**Home Exam**

**Question 1: Select True or False:**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
| **a)** | >> x = -5; y = 2;  
>> (y>3) & (x<1)  
ans =  
0 | **T** |
| **b)** | >> y = (7>8) + (5*3 = 60/4) + (6<10)  
y =  
2 | **T** |
| **c)** | >> 3 + (4<8)/2  
an s =  
3.5 | **T** |
| **d)** | A case statement can have more than one value | **T** |
| **e)** | Indefinite loop is a while-end loop with a conditional expression that is  
always true | |
| **f)** | >> A = [1 3 9 -2 0 7 2];  
>> find(A>3)  
an s =  
3 7 | **F** |
| **g)** | In the following nested loop, the 'disp' statement will be executed 25 times.  
for k = 1: 5  
for m = 1: 4  
disp ('my name')  
end  
end | **F** |
| **h)** | Nested loops may have the same 'end' statement | **F** |
| **i)** | In for-end loops, the value of the counter can be redefined within the loop | **F** |
| **j)** | Rational and logical operators may be combined in one conditional statement | **T** |
| **k)** | When inside a loop, the break command terminates the execution of the file | **F** |
| **l)** | A script file must use a definition line | **F** |
Question 2: Fill the blanks:

a) In a loop, the commands are repeated several times. Every execution or round is called pass.

b) When the loop ends, the loop index variable (k) has the value that was last assigned to it.

c) The looping process continues until a stated condition is false.

d) The differences between Logical vectors and numerical vectors are:
   1. Numerical vectors can’t be used for addressing.
   2. Logical vectors can be used in arithmetic operations and then it changed to numerical vectors or array

e) Logical operators (like relational operators) are used in:
   1. As arithmetic operators within a mathematical expression.
   2. In addressing arrays.
   3. To control the flow of the program.

f) The switch expression is a variable that is:
   1. a scalar,
   2. a string,
   3. a mathematical expression that includes preassigned variables and can be evaluated.

g) List three of MATLAB logical built-in functions and use them in the command window

```
>> xor(7,0)
>> A=[6 2 15 7]; all(A)
>> any(A)
```

Question 3: Find the mistake and make a correction:

<table>
<thead>
<tr>
<th></th>
<th>If age &gt;45</th>
<th>for k = 5 &gt; 8:40 -&gt; 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>.... Group 1 of commands</td>
<td>.... group of commands</td>
</tr>
<tr>
<td></td>
<td>else</td>
<td>end</td>
</tr>
<tr>
<td>b)</td>
<td>x = input('Enter x = ')</td>
<td>end</td>
</tr>
<tr>
<td></td>
<td>n=1;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>s=1;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>while n &lt;= 30</td>
<td></td>
</tr>
<tr>
<td></td>
<td>s = s + x^n / factorial (n);</td>
<td></td>
</tr>
<tr>
<td></td>
<td>n = n + 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>end</td>
<td></td>
</tr>
<tr>
<td>c)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d)</td>
<td>switch bloodpress</td>
<td></td>
</tr>
<tr>
<td></td>
<td>case 120</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.... Group 1 of commands</td>
<td></td>
</tr>
<tr>
<td></td>
<td>case 200</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.... Group 2 of commands</td>
<td></td>
</tr>
<tr>
<td></td>
<td>else</td>
<td></td>
</tr>
<tr>
<td></td>
<td>otherwise</td>
<td></td>
</tr>
<tr>
<td></td>
<td>....... Group 3 of commands</td>
<td></td>
</tr>
<tr>
<td></td>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>
Question 4: Write the output of the following for-end loop:
\[ n=1; \]
\[ \text{for } k = 25:5:10 \]
\[ \quad x(n) = n * k; \]
\[ \quad n = n + 1; \]
\[ \text{end} \]

\[ >> x= \begin{array}{cccc}
25 & 40 & 45 & 40 
\end{array} \]

