i. Verify that  $\text{sgn}(u_1(c_i)) \text{sgn}(u_2(c_i)) \dots \text{sgn}(u_m(c_i)) \neq \text{sgn}(u_1(d_i)) \text{sgn}(u_2(d_i)) \dots \text{sgn}(u_m(d_i))$ .
    - ii. Therefore verify that  $\frac{\text{sgn}(u_1(c_i)u_2(c_i) \dots u_m(c_i))}{\text{sgn}(u_1(d_i)u_2(d_i) \dots u_m(d_i))} =$
    - iii. Therefore verify that  $\text{sgn}(s_t(c_i)) = \text{sgn}(s_t(d_i))$ .
    - iv. Cognizant of the execution of **procedure 2.08**, verify that  $\text{sgn}(s_t(c_i)) = -\text{sgn}(s_t(d_i))$ .
    - v. Therefore verify that  $\text{sgn}(s_t(c_i)) = -\text{sgn}(s_t(d_i))$ .
    - vi. Therefore verify that  $1 = -1$ .
    - vii. **Abort procedure.**
  - (b) Otherwise do the following:
    - i. **Let  $j$  be the least integer such that**  $\text{sgn}(u_j(c_i)) = -\text{sgn}(u_j(d_i))$ .

- ii. **Let**  $k_i = j$ .

3. **Yield**  $\langle k \rangle$ .

### Procedure 3.65

#### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute **procedure 3.20** on  $A$  and let the tuple  $\langle, , u, \rangle$  receive the result. Execute **procedure 2.03** on  $A$  and let  $k$  receive. Let  $t = \deg(\text{last}_A)$ . Let  $n_j = \sum_{i=1}^t [k_i = j]$ . The objective of the following instructions is to either show that  $0 < 0$ , or to show that  $n_i = \deg(u_i)$  for  $i = 1$  to  $i = m$ .

#### Implementation

1. Verify that  $\sum_{j=1}^m n_j = \sum_{j=1}^m \sum_{i=1}^t [k_i = j] = \sum_{i=1}^t \sum_{j=1}^m [k_i = j] = \sum_{i=1}^t 1 = t$ .
2. If for any  $i = 1$  to  $i = m$ ,  $n_i > \deg(u_i)$ , then do the following:
  - (a) Execute **procedure 2.03** on the polynomial  $u_i$  along with  $\deg(u_i) + 1$  of the distinct pairs  $\langle c_l, d_l \rangle$  such that  $k_l = i$ .
3. Otherwise if for any  $i = 1$  to  $i = m$ ,  $n_i < \deg(u_i)$ , then do the following:
  - (a) Verify that  $\sum_{i=1}^m n_j < \sum_{i=1}^m \deg(u_j) = t$ .
  - (b) Therefore using (1) and (a), verify that  $\sum_{i=1}^m n_j < \sum_{i=1}^m n_j$ .
  - (c) **Abort procedure.**
4. Otherwise, do the following:
  - (a) **For all  $i = 1$  to  $i = m$ , verify that**  $n_i = \deg(u_i)$ .

### Notation 3.28

Let us use the notation " $A$  is upper triangular" as a shorthand for "all the entries of  $A$  below the diagonal are zero" in what follows.



### Procedure 3.66

#### Objective

Choose two upper triangular  $\mathcal{M}_{m,m}(\mathbb{Q}[x])$ s,  $A$  and  $B$ . Let  $C = AB$ . The objective of the following instructions is to show that  $C$  is an upper triangular matrix where  $C_{i,i} = A_{i,i}B_{i,i}$  for  $i = 1$  to  $i = m$ .

#### Implementation

1. For  $i = 1$  to  $i = m$ , do the following:
  - (a) **Verify that**  $C_{i,i} = \sum_{k=1}^m (A_{i,k}B_{k,i}) = \sum_{k=1}^{i-1} (A_{i,k}B_{k,i}) + A_{i,i}B_{i,i} + \sum_{k=i+1}^m (A_{i,k}B_{k,i}) = \sum_{k=1}^{i-1} (0 * B_{k,i}) + A_{i,i}B_{i,i} + \sum_{k=i+1}^m (A_{i,k} * 0) = A_{i,i}B_{i,i}$ .
2. For  $i = 2$  to  $i = m$ , do the following:
  - (a) For  $j = 1$  to  $j = i - 1$ , do the following:
    - i. Verify that  $C_{i,j} = \sum_{k=1}^m A_{i,k}B_{k,j} = \sum_{k=1}^{i-1} A_{i,k}B_{k,j} + \sum_{k=i}^m A_{i,k}B_{k,j} = \sum_{k=1}^{i-1} 0 * B_{k,j} + \sum_{k=i}^m A_{i,k} * 0 = 0$ .
3. **Therefore verify that  $C$  is upper triangular.**

