

## APPENDIX B

This appendix displays the field function test data and the results of the menu function testing.

### Field Function Test Data

*mod(a)*

| input | output |
|-------|--------|
| 2     | 2      |
| -1    | -1     |
| 0     | 0      |

*maxInt(a,b)*

| input   | output |
|---------|--------|
| (1,2)   | 2      |
| (2,1)   | 2      |
| (1,1)   | 1      |
| (-1,1)  | 1      |
| (-2,-1) | 1      |
| (-1,-2) | 1      |
| (-1,0)  | 0      |
| (0,0)   | 0      |

*modgcd(a,b)*

| input      | output |
|------------|--------|
| (28,49)    | 7      |
| (-9,39)    | 3      |
| (-256,-48) | 16     |
| (11,97)    | 1      |

*longToMod3(a)*

| input | output |
|-------|--------|
| 0     | 0      |
| 1     | 1      |
| 2     | 2      |
| -1    | 2      |
| -2    | 1      |
| 105   | 0      |
| -98   | 1      |
| -58   | 2      |

*rationalise(a)*

| input       | Output        |
|-------------|---------------|
| (1,2)       | (1,2)         |
| (2,4)       | (1,2)         |
| (-2,5)      | (-2,5)        |
| (2,-5)      | (-2,5)        |
| (1,0)       | error message |
| (0,-45)     | (0,1)         |
| (-7,-5)     | (7,5)         |
| (-123,-321) | (-41,107)     |

*stringToRational(a)*

| input   | output        |
|---------|---------------|
| 1/0     | error message |
| -2/-4   | (1,2)         |
| 3/-7    | (-3,7)        |
| -26/169 | (-2,13)       |
| 89/1    | (89/1)        |
| 45/62   | (45,62)       |

*compareInt(a,b)*

| input   | output |
|---------|--------|
| (1,2)   | -1     |
| (2,1)   | 1      |
| (1,1)   | 0      |
| (-1,1)  | -1     |
| (-2,-1) | -1     |
| (-1,-2) | 1      |
| (-1,0)  | -1     |
| (0,0)   | 0      |

*compareMod3(a,d)*

| input    | output |
|----------|--------|
| (1,1)    | 0      |
| (0,1)    | -1     |
| (2,1)    | 1      |
| (11,-5)  | 1      |
| (-3,-1)  | -1     |
| (-1,-3)  | 1      |
| (-17,16) | 0      |

*compareRational(a,b)*

| input           | output |
|-----------------|--------|
| ((1,2),(5,6))   | -1     |
| ((3,7),(4,11))  | 1      |
| ((-3,7),(4,11)) | -1     |
| ((1,2),(2,4))   | 0      |

|                       |    |
|-----------------------|----|
| $((-5,8),(11,-16))$   | 1  |
| $((2,8),(-456,-457))$ | -1 |

*mod3Add(a,b)*

| input    | output |
|----------|--------|
| (1,1)    | 2      |
| (0,1)    | 1      |
| (2,1)    | 0      |
| (13,-5)  | 2      |
| (-1,-4)  | 1      |
| (-12,12) | 0      |

*rationalAdd(a,b)*

| input                     | output        |
|---------------------------|---------------|
| $((1,3),(4,3))$           | (5,3)         |
| $((1,3),(-5,3))$          | (-4,3)        |
| $((1,3),(-4,3))$          | (-1,1)        |
| $((-456,-789),(32,-126))$ | ((568,16569)) |
| $((0,1),(13,2))$          | (13,2)        |

*mod3Sub(a,b)*

| input    | output |
|----------|--------|
| (1,1)    | 0      |
| (0,1)    | 2      |
| (2,1)    | 1      |
| (11,-5)  | 1      |
| (-1,-3)  | 2      |
| (-12,12) | 0      |

*rationalSub(a,b)*

| input                     | output          |
|---------------------------|-----------------|
| $((1,3),(4,3))$           | (-1,1)          |
| $((1,3),(-5,3))$          | (2,1)           |
| $((1,3),(-4,3))$          | (5,3)           |
| $((-456,-789),(32,-126))$ | ((13784,16569)) |
| $((0,1),(13,2))$          | (-13,2)         |

*mod3Mult(a,b)*

| input    | output |
|----------|--------|
| (1,1)    | 1      |
| (0,1)    | 0      |
| (2,1)    | 2      |
| (11,-5)  | 2      |
| (-1,-3)  | 0      |
| (-17,16) | 1      |

*rationalMult(a,b)*

| input                 | output        |
|-----------------------|---------------|
| ((-7,4),(-3,8))       | (21,32)       |
| ((12,8),(32,-6))      | (-8,1)        |
| ((0,3),(-4,3))        | (0,1)         |
| ((-12,-34),(-56,-78)) | ((56,221)     |
| ((1,-65),(65,-1))     | (1,1)         |
| ((23,-69),(1,0))      | error message |

*mod3Div(a,b)*

| input    | output |
|----------|--------|
| (1,0)    | 0      |
| (0,1)    | 0      |
| (1,2)    | 2      |
| (2,1)    | 2      |
| (5,-4)   | 1      |
| (34,-22) | 2      |