Question 5: Write the matrix A that results by executing the following program:
\[ \text{for } k = 1:4 \]
\[ \quad \text{for } m = 1:4 \]
\[ \quad \quad \text{if } k == 1 \]
\[ \quad \quad \quad A(k,m) = m; \]
\[ \quad \quad \text{elseif } m == 1 \]
\[ \quad \quad \quad A(k,m) = k; \]
\[ \quad \quad \text{else} \]
\[ \quad \quad \quad A(k,m) = A(k-1,m) - A(k,m-1); \]
\[ \quad \text{end} \]
\[ \text{end} \]

\[ >> A= \]
\[
\begin{array}{cccc}
1 & 2 & 3 & 4 \\
2 & 0 & -3 & -7 \\
3 & 3 & 6 & 13 \\
4 & 1 & -5 & -18 \\
\end{array}
\]

انظري طريقة الحل في الصفحة التالية: ...
،، هنا نتوقع ظهور مصفوفة من 4 سطور و 5 اعمدة kxm matrix تعطينا Nested loop كنا نعلم أن لكل سطر n قبل أن نصل للسطر التالي في حالتنا سوف n times loop الأولى تعتبر عن الأسلوب، أي أن تلفح loop ال m times loop الأولى والخلي هي L:4 عندما k=1 m تم تصل للend for loop. وتفص يتفح أن تجتاز & m=4 يتراوح عدد المرات الأخرى حتى تصبح 4 loops. ولدنا أيضًا لتحديد قيمة عناصر هذه المصفوفة:

<table>
<thead>
<tr>
<th>1 2 3 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 0 -3 -7</td>
</tr>
<tr>
<td>3 3 6 13</td>
</tr>
<tr>
<td>4 1 -5 -18</td>
</tr>
</tbody>
</table>

k=1, m=1:4
k=2, m=1:4
k=3, m=1:4
k=4, m=1:4

لعدد الأول: if k==1 هذا الشرط فقط ينطبق على السطر الأول من المصفوفة.

في الشرط الثاني: elif m=1 ينطبق على العمود الأول من المصفوفة.

في الشرط الثالث: if m=1 المرات، مرة في كل دورة ل k عند أول قيمة ل m.

الأمر الثالث: يحقق بباقي عنصر المصفوفة معاً السطر الأول والعمود الأول.

بالنسبة للأمر الثالث حتى لاأخذ معنا وقت بالحسابات وتتلاشى حدوث ارتكاب أو خطأ بالحل نتبع التالي:

- نوجد علاقة على المصفوفة نفسها كما هو موضح بالرسم أدناه
- ونطبق هذه الطريقة على كل العناصر المتبقية.

المعادلة السابقة تعني العنصر الجديد=العنصر السابق+العنصر السابق ل نفس السطر- العنصر السابق له بنفس العمود (النظرى الرسم أدناه)
Question 6:
a. What does the following program segments do?

<table>
<thead>
<tr>
<th>PROGRAMS</th>
<th>OUTPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. clc</td>
<td></td>
</tr>
<tr>
<td>for i=1:1</td>
<td></td>
</tr>
<tr>
<td>for j=1:2</td>
<td></td>
</tr>
<tr>
<td>for k=1:3</td>
<td></td>
</tr>
<tr>
<td>fprintf('*')</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>
| fprintf('
')|     |
| end      |         |
| fprintf('
')|     |
| end      |         |
|          | * * *   |
|          | * * *   |

<table>
<thead>
<tr>
<th>PROGRAMS</th>
<th>OUTPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. clc</td>
<td></td>
</tr>
<tr>
<td>for i=1:2</td>
<td></td>
</tr>
<tr>
<td>for j=1:2</td>
<td></td>
</tr>
<tr>
<td>for k=1:2</td>
<td></td>
</tr>
<tr>
<td>fprintf('*')</td>
<td></td>
</tr>
<tr>
<td>end</td>
<td></td>
</tr>
</tbody>
</table>
| fprintf('
')|     |
| end      |         |
| fprintf('
')|     |
| end      |         |
|          | *       |
|          | *       |
|          | *       |
|          | *       |

b. given the following loop:
for i = 1:5
    for j = i/2 : i
        fprintf('one');
    end
    fprintf('two');
end

Find the number of times that each string in the fprintf statements will be printed

Solution:

one → 11 times
two → 5 times

![Diagram showing iterations and outputs](attachment:image.png)
**Question 7:** Write a program in a script file that finds the smallest even number and whose square is greater than 2000. Use a loop in the program. The loop should start from 1 and stop when the number is found.

