

190

Name: Sample

NOTE: The midterm is open textbook, open notes, and open lab. Completely answer all questions. Presentation is as important as the correctness of your results. Superfluous writing will count against you. Good luck.

The code for the problems in the exam are available on the course webpage. Write your answers in a **single file named exam.doc** and upload it using handin.

1. Study the following piece of code: a)

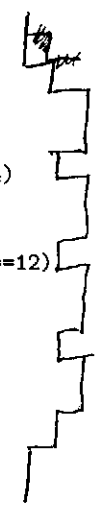
```

Line 1: a=5;
Line 2: for k=1:10
Line 3: if(k==3 | k<2)
Line 4: kk=k+2;
Line 5: disp(kk)
- Line 6: elseif (k<=6 & k>4)
Line 7: a=k*2;
Line 8: disp(a)
Line 9: elseif (k==10 | k==12)
Line 10: a=k;
Line 11: disp(a)
Line 12: elseif (k>9)
Line 13: a=a-1;
Line 14: disp(a)
Line 15: end
Line 16: end
Line 17: disp('done')
```

c)

```

k=1
while k <= 10
:
:
:
:
:
:
:
:
:
k=k+1
```



- 5 (a) Rewrite the code with proper indentation and documentation, which shows good programming practice.
- 5 (b) Explain what *Line 6* does. *check for 5, 6*
- 5 (c) Modify *Line 2* with a **while** loop. Clearly indicate what changes are needed.
- 5 (d) What is the output of the fragment of code and **very** briefly explain the intuition. *3 5 10 12 10 displays of kk or a depending on if elseif as k goes from 1 to 10*

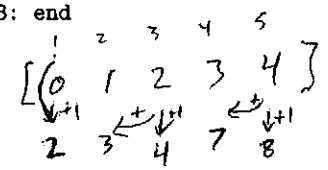
2. Study the following piece of code:

```

Line 1: v=[0 1 2 3 4];
Line 2: for k = 1:5
Line 3: switch (k-1)
Line 4: case {1,3}
- Line 5: v(k)=v(k)+v(k+1);
Line 6: case {2,4}
Line 7: v(k)=new+1;
Line 8: otherwise
Line 9: v(k)=k+1;
Line 10: end
Line 12: new=v(k)
Line 13: end
```

```

v = [0 1 2 3 4]
for k = 1:5
  switch (k-1)
    case {1,3}
      v(k) = v(k) + v(k+1)
    case {2,4}
      v(k) = new + 1
    otherwise
      v(k) = k + 1
  end
  new = v(k)
end
```



- 15
- 5 (a) Explain what *Line 5* does. *adds next k value to current one*
 5 (b) Modify *Line 3* with an if structure. Clearly indicate what changes are needed.
 5 (c) What is the output of the fragment of code and very briefly explain the intuition.

3. Study the following piece of code:

```
Line 1: a = [12 17 -45 53 47];
Line 2: N=5;
Line 3: for m=1:N
Line 4: if ( m==1 )
Line 5: disp(a(m));
Line 6: elseif ( (m==3) | (m>=4) )
Line 7: disp( a(m-1) );
Line 8: else
Line 9: disp( a(3) );
Line 10: end
Line 11: end
```

a = [12 17 -45 53 47];
N = 5
for m = 1:N
if (m==1)
disp(a(m));
elseif (m >= 3)
disp(a(m-1))
else disp(a(3))
end
end

1 → 12
3, 4, 5... → 17
-45
53
2 → -45

- 45
- 15 (a) What does the semicolon at the end of *Line 1* do?
 5 (b) Explain what *Line 4* does.
 5 (c) What is the output of the fragment of code and very briefly explain the intuition. (not shown)

4. What is the output of the following fragment of code and very briefly explain the intuition.

5

```
format long;
x=1/3;
sum=0;
for ii=1:30000
    sum=sum+x;
end
disp(sum);
```

1.0000... (31) + 004

round-off error
1/3 can not be
absolutely represented
in binary sci-notation

5. Study the following piece of code:

```
count = 0;
for d = 1:7
    for h = 1:24
        for m = 1:60
            for s = 1:60
                count = count + 1;
            end
        end
    end
end
disp(count);
```

number of seconds in a week

- 10 (a) What is the output of the fragment of code and very briefly explain the intuition.
 5 (b) What is a simpler way to achieve the same results?

simple multiplication

6. Study the following piece of code:

```
x=-50000;
for a=-1:1:x
    disp('Hello World');
end
```

5 (a) How many times will this program print "Hello World"?