### Procedure 3.67

#### Objective

Choose integers  $m \geq n \geq 0$ . Choose a  $\mathcal{M}_{n,m}(\mathbb{Q}[x])$ ,  $M$ , and a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A_0$ , such that  $MA_0 = I_n$ . The objective of the following instructions is to either show that  $1 = 0$  or to construct  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ s  $A_1, A_2, \dots, A_n$ .

#### Implementation

1. Using [procedure 3.10](#), verify that  $C_n(M_0A_0) = C_n(I_n) = 1$ .
2. If  $C_n(A_0) = 0_{\binom{m}{n} \times 1}$ , then do the following:
  - (a) Verify that  $C_n(M_0A_0) = C_n(M_0)C_n(A_0) = C_n(M_0)0_{\binom{m}{n} \times 1} = 0$ .
  - (b) **Therefore verify that  $1 = 0$ .**
  - (c) **Abort procedure.**

3. Verify that  $C_n(A_0) \neq 0_{\binom{m}{n} \times 1}$ .
4. For  $i = 1$  to  $i = n$ , do the following:
  - (a) If  $A_{i-1}e_i = 0_{m \times 1}$ , then do the following:
    - i. Verify that  $C_n(A_{i-1}) = 0$ .
    - ii. Cognizant of the execution of the previous iteration, verify that  $C_n(A_{i-1}) \neq 0$ .
    - iii. **Therefore verify that  $0 \neq 0$ .**
    - iv. **Abort procedure.**
  - (b) Verify that  $\|A_{i-1}e_i\|^2 \neq 0$ .
  - (c) Let  $D_i$  be a  $n \times n$  diagonal matrix comprising  $i$  1s followed by  $n - i$   $\|A_{i-1}e_i\|^2$ s.
  - (d) Verify that  $C_n(D_i) = (\|A_{i-1}e_i\|^2)^{n-i} \neq 0$ .
  - (e) Let  $N_i = I_n$  except that its  $i^{th}$  row is  $i - 1$  0s followed by a 1 followed by  $-(A_{i-1}^T A_{i-1})_{i,i+1}$ , then  $-(A_{i-1}^T A_{i-1})_{i,i+2}$ , all the way up to  $-(A_{i-1}^T A_{i-1})_{i,n}$ .
  - (f) Using [procedure 3.10](#), verify that  $C_n(N_i) = 1 \neq 0$ .
  - (g) Let  $A_i = A_{i-1}D_iN_i$ .
  - (h) **Verify that**  $C_n(A_i) = C_n(A_{i-1}D_iN_i) = C_n(A_{i-1})C_n(D_i)C_n(N_i) = C_n(A_{i-1})C_n(D_i) \neq 0$ .
5. **Yield the tuple  $\langle A_0, A_1, \dots, A_n \rangle$ .**

### Procedure 3.68

#### Objective

Choose integers  $m \geq n \geq 0$ . Choose a  $\mathcal{M}_{n,m}(\mathbb{Q}[x])$ ,  $M$ , and a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A_0$ , such that  $MA_0 = I_n$ . Execute [procedure 3.67](#) on  $M$  and  $A_0$  and let the tuple  $\langle A_1, \dots, A_n \rangle$  receive the result. The objective of the following instructions is to either show that  $1 = 0$  or to show that  $(A_i^T A_i)_{[1:i+1],[1:i+1]}$  is a  $\mathcal{D}_{i,i}(\mathbb{Q}[x])$  and  $A_i^T A_i = \text{bdiag}((A_i^T A_i)_{[1:i+1],[1:i+1]}, (A_i^T A_i)_{[i+1:n+1],[i+1:n+1]})$  for  $i = 1$  to  $i = n$ .