**Solution:**

```plaintext
n=1;
while n^2<2000|rem(n,2)==0
n=n+1;
end
fprintf('the smallest even number whose square is greater than 2000 is %f',n)
```

**Question 8:** Write a function m-file that creates an m by n matrix in which the value of each element is:

- The multiple of its indices (the row number and column number of the element) if the element is an even-numbered column or row.
- The sum of its indices squared if the element is not in an even-numbered column or row.

For example:

If the user enters m=2 and n=4 then the resulted matrix should be as follow:

```
   2   2  10   4
   2   4   6   8
```

**Solution:**

```plaintext
function [a]=mat(m,n)
for i=1:m
    for j=1:n
        if rem(i,2) == 0|rem(j,2) == 0
            a(i,j)=i*j;
        else
            a(i,j)=i^2+j^2;
        end
    end
end
```
Question 9: Write a script file to print out the electricity bills. The program reads the current and previous readings of the gauge in Kwatt, then calculates the total charges as follows:
1. the customers pays 0.5 NIS per Kwatt for the first 200 Kwatts, 0.7 NIS per Kwatt for the next 200 Kwatt, and 0.9 NIS for every extra KW.
2. An additional 8% sales tax and 15 NIS monthly services are added to the bill.

Solution:
current=input('Enter the current in kwatt : ');
previous=input(' Enter the previous reading in kwatt');
cr=current-previous;  
if cr <=200
    pay=0.5*cr;
else if cr <=400& cr >=200
    pay=0.5*200+0.7*(cr-200)
else
    pay=0.5*200+0.7*200+0.9*(cr-400);
end
bill=pay+0.8*pay+15;
fprintf('you must pay %g NIS ',bill)
Question 10: Write a function m-file called xtymin that takes a vector x and returned a vector y whose elements values are of vector x with the minimum value of x vector subtracted from each value in the vector x. (each element in vector y = the corresponding element in vector x – minimum element in vector x)

For example:

If $x = [10 \ 2 \ 5 \ 3]$  
Then $y = [8 \ 0 \ 3 \ 1]$

The function should also return the sum of elements in the new vector y.

Note: you can’t use the MATLAB built in functions: min and sum.

Solution:
function [y,s]=xtymin(x)
n=length(x);
smallest = x(1); %suppose the first element in the vector is the minimum
for k=2:n
    if x(k)<=smallest
        smallest = x(k)
    end
end
x=x-smallest;
y=x;
s=0;
for i=1:n
    s=s+x(i);
end
s;
Question 11: Write a script m-file that print out all the three digit integers which satisfy the following:

\[ KLM = K^3 + L^3 + M^3 \]

For example, the number \(153 = 1^3 + 5^3 + 3^3\)

Solution:
for \(i=100:999\)
    \(a = \text{str2num}(i(1));\)
    \(b = \text{str2num}(i(2));\)
    \(c = \text{str2num}(i(3));\)
    if \((a^3 + b^3 + c^3) == \text{str2num}(i)\)
        fprintf(’%g’,str2num(i));
    end
end

Question 12: Write a program in a function file that takes \(n\) as input and returns a vector \(y\) as output. This output vector \(y\) is the same as vector \(x=[1:n]\) but with all elements (that divides by 2, 3, or 5) being removed.

Solution:
function \(y=x2y(x)\)
    \(n = \text{length}(x)\)
    for \(k=1:n\)
        if \(
            \text{rem}(x(k),2)==0 \|
            \text{rem}(x(k),3)==0 \|
            \text{rem}(x(k),5)==0
        \);
            \(x(k)=[\ ]\)
        end
    end
    \(y=x\)