ID (e.g., ab1234): _____ Name: _____ Expected Course Grade: _____

BIOMED 201 - Programming & Modeling for BME

Midterm Exam, 2014.05.06, Instructor: Ahmet Sacan

Sign the honor code below. **No credit will be given for the exam without a signed pledge.**

I have neither given nor received aid on this examination.

Signed: _____

There are 7 questions in this exam. Turn in your exam paper before you start working on Question 7. Submit your code for Question 7 on ProgrammingBank. Sign the sign-out sheet before leaving the room.

Q1 (10 pts). *Indexing.* Let **A** be a row vector with **C** columns. Write a single statement that will remove from **A** all even-ordered elements. For example, if **A** is [3 7 2 9 8 10], after your statement, **A** should become [3 2 8]. Do not use loops.

Q2 (10 pts). *Creating vectors.* Fill in the blanks below with what Matlab would display for the given expression.

```
>> [ 3:3:3 7:3:3 8:-3:3 reshape(repmat([1 2],2,2),1,[]) ]
ans =


_____
```

Q3 (10 pts). *Variable scope.* Fill in the blanks in the output.

```
>> a = 'apple';
>> b = apple( a );
>> apple( 'orange' );
>> [a b]
ans =


_____
```

```
function b = apple( a )
b=char(a+2);
a=a+2;
```

Q4 (10 pts). *Indexing.* Let **M** be a matrix with **1000** rows and **10** columns. **M(2468)** can equivalently be expressed as **M(x,y)**. What are the values of **x** and **y**?

x=_____, y=_____

ID (e.g., ab1234): _____ Name: _____ Expected Course Grade: _____

Q5 (10 pts). *Vectorizing code, element-wise operations, logical indexing, loops.* Let **M** be a matrix with 2 rows. Write code that removes from **M** any columns where the element in the first row is greater than that in the second row and also remove from **M** any columns where the element in second row is a square of that in the first row. E.g., if **M** is [3 2 5 7 1; 1 6 25 8 1], after your code, M should contain [2 7; 6 8].

Q6 (10 pts). *Nested for loops.* Fill in the blanks in the output below.

```
>> m = [5 7 9; 3 4 8; 2 6 1];   x = [];
>> for a = 1:size(m,1);  for b = 1:a
>>    x(end+1) = m(a,b);
>>  end; end
>> disp ( x )




_____________________________________________________
```

PROGRAMMING QUESTION – Q7 (40 pts). *Loops or vectorized code.* Consider an experiment where ants are placed on a rectangular board of height **R** inches and width **C** inches and are allowed to move on the board. Let us represent the number of ants on one inch-square segments of the board by the matrix **m**. The figure below shows an example of a 3-by-4 board. After one minute, for any given segment, 1/4th of the ants on that segment move up, 1/4th move down, 1/4th move left, and 1/4th move right. This means that for the ants on the segments along the edges of the board, some of the ants move off the board.

8	0	0	4
0	0	10	0
0	0	0	0

0	2	3.5	0
2	2.5	0	3.5
0	0	2.5	0

Write a function **simulateants2d(m,T)** that takes a matrix **m** and returns the number of ants on the segments of the board after **T** minutes. For simplicity, assume that the number of ants in a segment does not have to be a whole integer. Assume the time length **T** is an integer. If **T** is not given, use **T**=1.

```
>> disp( simulateants2d( [8 0 0 4; 0 0 10 0; 0 0 0 0] ) )
         0    2.0000    3.5000         0
    2.0000    2.5000         0    3.5000
         0         0    2.5000         0

>> disp( simulateants2d( [8 0 0 4; 0 0 10 0; 0 0 0 0], 2 ) )
    1.0000    1.5000    0.5000    1.7500
    0.6250    1.0000    3.0000         0
    0.5000    1.2500         0    1.5000
```