#### Implementation

1. For  $i = 1$  to  $i = n$ , do the following:

- (a) Let  $D_i$  be a  $n \times n$  diagonal matrix comprising  $i$  1s followed by  $n - i$   $\|A_{i-1}e_i\|^2$ s.
- (b) Let  $N_i = I_n$  except that its  $i^{th}$  row is  $i - 1$  0s followed by a 1 followed by  $-(A_{i-1}^T A_{i-1})_{i,i+1}$ , then  $-(A_{i-1}^T A_{i-1})_{i,i+2}$ , all the way up to  $-(A_{i-1}^T A_{i-1})_{i,n}$ .
- (c) Verify that  $A_i = A_{i-1} D_i N_i$ .
- (d) Verify that  $A_i^T A_i = (A_{i-1} D_i N_i)^T (A_{i-1} D_i N_i) = N_i^T D_i^T (A_{i-1}^T A_{i-1}) D_i N_i$ .
- (e) Now using [procedure 3.08](#), verify that  $A_i^T A_i$  and  $A_{i-1}^T A_{i-1}$  are the same modulo the bottom-right  $(n - i + 1) \times (n - i + 1)$  block.
- (f) **Therefore using (1e) and the previous instance of (1i), verify that  $(A_i^T A_i)_{[1:i+1],[1:i+1]}$  is a  $\mathcal{D}_{i,i}(\mathbb{Q}[x])$ .**
- (g) Also verify that  $(A_i^T A_i)_{i,[i+1:n+1]} = 0$ .
- (h) Also verify that  $(A_i^T A_i)_{[i+1:n+1],i} = 0$ .
- (i) **Therefore using (1g), (1h), and the previous instance of (1i), verify that  $A_i^T A_i = \text{bdiag}((A_i^T A_i)_{[1:i+1],[1:i+1]}, (A_i^T A_i)_{[i+1:n+1],[i+1:n+1]})$ .**

## Procedure 3.69

### Objective

Choose integers  $m \geq n \geq 0$ . Choose a  $\mathcal{M}_{n,m}(\mathbb{Q}[x])$ ,  $M$ , and a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A_0$ , such that  $MA_0 = I_n$ . Execute [procedure 3.67](#) on  $M$  and  $A_0$  and let the tuple  $\langle A_1, \dots, A_n \rangle$  receive the result. The objective of the following instructions is to either show that  $1 = 0$  or to show that  $A_0 M A_i = A_i$  and  $(e_j^T M)(A_i e_j) = \|A_0 e_1\|^2 \cdots \|A_{\min(i,j-1)-1} e_{\min(i,j-1)}\|^2$  for  $j = 1$  to  $j = n$ , for  $i = 1$  to  $i = n$ .

### Implementation

1. For  $i = 1$  to  $i = n$ , do the following:
  - (a) Let  $D_i$  be a  $n \times n$  diagonal matrix comprising  $i$  1s followed by  $n - i$   $\|A_{i-1}e_i\|^2$ s.
  - (b) **Verify that  $D_i$  is upper triangular.**
  - (c) Let  $N_i = I_n$  except that its  $i^{th}$  row is  $i - 1$  0s followed by a 1 followed by

$-(A_{i-1}^T A_{i-1})_{i,i+1}$ , then  $-(A_{i-1}^T A_{i-1})_{i,i+2}$ , all the way up to  $-(A_{i-1}^T A_{i-1})_{i,n}$ .

- (d) **Verify that  $N_i$  is upper triangular.**
- (e) Verify that  $A_i = A_{i-1} D_i N_i$ .
- (f) Verify that  $A_i = A_0 (D_1 N_1) \cdots (D_i N_i)$ .
- (g) Verify that  $MA_i = (D_1 N_1) \cdots (D_i N_i)$ .
- (h) **Therefore verify that  $A_0 M A_i = A_i$ .**
- (i) Using [procedure 3.66](#), for  $j = 1$  to  $j = n$ , verify that  $(e_j^T M)(A_i e_j)$ 
  - i.  $= e_j^T (MA_i) e_j$
  - ii.  $= e_j^T ((D_1 N_1) \cdots (D_i N_i)) e_j$
  - iii.  $= (D_{1,j,j} N_{1,j,j}) \cdots (D_{i,j,j} N_{i,j,j})$
  - iv.  $= D_{1,j,j} \cdots D_{i,j,j}$
  - v.  $= D_{1,j,j} \cdots D_{\min(i,j-1),j,j}$
  - vi.  $= \|A_0 e_1\|^2 \cdots \|A_{\min(i,j-1)-1} e_{\min(i,j-1)}\|^2$ .

