**%HW3.mat**
**%Patrick Utz, 1/26/18, 3.1**
**%Write Matlab statements to generate the following vectors using array operations.**
**%a = temporary matrix used, ans = final result**

**(1) [$2^3$, $3^3$, $4^3$, $5^3$, $6^3$, $7^3$, $8^3$, $9^3$, $10^3$]**

```
a = 2:10;
ans = a.^3;
```

```
ans =
     8    27    64   125   216   343   512   729  1000
```

**(2) [$3^2$, $3^3$, $3^4$, $3^5$, $3^6$, $3^7$, $3^8$, $3^9$, $3^{10}$]**

```
a = [3 3 3 3 3 3 3 3 3]
ans = a.^(2:10);
```

```
ans =
     9    27    81   243   729  2187  6561 19683 59049
```

**(3) [$1^{10}$, $2^9$, $3^8$, $4^7$, $5^6$, $6^5$, $7^4$, $8^3$, $9^2$, $10^1$]**

```
a = 1:10;
ans = a.^(10:-1:1);
```

```
ans =
     1   512  6561 16384 15625  7776  2401   512    81    10
```

**(4)** $\left[ \frac{1}{1^2 2}, \frac{1}{3^2 4}, \frac{1}{5^2 6}, \ldots \frac{1}{99^2 100} \right]$

format long;
ans = 1./(((1:2:99).^2).*(2:2:100));

ans =

Columns 1 through 5

0.500000000000000 0.027777777777778 0.006666666666667 0.002551020408163 0.001234567901235

Columns 6 through 10

0.000688705234160 0.000422654268808 0.000277777777778 0.000192233756248 0.000138504155125

Columns 11 through 15

0.000103071531643 0.000078764965343 0.000061538461538 0.000048990789732 0.000039635354736

Columns 16 through 20

0.000032518210198 0.000027008048398 0.000022675736961 0.000019222636577 0.000016436554898

Columns 21 through 25

0.000014163904705 0.000012291656424 0.000010735373054 0.000009431115135 0.000008329862557

Columns 26 through 30

0.000007393606010 0.000006592566222 0.000005903187721 0.000005306672610 0.000004787896198

Columns 31 through 35

0.000004334596146 0.000003936759889 0.000003586157432 0.000003275981812 0.000003000570108

Columns 36 through 40

0.000002755185259 0.000002535844157 0.000002339181287 0.000002162339825 0.000002002884153

Columns 41 through 45

0.000001858729150 0.000001728082727 0.000001609398890 0.000001501339195 0.000001402740956

Columns 46 through 50

0.000001312590897 0.000001230003223 0.000001154201293 0.000001084502246 0.000001020304051

**(5) [1, $\frac{-1}{3}$ , $\frac{1}{5}$ , $\frac{-1}{7}$ ,.... $\frac{-1}{999}$ ]**

```
format long;
a = linspace(1,1,500);
a(2:2:500) = (-1);
ans = a./(1:2:999);
```

```
ans =
  Columns 1 through 5
   1.000000000000000  -0.333333333333333   0.200000000000000  -0.142857142857143   0.111111111111111

  Columns 6 through 10
  -0.090909090909091   0.076923076923077  -0.066666666666667   0.058823529411765  -0.052631578947368

  Columns 11 through 15
   0.047619047619048  -0.043478260869565   0.040000000000000  -0.037037037037037   0.034482758620690

  Columns 16 through 20
  -0.032258064516129   0.030303030303030  -0.028571428571429   0.027027027027027  -0.025641025641026

  Columns 21 through 25
   0.024390243902439  -0.023255813953488   0.022222222222222  -0.021276595744681   0.020408163265306

  Columns 26 through 30
  -0.019607843137255   0.018867924528302  -0.018181818181818   0.017543859649123  -0.016949152542373

  Columns 31 through 35
   0.016393442622951  -0.015873015873016   0.015384615384615  -0.014925373134328   0.014492753623188

  Columns 36 through 40
  -0.014084507042254   0.013698630136986  -0.013333333333333   0.012987012987013  -0.012658227848101

  Columns 41 through 45
   0.012345679012346  -0.012048192771084   0.011764705882353  -0.011494252873563   0.011235955056180

  Columns 46 through 50
  -0.010989010989011   0.010752688172043  -0.010526315789474   0.010309278350515  -0.010101010101010

  Columns 51 through 55
   0.009900990099010  -0.009708737864078   0.009523809523810  -0.009345794392523   0.009174311926606

  Columns 56 through 60
  -0.009009009009009   0.008849557522124  -0.008695652173913   0.008547008547009  -0.008403361344538

  Columns 61 through 65
   0.008264462809917  -0.008130081300813   0.008000000000000  -0.007874015748031   0.007751937984496

  Columns 66 through 70
  -0.007633587786260   0.007518796992481  -0.007407407407407   0.007299270072993  -0.007194244604317
```