1. 50000

2. 1

3. 0

4. Infinite

5. None of the above

$$500 * 100 = 50000$$

5 (b) What is the time complexity of the above program?

Hint: Complexity is related to the unrestricted growth of x and the execution time of the algorithm

linear

7. Study the following piece of code:

```
x=5000;
y=10;
for a=10:10:x
    for b=0.1:0.1:y
        disp('Hello World');
    end
end
```

5 (a) How many times will this program print "Hello World"?

1. 50000

2. 1

3. 0

4. Infinite

5. None of the above

5 (b) What is the time complexity of the above program?

Hint: Express it in terms of x and y .

$$O(xy)$$

8. What is the time complexity of the following program?

Hint: Express it in terms of x .

```
x=5000;
for a=1:x^2
    for b=5:x^2
        disp('Hello World');
    end
end
```

$$O(x^4)$$

9. Evaluate the following relational and logical expressions as MATLAB does and very briefly explain the results. The function *fun* is defined as follows:

```
function y=fun(x)
disp('This is a fun exam');
y=x;
```

40
2 each

1. $4 < 5 \ \& \ 6 > 7$	0	
2. $5 \neq 5 \ \ 1$	1	
3. $\neg(1 \ \& \ 1 \ \ 0)$	0	
4. $4 \ \& \ 3 < 2$	0	
5. $4 > 2 \ \ fun(5)$	1	sc
6. $4 > 2 \ \ fun(0)$	1	sc
7. $4 < 2 \ \ fun(5)$	1	msg
8. $4 < 2 \ \ fun(0)$	0	msg
9. $4 > 2 \ \&\& \ fun(5)$	1	msg
10. $4 > 2 \ \&\& \ fun(0)$	0	msg
11. $4 < 2 \ \&\& \ fun(5)$	0	sc
12. $4 < 2 \ \&\& \ fun(0)$	0	sc
13. $4 > 2 \ \ fun(5)$	1	msg
14. $4 > 2 \ \ fun(0)$	1	msg
15. $4 < 2 \ \ fun(5)$	1	msg
16. $4 < 2 \ \ fun(0)$	0	msg
17. $4 > 2 \ \& \ fun(5)$	1	msg
18. $4 > 2 \ \& \ fun(0)$	0	msg
19. $4 < 2 \ \& \ fun(5)$	0	msg
20. $4 < 2 \ \& \ fun(0)$	0	msg

potential short circuit
r.h.s. never is executed

10. The following data is stored in variable **array**:

```

2 3 4 7
1 1 0 9
4 2 3 1
    
```

Give a MATLAB command that will set the following variables equal to the indicated portion of **array**.

30
 / (a) X = 2
 1
 4

$X = \text{array}(:, 1)$

/ (b) Y = 3
 1
 2

$Y = \text{array}(:, 2)$

/ (c) Z = 2 3
 1 1

$Z = \text{array}([1, 2], [1, 2])$

/ (d) A = 2 7
 4 1

$A = \text{array}([1, 3], [1, 4])$

/ (e) B = 4 2 3 1

$B = \text{array}(3, :)$

/ (f) C = 1 4 3 1

$C = \text{array}(2:5)$

11. What is the difference between the **assignment (=)** and **comparison (==)** operators? Give examples to justify your answer.

30
 / / 1) **assignment** sets a variable equal to an expression etc. $a = 5$
 if $a == 5$

12. What is the difference between the ***** and **.*** operators? Give examples to justify your answer.

matrix \rightarrow \leftarrow array or scalar

13. What is the difference between **&&** and **&** operators? Give examples to justify your answer.

short circuit \rightarrow \leftarrow always do both sides

> not shown here