## Procedure 3.70

### Objective

Choose a  $\mathcal{M}_{1,m}(\mathbb{Q})$ ,  $A$ , and a  $\mathcal{M}_{m,1}(\mathbb{Q})$ ,  $B$ . The objective of the following instructions is to show that  $(AB)^2 \leq (AA^T)(B^T B)$ .

### Implementation

1. Verify that 0
  - (a)  $\leq \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (A_i B_j - A_j B_i)^2$
  - (b)  $= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (A_i^2 B_j^2 - 2A_i B_j A_j B_i + A_j^2 B_i^2)$
  - (c)  $= \frac{1}{2} \sum_{i=1}^m A_i^2 \sum_{j=1}^m B_j^2 + \frac{1}{2} \sum_{i=1}^m B_i^2 \cdot \sum_{j=1}^m A_j^2 - \sum_{i=1}^m A_i B_i \sum_{j=1}^m A_j B_j$
  - (d)  $= \frac{1}{2} (AA^T)(B^T B) + \frac{1}{2} (AA^T)(B^T B) - (AB)^2$
  - (e)  $= (AA^T)(B^T B) - (AB)^2$ .
2. **Therefore verify that  $(AB)^2 \leq (AA^T)(B^T B)$ .**

## Procedure 3.71

### Objective

Choose integers  $m \geq n > 0$ . Choose a  $\mathcal{M}_{n,m}(\mathbb{Q}[x])$ ,  $M$ , and a  $\mathcal{M}_{m,n}(\mathbb{Q}[x])$ ,  $A_0$ , such that  $MA_0 = I_n$ . Choose a  $\mathbb{Q}$ ,  $x$ . Let  $a = \max(\|M(x)\|^2, 1)$ . Choose a column index  $1 \leq j \leq n$  such that  $\|A_n(x)e_j\|^2 < \frac{1}{a^{(2n+2)!!}}$ . The objective of the following instructions is to show that  $1 < 1$ .

### Implementation

1. Execute **procedure 3.67** on  $M$  and  $A_0$  and let the tuple  $\langle A_0, A_1, \dots, A_n \rangle$  receive the result.
2. Let  $i = n$ .
3. Verify that  $\|A_i(x)e_j\|^2 < \frac{1}{a^{(2i+2)!!}}$ .
4. Using **procedure 3.70**, verify that  $(e_j^T M(x)A_i(x)e_j)^2 \leq \|e_j^T M(x)\|^2 \|A_i(x)e_j\|^2 < \|M(x)\|^2 \frac{1}{a^{(2i+2)!!}} \leq a \frac{1}{a^{(2i+2)!!}} \leq \frac{1}{a^{(2i)!! * 2i}} \leq 1$ .
5. If  $i = 0$ , then do the following:
  - (a) Verify that  $(e_j^T M(x)A_i(x)e_j)^2 = (e_j^T M(x)A_0(x)e_j)^2 = (e_j^T I_n e_j)^2 = 1$ .
  - (b) Therefore using (4) and (a), verify that  $1 < 1$ .
  - (c) **Abort procedure.**
6. Otherwise, do the following:
  7. Using **procedure 3.69**, verify that  $(1\|A_0 e_1\|^2 \cdots \|A_{\min(i,j-1)-1} e_{\min(i,j-1)}\|^2)^2 = (e_j^T M(x)A_i(x)e_j)^2 < \frac{1}{a^{(2i)!! * 2i}} \leq 1$ .
  8. If  $\min(i, j-1) = 0$ , then do the following:
    - (a) Verify that  $(1\|A_0(x)e_1\|^2 \cdots \|A_{\min(i,j-1)-1}(x)e_{\min(i,j-1)}\|^2)^2 = 1^2 = 1$ .
    - (b) Therefore using (7) and (a), verify that  $1 < 1$ .
    - (c) **Abort procedure.**
  9. Otherwise do the following:
    - (a) Verify that  $\min(i, j-1) > 0$ .
    - (b) If for all  $k = 0$  to  $k = \min(i, j-1) - 1$ ,  $\|A_k(x)e_{k+1}\|^2 \geq \frac{1}{a^{(2i)!!}}$ , then do the following:
      - i. Verify that  $(e_j^T M(x)A_i(x)e_j)^2 = (\|A_0(x)e_1\|^2 \cdots$

$$\|A_{\min(i,j-1)-1}(x)e_{\min(i,j-1)}\|^2)^2 \geq \left(\frac{1}{a^{(2i)!!}}\right)^{2\min(i,j-1)} \geq \left(\frac{1}{a^{(2i)!!}}\right)^{2i} = \frac{1}{a^{(2i)!! * 2i}}.$$