Columns 71 through 75

0.007092198581560 -0.006993006993007 0.006896551724138 -0.006802721088435 0.006711409395973

Columns 76 through 80

-0.006622516556291 0.006535947712418 -0.006451612903226 0.006369426751592 -0.006289308176101

Columns 81 through 85

0.006211180124224 -0.006134969325153 0.006060606060606 -0.005988023952096 0.005917159763314

Columns 86 through 90

-0.005847953216374 0.005780346820809 -0.005714285714286 0.005649717514124 -0.005586592178771

Columns 91 through 95

0.005524861878453 -0.005464480874317 0.005405405405405 -0.005347593582888 0.005291005291005

Columns 96 through 100

-0.005235602094241 0.005181347150259 -0.005128205128205 0.005076142131980 -0.005025125628141

Columns 101 through 105

0.004975124378109 -0.004926108374384 0.004878048780488 -0.004830917874396 0.004784688995215

Columns 106 through 110

-0.004739336492891 0.004694835680751 -0.004651162790698 0.004608294930876 -0.004566210045662

Columns 111 through 115

0.004524886877828 -0.004484304932735 0.004444444444444 -0.004405286343612 0.004366812227074

Columns 116 through 120

-0.004329004329004 0.004291845493562 -0.004255319148936 0.004219409282700 -0.004184100418410

Columns 121 through 125

0.004149377593361 -0.004115226337449 0.004081632653061 -0.004048582995951 0.004016064257028

Columns 126 through 130

-0.003984063745020 0.003952569169960 -0.003921568627451 0.003891050583658 -0.003861003861004

Columns 131 through 135

0.003831417624521 -0.003802281368821 0.003773584905660 -0.003745318352060 0.003717472118959

Columns 136 through 140

-0.003690036900369 0.003663003663004 -0.003636363636364 0.003610108303249 -0.003584229390681

Columns 141 through 145

0.003558718861210 -0.003533568904594 0.003508771929825 -0.003484320557491 0.003460207612457

Columns 146 through 150

-0.003436426116838 0.003412969283276 -0.003389830508475 0.003367003367003 -0.003344481605351

Columns 151 through 155

0.003322259136213 -0.003300330033003 0.003278688524590 -0.003257328990228 0.003236245954693

Columns 156 through 160
-0.003215434083601 0.003194888178914 -0.003174603174603 0.003154574132492 -0.003134796238245

Columns 161 through 165
0.003115264797508 -0.003095975232198 0.003076923076923 -0.003058103975535 0.003039513677812

Columns 166 through 170
-0.003021148036254 0.003003003003003 -0.002985074626866 0.002967359050445 -0.002949852507375

Columns 171 through 175
0.002932551319648 -0.002915451895044 0.002898550724638 -0.002881844380403 0.002865329512894

Columns 176 through 180
-0.002849002849003 0.002832861189802 -0.002816901408451 0.002801120448179 -0.002785515320334

Columns 181 through 185
0.002770083102493 -0.002754820936639 0.002739726027397 -0.002724795640327 0.002710027100271

Columns 186 through 190
-0.002695417789757 0.002680965147453 -0.002666666666667 0.002652519893899 -0.002638522427441

Columns 191 through 195
0.002624671916010 -0.002610966057441 0.002597402597403 -0.002583979328165 0.002570694087404

Columns 196 through 200
-0.002557544757033 0.002544529262087 -0.002531645569620 0.002518891687657 -0.002506265664160

Columns 201 through 205
0.002493765586035 -0.002481389578164 0.002469135802469 -0.002457002457002 0.002444987775061

Columns 206 through 210
-0.002433090024331 0.002421307506053 -0.002409638554217 0.002398081534772 -0.002386634844869

Columns 211 through 215
0.002375296912114 -0.002364066193853 0.002352941176471 -0.002341920374707 0.002331002331002

Columns 216 through 220
-0.002320185614849 0.002309468822171 -0.002298850574713 0.002288329519451 -0.002277904328018