*rationalDiv(a,b)*

| input                 | output        |
|-----------------------|---------------|
| ((-7,4),(-3,8))       | (14,3)        |
| ((12,8),(32,-6))      | (-9,32)       |
| ((0,3),(-4,3))        | (0,1)         |
| ((-12,-34),(-56,-78)) | ((117,238)    |
| ((1,-65),(1,-65))     | (1,1)         |
| ((4,5),(0,1))         | error message |

## Menu Function Test Data

Results for function *menu* and *menuFieldChoice*:

| Feature Tested              | Action/Input   | Expected Result             | Actual Result |
|-----------------------------|----------------|-----------------------------|---------------|
| Choice input                | Valid choice   | Correct function called     | PERFECT       |
|                             | Invalid choice | User ask to re enter choice | OK            |
| Display description of task | Void           | Display apt method          | PERFECT       |

Results for functions *menuMatrixConstMult*, *menuMatrixElementWiseMult*, *menuMatrixDirectSum* and *menuSystemSolve*:

| Feature Tested              | Action/Input   | Expected Result             | Actual Result |
|-----------------------------|----------------|-----------------------------|---------------|
| Choice input                | Valid choice   | Correct function called     | PERFECT       |
|                             | Invalid choice | User ask to re enter choice | OK            |
| Display description of task | Void           | Display apt method          | PERFECT       |
| Valid matrix                | Void           | Valid dimension options     | PERFECT       |

|                    |                           |   |         |
|--------------------|---------------------------|---|---------|
| dimensions         |                           | given   |         |
| Matrix calculation | Arbitrary square matrices | Correct result and method, tested up to dim = 4 | PERFECT |
|                    | Diagonal matrices         | Correct result and method, tested up to dim = 4 | PERFECT |
|                    | Lower triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT |
|                    | Upper triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT |
|                    | Rectangular matrices      | Correct result and method, tested up to dim = 4 | PERFECT |

Results for functions *menuMatrixAdd*, *menuMatrixSub* and *menuMatrixMult*:

| Feature Tested              | Action/Input              | Expected Result                                 | Actual Result |
|-----------------------------|---------------------------|---|---------------|
| Choice input                | Valid choice              | Correct function called                         | PERFECT       |
|                             | Invalid choice            | User ask to re enter choice                     | OK            |
| Display description of task | Void                      | Display apt method                              | PERFECT       |
| Valid matrix dimensions     | Void                      | Valid dimension options given                   | PERFECT       |
| Display a complexity review | Asked for review          | Complexity review shows expected method         | PERFECT       |
| Matrix calculation          | Arbitrary square matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Diagonal matrices         | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Lower triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Upper triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Rectangular matrices      | Correct result and method, tested up to dim = 4 | PERFECT       |

Results for functions *menuCalculateInverse* and *menuRowEchelonForm*:

| Feature Tested              | Action/Input              | Expected Result                                 | Actual Result |
|-----------------------------|---------------------------|---|---------------|
| Choice input                | Valid choice              | Correct function called                         | PERFECT       |
|                             | Invalid choice            | User ask to re enter choice                     | OK            |
| Display description of task | Void                      | Display apt method                              | PERFECT       |
| Valid matrix dimensions     | Void                      | Valid dimension options given                   | PERFECT       |
| Matrix calculation          | Arbitrary square matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Diagonal matrices         | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Lower triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Upper triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |

Results for function *menuDet*:

| Feature Tested              | Action/Input              | Expected Result                                 | Actual Result |
|-----------------------------|---------------------------|---|---------------|
| Choice input                | Valid choice              | Correct function called                         | PERFECT       |
|                             | Invalid choice            | User ask to re enter choice                     | OK            |
| Display description of task | Void                      | Display apt method                              | PERFECT       |
| Valid matrix dimensions     | Void                      | Valid dimension options given                   | PERFECT       |
| Display a complexity review | Asked for review          | Complexity review shows expected method         | PERFECT       |
| Matrix calculation          | Arbitrary square matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Diagonal matrices         | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Lower triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |
|                             | Upper triangular matrices | Correct result and method, tested up to dim = 4 | PERFECT       |

Results for function *menuControllability*:

| Feature Tested              | Action/Input              | Expected Result                                 | Actual Result         |
|-----------------------------|---------------------------|---|-----------------------|
| Choice input                | Valid choice              | Correct function called                         | PERFECT               |
|                             | Invalid choice            | User ask to re enter choice                     | OK                    |
| Display description of task | Void                      | Display apt method                              | PERFECT               |
| Valid matrix dimensions     | Void                      | Valid dimension options given                   | PERFECT               |
| Matrix calculation          | Arbitrary square matrices | Correct result and method, tested up to dim = 4 | INVALID FOR DIM A > 2 |
|                             | Diagonal matrices         | Correct result and method, tested up to dim = 4 | INVALID FOR DIM A > 2 |
|                             | Lower triangular matrices | Correct result and method, tested up to dim = 4 | INVALID FOR DIM A > 2 |
|                             | Upper triangular matrices | Correct result and method, tested up to dim = 4 | INVALID FOR DIM A > 2 |
|                             | Rectangular matrices      | Correct result and method, tested up to dim = 4 | INVALID FOR DIM A > 2 |