- ii. Therefore using (4) and (i), verify that  $(e_j^T M(x)A_i(x)e_j)^2 < \frac{1}{a^{(2i)!! * 2i}} \leq (e_j^T M(x)A_i(x)e_j)^2$ .

iii. **Abort procedure.**

(c) Otherwise, do the following:

- i. Let  $k$ , where  $0 \leq k < i$ , be one of the integers for which  $\|A_k(x)e_{k+1}\|^2 < \frac{1}{a^{(2i)!!}}$ .
- ii. Verify that  $\|A_k(x)e_{k+1}\|^2 < \frac{1}{a^{(2i)!!}} \leq \frac{1}{a^{(2k+2)!!}}$ .
- iii. Simultaneously set  $i$  to  $k$  and  $j$  to  $k+1$ .
- iv. Go to (3).

## Procedure 3.72

### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Execute **procedure 3.64** on the matrix  $A$  and let the tuple  $\langle k \rangle$  receive the result. The objective of the following instructions is to either show that  $0 < 0$  or to show that  $\sum_{i=1}^t (m+1-k_i) = m$ .

### Implementation

1. Execute **procedure 3.20** on the matrix  $A$  and let the tuple  $\langle D, u, \rangle$ .
2. Using **procedure 3.65**, verify that  $\sum_{i=1}^t (m+1-k_i)$ 
  - (a)  $= \sum_{i=1}^t \sum_{j=1}^m [k_i \leq j]$
  - (b)  $= \sum_{j=1}^m \sum_{i=1}^t [k_i \leq j]$
  - (c)  $= \sum_{j=1}^m \sum_{i=1}^t [k_i \leq j] \sum_{l=1}^m [k_i = l]$
  - (d)  $= \sum_{j=1}^m \sum_{l=1}^m \sum_{i=1}^t [k_i \leq j][k_i = l]$
  - (e)  $= \sum_{j=1}^m \sum_{l=1}^m \sum_{i=1}^t [l \leq j][k_i = l]$
  - (f)  $= \sum_{j=1}^m \sum_{l=1}^m [l \leq j] \sum_{i=1}^t [k_i = l]$
  - (g)  $= \sum_{j=1}^m \sum_{l=1}^m [l \leq j] \deg u_l$
  - (h)  $= \sum_{j=1}^m \sum_{l=1}^j \deg u_l$

- (i)  $= \sum_{j=1}^m \deg D_{j,j}$
- (j)  $= m$

### Notation 3.29

Let us use the notation  $(2k)!!$  as a shorthand for " $2^k(k!)$ ".

## Procedure 3.73

### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . The objective of the following instructions is to either show that  $0 < 0$  or to construct the rational,  $\text{disc}(A)$ .

### Implementation

1. Execute [procedure 3.63](#) on the matrix  $A$  and let the tuple  $\langle c, d \rangle$  receive the result.
2. Execute [procedure 3.04](#) with  $xI_m - A$  as the choice matrix. Let the tuple  $\langle M, D, , N \rangle$  receive the result.
3. Let  $L = |(\|N^{-1}\|^2)^{(2m+2)!!}|$ .
4. Let  $\text{disc}(A) = \frac{1}{\max(1, L(|c_1|), L(|d_t|))}$ .
5. **Verify that**  $\text{disc}(A) > 0$ .
6. **Yield the tuple**  $\langle \text{disc}(A) \rangle$ .

### Notation 3.30

Let us use the notation  $\text{disc}(A)$  to refer to the result yielded by executing [procedure 3.73](#) on the matrix  $A$ .