Columns 221 through 225
0.002267573696145 -0.002257336343115 0.002247191011236 -0.002237136465324 0.002227171492205

Columns 226 through 230
-0.002217294900222 0.002207505518764 -0.002197802197802 0.002188183807440 -0.002178649237473

Columns 231 through 235
0.002169197396963 -0.002159827213823 0.002150537634409 -0.002141327623126 0.002132196162047

Columns 236 through 240
-0.002123142250531  0.002114164904863  -0.002105263157895  0.002096436058700  -0.002087682672234

Columns 241 through 245
 0.002079002079002  -0.002070393374741  0.002061855670103  -0.002053388090349  0.002044989775051

Columns 246 through 250
-0.002036659877800  0.002028397565923  -0.002020202020202  0.002012072434608  -0.002004008016032

Columns 251 through 255
 0.001996007984032  -0.001988071570577  0.001980198019802  -0.001972386587771  0.001964636542240

Columns 256 through 260
-0.001956947162427  0.001949317738791  -0.001941747572816  0.001934235976789  -0.001926782273603

Columns 261 through 265
 0.001919385796545  -0.001912045889101  0.001904761904762  -0.001897533206831  0.001890359168242

Columns 266 through 270
-0.001883239171375  0.001876172607880  -0.001869158878505  0.001862197392924  -0.001855287569573

Columns 271 through 275
 0.001848428835490  -0.001841620626151  0.001834862385321  -0.001828153564899  0.001821493624772

Columns 276 through 280
-0.001814882032668  0.001808318264014  -0.001801801801802  0.001795332136445  -0.001788908765653

Columns 281 through 285
 0.001782531194296  -0.001776198934281  0.001769911504425  -0.001763668430335  0.001757469244288

Columns 286 through 290
-0.001751313485114  0.001745200698080  -0.001739130434783  0.001733102253033  -0.001727115716753

Columns 291 through 295
 0.001721170395869  -0.001715265866209  0.001709401709402  -0.001703577512777  0.001697792869270

Columns 296 through 300
-0.001692047377327  0.001686340640809  -0.001680672268908  0.001675041876047  -0.001669449081803

Columns 301 through 305
 0.001663893510815  -0.001658374792703  0.001652892561983  -0.001647446457990  0.001642036124795

Columns 306 through 310
-0.001636661211129  0.001631321370310  -0.001626016260163  0.001620745542950  -0.001615508885299

Columns 311 through 315
 0.001610305958132  -0.001605136436597  0.001600000000000  -0.001594896331738  0.001589825119237

Columns 316 through 320
-0.001584786053883  0.001579778830964  -0.001574803149606  0.001569858712716  -0.001564945226917

Columns 321 through 325
 0.001560062402496  -0.001555209953344   0.001550387596899  -0.001545595054096   0.001540832049307

Columns 326 through 330
-0.001536098310292   0.001531393568147  -0.001526717557252   0.001522070015221  -0.001517450682853

Columns 331 through 335
 0.001512859304085  -0.001508295625943   0.001503759398496  -0.001499250374813   0.001494768310912

Columns 336 through 340
-0.001490312965723   0.001485884101040  -0.001481481481481   0.001477104874446  -0.001472754050074

Columns 341 through 345
 0.001468428781204  -0.001464128843338   0.001459854014599  -0.001455604075691   0.001451378809869

Columns 346 through 350
-0.001447178002894   0.001443001443001  -0.001438848920863   0.001434720229555  -0.001430615164521

Columns 351 through 355
 0.001426533523538  -0.001422475106686   0.001418439716312  -0.001414427157001   0.001410437235543

Columns 356 through 360
-0.001406469760900   0.001402524544180  -0.001398601398601   0.001394700139470  -0.001390820584145

Columns 361 through 365
 0.001386962552011  -0.001383125864454   0.001379310344828  -0.001375515818432   0.001371742112483

Columns 366 through 370
-0.001367989056088   0.001364256480218  -0.001360544217687   0.001356852103121  -0.001353179972936

Columns 371 through 375
 0.001349527665317  -0.001345895020188   0.001342281879195  -0.001338688085676   0.001335113484646

Columns 376 through 380
-0.001331557922770   0.001328021248340  -0.001324503311258   0.001321003963012  -0.001317523056653

Columns 381 through 385
 0.001314060446781  -0.001310615989515   0.001307189542484  -0.001303780964798   0.001300390117035