## Procedure 3.74

### Objective

Choose integers  $0 < k \leq m$  and a list of  $\mathcal{T}_m(\mathbb{Q}[x])$ ,  $N$ . Let  $Q = (I_m)_{*, [k:m]}$ . The objective of the following instructions is to either show that  $1 = 0$  or to construct an  $\mathcal{M}_{m, m+1-k}(\mathbb{Q}[x])$ ,  $K$ , and an  $\mathcal{M}_{m+1-k, m+1-k}(\mathbb{Q}[x])$ ,  $E$ , such that  $K_i = NQE$  and  $K^T K$  is a  $\mathcal{D}_{m+1-k, m+1-k}(\mathbb{Q}[x])$ .

### Implementation

1. Verify that  $(Q^T N^{-1})(NQ) = Q^T(N^{-1}N)Q = Q^T I_m Q = Q^T Q = I_{m+1-k}$ .
2. Execute [procedure 3.67](#) on the matrices  $Q^T N^{-1}$  and  $NQ$ . Let the tuple  $\langle , \dots , K \rangle$  receive the result.
3. **Verify that**  $K$  is a  $\mathcal{M}_{m, m+1-k}(\mathbb{Q}[x])$ .
4. Using [procedure 3.68](#), verify that  $K^T K$  is a  $\mathcal{D}_{m+1-k, m+1-k}(\mathbb{Q}[x])$ .
5. Let  $E = Q^T N^{-1} K$ .
6. **Verify that**  $E$  is a  $\mathcal{M}_{m+1-k, m+1-k}(\mathbb{Q}[x])$ .
7. Execute [procedure 3.69](#) on the matrices  $Q^T N^{-1}$  and  $NQ$ .
8. **Now verify that**  $K = NQE$ .
9. **Yield**  $\langle K, E \rangle$ .

## Procedure 3.75

### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Choose a  $\mathbb{Q} \epsilon > 0$ . Execute [procedure 3.64](#) on the matrix  $A$  and let the tuple  $\langle k \rangle$  receive the result. The objective of the following instructions is to either show that  $1 < 1$  or to construct  $\mathbb{Q}s$ ,  $0 < \delta \leq 1 \leq K'$ , a list of  $\mathcal{M}_{m,*}(\mathbb{Q})s$ ,  $K$ , and a list of  $\mathbb{Q}s$ ,  $g$ , such that for  $1 \leq i \leq |k|$ :

1.  $\text{cols}(K_i) = m + 1 - k_i$ .
2.  $(K_i)_{p,q} \leq K'm$ , for  $1 \leq p \leq m$ , for  $1 \leq q \leq \text{cols}(K_i)$ .
3.  $K_i^T K_i$  is a  $\mathcal{D}_{*,*}(\mathbb{Q})$ .
4.  $(K_i^T K_i)_{j,j} \geq \text{disc}(A)$  for  $1 \leq j \leq \text{cols}(K_i)$ .
5.  $|(g_i K_i - A K_i)_{p,q}| < \frac{\epsilon \delta}{K'm^2}$ , for  $1 \leq p \leq m$ , for  $1 \leq q \leq \text{cols}(K_i)$ .
6.  $\delta \leq \min_{1 \leq i \neq j \leq |g|} |g_j - g_i|$ .