Columns 386 through 390
-0.001297016861219   0.001293661060802  -0.001290322580645   0.001287001287001  -0.001283697047497

Columns 391 through 395
 0.001280409731114  -0.001277139208174   0.001273885350318  -0.001270648030496   0.001267427122940

Columns 396 through 400
-0.001264222503161   0.001261034047919  -0.001257861635220   0.001254705144291  -0.001251564455569

Columns 401 through 405

0.001248439450687  -0.001245330012453  0.001242236024845  -0.001239157372986  0.001236093943140

Columns 406 through 410
-0.001233045622688  0.001230012300123  -0.001226993865031  0.001223990208078  -0.001221001221001

Columns 411 through 415
0.001218026796590  -0.001215066828676  0.001212121212121  -0.001209189842805  0.001206272617612

Columns 416 through 420
-0.001203369434416  0.001200480192077  -0.001197604790419  0.001194743130227  -0.001191895113230

Columns 421 through 425
0.001189060642093  -0.001186239620403  0.001183431952663  -0.001180637544274  0.001177856301531

Columns 426 through 430
-0.001175088131610  0.001172332942556  -0.001169590643275  0.001166861143524  -0.001164144353900

Columns 431 through 435
0.001161440185830  -0.001158748551564  0.001156069364162  -0.001153402537486  0.001150747986191

Columns 436 through 440
-0.001148105625718  0.001145475372279  -0.001142857142857  0.001140250855188  -0.001137656427759

Columns 441 through 445
0.001135073779796  -0.001132502831257  0.001129943502825  -0.001127395715896  0.001124859392576

Columns 446 through 450
-0.001122334455668  0.001119820828667  -0.001117318435754  0.001114827201784  -0.001112347052280

Columns 451 through 455
0.001109877913430  -0.001107419712071  0.001104972375691  -0.001102535832415  0.001100110011001

Columns 456 through 460
-0.001097694840834  0.001095290251917  -0.001092896174863  0.001090512540894  -0.001088139281828

Columns 461 through 465
0.001085776330076  -0.001083423618635  0.001081081081081  -0.001078748651564  0.001076426264801

Columns 466 through 470
-0.001074113856069  0.001071811361200  -0.001069518716578  0.001067235859125  -0.001064962726305

Columns 471 through 475
0.001062699256111  -0.001060445387063  0.001058201058201  -0.001055966209081  0.001053740779768

Columns 476 through 480
-0.001051524710831  0.001049317943337  -0.001047120418848  0.001044932079415  -0.001042752867570

Columns 481 through 485
0.001040582726327  -0.001038421599169  0.001036269430052  -0.001034126163392  0.001031991744066

Columns 486 through 490

-0.001029866117405  0.001027749229188  -0.001025641025641  0.001023541453429  -0.001021450459653

Columns 491 through 495

0.001019367991845  -0.001017293997965  0.001015228426396  -0.001013171225937  0.001011122345804

Columns 496 through 500

-0.001009081735621  0.001007049345418  -0.001005025125628  0.001003009027081  -0.001001001001001

**%HW3.mat**
**%Patrick Utz, 1/26/18, 3.2**
**%Write Matlab statements that use the built-in function sum to calculate the following summations, each for the first n terms, Your statements should be general for any positive integer values of n. Show your results for n=10.**
**%n = the number of terms used, a = temporary matrix, ans = final answer**

**(1) 3+5+7+9+11+...**

```
n = 10;
a = 3:2:((2*n)+2);
ans = sum(a);
```

```
ans =
   120
```

**(2) 1 + $\frac{1}{2}$ + $\frac{1}{3}$ + $\frac{1}{4}$ + $\frac{1}{5}$ + …**

```
n = 10;
a = 1./(1:n);
ans = sum(a);
```

```
ans =
   2.928968253968254
```

**(3) 1 + $\frac{1}{2}$ + $\frac{1}{4}$ + $\frac{1}{8}$ + $\frac{1}{16}$ + …**

```
n = 10;
a = 1./(2.^(0:(n-1)));
ans = sum(a);
```

```
ans =
   1.998046875000000
```

**(4) $\frac{3}{1}$ + $\frac{5}{2}$ + $\frac{7}{3}$ + $\frac{9}{4}$ + …**

```
n = 10;
a = (3:2:((2*n)+2))./(1:n);
ans = sum(a);
```

```
ans =
  22.928968253968257
```