### Implementation

1. Execute [procedure 3.63](#) on the matrix  $A$  and let the tuple  $\langle c, d \rangle$  receive the result.

2. Execute **procedure 3.20** with  $xI_m - A$  as the choice matrix. Let the tuple  $\langle M, D, u, N \rangle$  receive the result.
  3. Let  $M' = 1 + \max_{i=1}^m \max_{j=1}^m |M^{-1}_{*i,j}| (\max(|c_1|, |d_{|d|}|))$ .
  4. Let  $N' = 1 + \max_{i=1}^m \max_{j=1}^m |N_{*i,j}| (\max(|c_1|, |d_{|d|}|))$ .
  5. Let  $\delta = \min(1, \min_{i=1}^{|d|} (d_{i+1} - c_i))$ .
  6. Execute **procedure 3.74** on  $\langle k, m, N \rangle$  and let the tuple  $\langle \langle K_1, E_1 \rangle, \langle K_2, E_2 \rangle, \dots, \langle K_{|k|}, E_{|k|} \rangle \rangle$  receive.
  7. Using **procedure 3.72**, verify that  $\sum_{p=1}^{|k|} \text{cols}(K_p) = \sum_{p=1}^{|k|} m + 1 - k_p = m$ .
  8. Let  $E' = 1 + \max_{i=1}^t \max_{j=1}^{m+1-k_i} \max_{l=1}^{m+1-k_i} |E_{j,l}| (\max(|c_1|, |d_{|d|}|))$ .
  9. Let  $U = (1 + |u_1|)(1 + |u_2|) \cdots (1 + |u_m|)$ .
  10. Let  $U' = U(\max(|c_1|, |d_{|d|}|))$ .
  11. Let  $b = \frac{\epsilon\delta}{M'N'E'^2m^3}$ .
  12. For  $i = 1$  to  $i = |k|$ , do the following:
    - (a) Verify that  $\text{sgn}(u_{k_i}(c_i)) \neq \text{sgn}(u_{k_i}(d_i))$ .
    - (b) Execute **procedure 2.02** on the formal polynomial  $u_{k_i}$ , interval  $(c_i, d_i)$ , and target of  $\frac{b}{U'}$ . Let  $\langle g_i \rangle$  receive the result.
    - (c) Now verify that  $|u_{k_i}(g_i)| < \frac{b}{U'}$ .
    - (d) Also verify that  $c_i \leq g_i \leq d_i$ .
    - (e) For  $j = k_i$  to  $j = m$ , do the following:
      - i. Verify that  $|D_{j,j}(g_i)| = |u_1(g_i)| |u_2(g_i)| \cdots |u_m(g_i)| \leq |u_{k_i}(g_i)| |u_1(|g_i|)| \cdots |u_{k_i-1}(|g_i|)| \cdot |u_{k_i+1}(|g_i|)| \cdots |u_m(|g_i|)| < \frac{b}{U'} U(|g_i|) = \frac{b}{U'} U' = b$ .
    - (f) Let  $Q = (I_m)_{*, [k_i:m]}$ .
    - (g) If a diagonal entry of  $K_i(g_i)^T K_i(g_i)$  is less than  $\text{disc}(A)$ , then do the following:
      - i. Let  $z$  be the column index of the diagonal entry less than  $\text{disc}(A)$ .
      - ii. Verify that  $\frac{1}{\max(\|(Q^T N^{-1})(g_i)\|^2, 1)^{(2(m+1-k_i)+2)!}}$
  - iii. Execute **procedure 3.71** with matrices  $Q^T N^{-1}$  and  $NQ$ , rational number  $g_i$ , and column index  $z$ .
  - iv. **Abort procedure.**
- (h) Otherwise, do the following:
- i. **For**  $j = 1$  **to**  $j = m + 1 - k_i$ , **verify that**  $(K_i(g_i)^T K_i(g_i))_{j,j} \geq \text{disc}(A) > 0$ .
  - ii. Verify that  $xK_i - AK_i = (xI_m - A)K_i = M^{-1}DN^{-1}K_i = M^{-1}DN^{-1}NQE_i = M^{-1}DQE_i$ .
  - iii. **Verify that**  $(g_i K_i(g_i) - AK_i(g_i))_{p,q} = (M^{-1}(g_i)D(g_i)QE_i(g_i))_{p,q} < M'b(m+1-k_i)E' = M' \frac{\epsilon\delta}{M'N'E'^2m^3} (m+1-k_i)E' \leq \frac{\epsilon\delta}{N'E'm^2}$  **for**  $1 \leq p \leq m$ , **for**  $1 \leq q \leq m+1-k_i$ .
  - iv. **Verify that**  $K_i(g_i)_{p,q} = (N(g_i)QE_i(g_i))_{p,q} = N'(m+1-k_i)E' \leq N'E'm$ .
13. **Yield the tuple**  $\langle \delta, N'E', \langle K_1(g_1), \dots, K_t(g_t) \rangle, g \rangle$ .

## Procedure 3.76

### Objective

Choose a symmetric  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $A$ . Choose a  $\mathbb{Q} \epsilon > 0$ . The objective of the following instructions is to either show that  $1 < 1$  or to construct an  $\mathcal{M}_{m,m}(\mathbb{Q})$ ,  $K$ , and a  $\mathcal{D}_{m,m}(\mathbb{Q})$ ,  $C$ , such that:

1.  $\sum_{p=1}^m \sum_{q=1}^m |(KC - AK)_{p,q}| < \epsilon$ .
2.  $|(K^T K)_{i,j}| \leq 2\epsilon$  for  $1 \leq i \neq j \leq m$ .
3.  $(K^T K)_{j,j} \geq \text{disc}(A) > 0$  for  $1 \leq j \leq m$ .