**%HW3.mat**
**%Patrick Utz, 1/26/18, 3.3**
**%Generate a vector of 20 random integers, each in the range from 50 to 100. Create a variable evens that stores all of the even numbers from the vector, and a variable odds that stores the odd numbers.**
**%a = random integer vector, oddsReference = vector that has a 1 in place of the odds and a 0 for the evens for multiplying with a to get the odd numbers, evensReference = opposite of oddsReference and used to multiply by a to get the even numbers, oddsTemp = has all the odd numbers and 0s, evensTemp = has all the even numbers and 0s, evens = vector with only the even numbers, odds = vector with only the odd numbers**

```
a = randi([50,100],1,20);
oddsReference = rem((a(1:end)),2);
evensReference = not(oddsReference);
oddsReference = not(evensReference);
oddsTemp = a.*oddsReference;
evensTemp = a.*evensReference;
odds = oddsTemp(oddsTemp ~= 0);
evens = evensTemp(evensTemp ~= 0);
odds
evens
```

Test.m

```
1 -    a = randi([50,100],1,20)
2 -    oddsReference = rem((a(1:end)),2)
3 -    logicalOdds = logical(oddsReference)
4 -    odds = a(logicalOdds)
5 -    logicalEvens = ~logicalOdds
6 -    evens = a(logicalEvens)
```

```
odds =
   83    51    93    97    87    83    51    91    85    51


evens =
   84    88    70    58    86    64    52    54    66    98

a =

  Columns 1 through 19
   83    51    93    97    84    88    87    70    83    58    86    51    64    52    54    91    85    66    98

  Column 20
   51
```

**%HW3.mat**
**%Patrick Utz, 1/26/18, 3.4**
**%Create a vector variable vec; it can have any length. Then, write assignment statements that would store the first half of the vector in one variable and the second half in another. Make sure that your assignment statements are general, and work whether vec has an even or odd number of elements. If the length of the vector is odd, the element in the middle should be the last element if the first half vector, as well as the first element in the second half vector. Show your results for two different vectors, one with even length, and the other with odd length.**
**%vec = vector being analyzed, testForOdd = returns a logical 1 if the vector is odd and a logical 0 if the vector is even, firstHalf = vector containing the first half of vector vec (if even) and the first half along with the middle element (if odd), secondHalf = vector containing the second half of vector vec (if even) and the second half along with the middle element (if odd)**

```
vec = 1:11;
vec
testForOdd = logical(rem((numel(vec)),2));
testForOdd
firstHalf = vec( 1:( (( (numel(vec)) + testForOdd )/2) ) );
firstHalf
secondHalf = vec( ( (( (numel(vec)) + testForOdd )/2) + not(testForOdd):end) );
secondHalf
```

```
vec =
    1    2    3    4    5    6    7    8    9   10   11


testForOdd =
  logical
   1


firstHalf =
    1    2    3    4    5    6


secondHalf =
    6    7    8    9   10   11
```

```
vec = 1:10;
vec
testForOdd = logical(rem((numel(vec)),2));
testForOdd
firstHalf = vec( 1:( (( (numel(vec)) + testForOdd )/2) ) );
firstHalf
secondHalf = vec( ( (( (numel(vec)) + testForOdd )/2) + not(testForOdd):end) );
secondHalf
```

```
vec =
    1     2     3     4     5     6     7     8     9    10


testForOdd =
  logical
   0


firstHalf =
    1     2     3     4     5


secondHalf =
    6     7     8     9    10
```

**%HW3.mat**
**%Patrick Utz, 1/26/18, 3.5**
**%Write MATLAB commands to :**
**1. Create two 4x4 matrices A and B, each has random integer elements whose values are uniformly distributed between -8 and 8.**
**2. Partition each of them into four 2x2 blocks as shown below.**

$$A=\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}; \quad B=\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

**3. Show that matrix addition A+B, matrix subtraction A-B, and scalar multiplication A*5, can be performed block-by-block, and concatenated for the overall result.**
**4. Show that matrix multiplication A*B can also be calculated block-by-block.**

$$A*B=\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

**% A = initial random vector 1, B = initial random vector 2, A(1-4) and B(1-4) = the respective parts of A and B in blocks, sum = sum of A and B, sumCon = sum of A and B by block, the other operands follow the same convention with the term "Con" denoting the block by block method,**