### Implementation

1. Execute **procedure 3.75** on matrix  $A$  and rational  $\epsilon$ . Let the tuple  $\langle \delta, K', K, g \rangle$  receive the result.
2. Let  $C$  be a diagonal matrix whose  $i^{th}$ , where  $1 \leq i \leq t$ , group of entries are  $m + 1 - k_i$   $g_i$ s.
3. Using **procedure 3.72**, verify that  $C$  is  $m \times m$ .

4. Let  $K$  be a matrix whose columns are the in-order concatenation of those of  $K_1, K_2, \dots, K_t$ .
5. **Using procedure 3.72, verify that  $K$  is  $m \times m$ .**
6. **Using (1), verify that  $\sum_{p=1}^m \sum_{q=1}^m |(KC - AK)_{p,q}| < \sum_{p=1}^m \sum_{q=1}^m \frac{\epsilon\delta}{K'm^2} = \frac{\epsilon\delta}{K'} \leq \epsilon$ .**
7. For  $i = 1$  to  $i = m$ , do the following: For  $j = 1$  to  $j = m$ , do the following:
  - (a) Let  $a, c$  be such that  $Ke_i$  came from  $K_a e_c$ .
  - (b) Let  $b, d$  be such that  $Ke_j$  came from  $K_b e_d$ .
  - (c) If  $a \neq b$ , then do the following:
    - i. Using (1), verify that  $|(g_b - g_a)(Ke_i)^T(Ke_j)|$
    - ii.  $= |g_b(Ke_i)^T(Ke_j) - g_a(Ke_i)^T(Ke_j)|$
    - iii.  $= |(Ke_i)^T(g_b Ke_j) - (g_a Ke_i)^T(Ke_j)|$
    - iv.  $= |(Ke_i)^T(AKe_j + g_b Ke_j - AKe_j) - (AKe_i + g_a Ke_i - AKe_i)^T(Ke_j)|$
    - v.  $\leq |(Ke_i)^T(AKe_j) - (AKe_i)^T(Ke_j)| + |(Ke_i)^T(g_b Ke_j - AKe_j)| + |(g_a Ke_i - AKe_i)^T(Ke_j)|$
    - vi.  $\leq |(Ke_i)^T A(Ke_j) - (Ke_i)^T A^T(Ke_j)| + |mK'J_{1 \times m} \frac{\epsilon\delta}{K'm^2} J_{m \times 1}| + |\frac{\epsilon\delta}{K'm^2} J_{1 \times m} mK'J_{m \times 1}|$
    - vii.  $= 2\epsilon\delta$ .
    - viii. **Therefore using (1) and (vii), verify that  $|e_i^T(K^T K)e_j| = |(Ke_i)^T(Ke_j)| \leq \frac{2\epsilon\delta}{|g_b - g_a|} \leq 2\epsilon$ .**
  - (d) Otherwise if  $c \neq d$ , do the following:
    - i. Using (1), verify that  $K_a^T K_b = K_a^T K_a$  is a  $\mathcal{D}_{*,*}(\mathbb{Q})$ .
    - ii. **Therefore verify that  $(Ke_i)^T(Ke_j) = (K_a e_c)^T(K_b e_d) = e_c^T K_a^T K_b e_d = 0 \leq 2\epsilon$ .**
8. **Therefore using (7), verify that  $|(K^T K)_{i,j}| \leq 2\epsilon$  for  $1 \leq i \neq j \leq m$ .**
9. **Using (1), verify that  $(K^T K)_{j,j} \geq \text{disc}(A) > 0$  for  $1 \leq j \leq m$ .**
10. **Yield the tuple  $\langle K, C \rangle$ .**

## 4 References

- [1] Harold Edwards. *Linear Algebra*. Springer Science+Business Media, 1995.
- [2] Ludwig Wittgenstein. *Philosophical Grammar*. Edited by Rush Rhees. Translated by Anthony Kenny. Basil Blackwell, Oxford, 1974.