1).

```
A = randi([-8,8],4,4)
B = randi([-8,8],4,4)
```

```
A =
     8     2     0     7
     1     2     0    -1
     6     8    -7     5
    -4    -7     7    -6

B =
     2    -5     5     3
    -4    -3     8    -2
```

```
    -1     0    -6     8
     6    -3    -4     8
```

2).

```
A1 = A(1:2,1:2);
A2 = A(3:4,1:2);
A3 = A(1:2,3:4);
A4 = A(3:4,3:4);
B1 = B(1:2,1:2);
B2 = B(3:4,1:2);
B3 = B(1:2,3:4);
B4 = B(3:4,3:4);
```

```
A1 =
     8     2
     1     2
A2 =
     6     8
    -4    -7
A3 =
     0     7
     0    -1
A4 =
    -7     5
     7    -6
B1 =
     2    -5
    -4    -3
B2 =
    -1     0
     6    -3
B3 =
     5     3
     8    -2
B4 =
    -6     8
    -4     8
```

3).

```
sum = A + B
sum1 = (A1+B1);
sum2 = (A2+B2);
sum3 = (A3+B3);
sum4 = (A4+B4);
sumCon = zeros(4,4);
sumCon(1:2,1:2) = sum1;
sumCon(3:4,1:2) = sum2;
sumCon(1:2,3:4) = sum3;
sumCon(3:4,3:4) = sum4;
sumCon
```

```
sum =
    10    -3     5    10
    -3    -1     8    -3
     5     8   -13    13
     2   -10     3     2

sumCon =
    10    -3     5    10
    -3    -1     8    -3
     5     8   -13    13
     2   -10     3     2
```

```
subt = A - B
subt1 = (A1-B1);
subt2 = (A2-B2);
subt3 = (A3-B3);
subt4 = (A4-B4);
subtCon = zeros(4,4);
subtCon(1:2,1:2) = subt1;
subtCon(3:4,1:2) = subt2;
subtCon(1:2,3:4) = subt3;
subtCon(3:4,3:4) = subt4;
subtCon
```

```
subt =
     6     7    -5     4
     5     5    -8     1
     7     8    -1    -3
   -10    -4    11   -14

subtCon =
     6     7    -5     4
     5     5    -8     1
     7     8    -1    -3
   -10    -4    11   -14
```

```
scal = A*5
scal1 = (A1*5);
scal2 = (A2*5);
scal3 = (A3*5);
scal4 = (A4*5);
scalCon = zeros(4,4);
scalCon(1:2,1:2) = scal1;
scalCon(3:4,1:2) = scal2;
scalCon(1:2,3:4) = scal3;
scalCon(3:4,3:4) = scal4;
 scalCon
```

```
scal =
    40    10     0    35
     5    10     0    -5
    30    40   -35    25
   -20   -35    35   -30

scalCon =
    40    10     0    35
     5    10     0    -5
```

```
   30    40   -35    25
  -20   -35    35   -30
```

4).

```
mult = A*B
part1A = zeros(2,4);
part1A(1:2,1:2) = A1;
part1A(1:2,3:4) = A3;
part1B = zeros(4,2);
part1B(1:2,1:2) = B1;
part1B(3:4,1:2) = B2;
part1mult = part1A*part1B;

part2A = zeros(2,4);
part2A(3:4,1:2) = A2;
part2A(3:4,3:4) = A4;
part2B = zeros(4,2);
part2B(1:2,1:2) = B1;
part2B(3:4,1:2) = B2;
part2mult = part2A*part2B;

part3A = zeros(2,4);
part3A(1:2,1:2) = A1;
part3A(1:2,3:4) = A3;
part3B = zeros(4,2);
part3B(1:2,3:4) = B3;
part3B(3:4,3:4) = B4;
part3mult = part3A*part3B;

part4A = zeros(2,4);
part4A(3:4,1:2) = A2;
part4A(3:4,3:4) = A4;
part4B = zeros(4,2);
part4B(1:2,3:4) = B3;
part4B(3:4,3:4) = B4;
part4mult = part4A*part4B;

multCon = part2mult;
multCon(1:2,1:2) = part1mult;
```

```
multCon(1:2,3:4) = part3mult(1:2,3:4);
multCon(3:4,3:4) = part4mult(3:4,3:4);
multCon

mult =
    50   -67    28    76
   -12    -8    25    -9
    17   -69   116   -14
   -23    59   -94    10

multCon =
    50   -67    28    76
   -12    -8    25    -9
    17   -69   116   -14
   -23    59   -94    10
```