

Lab Work #4

19/03/2008

Task 1. Boolean Logic in Matrices

Relational and logical operators are the two types of operators that produce true/false results in MATLAB programs. MATLAB interprets a zero value as false and any nonzero value as true. Relational operators (==, =, >, >=, <, <=) are operators with two operands that produce either a true (1) or a false (0) result, depending on the values of the operands. Relational operators can be used to compare a scalar value with an array. They can also be used to compare two arrays or two strings only if they have the same size. Be careful not to confuse the equivalence relational operator (==) with the assignment operator (=). Logic operators (&, |, xor, ~) are operators with one or two operands that yield a logical result such as 0 or 1. There are three binary logic operators: AND (&), OR (|), and exclusive OR (xor); and one unary operator: NOT (~). In the hierarchy of operations, logic operators are evaluated after all arithmetic and relational operators have been evaluated. The operator is evaluated before other logic operators. Create Matrix L given below.

$$L = \begin{vmatrix} 1 & 2 & 3 & 4 \\ -2 & 1 & 2 & 3 \\ -3 & -2 & 1 & 2 \\ -4 & -3 & -2 & 1 \end{vmatrix}$$

Using a Boolean expression convert L to Identity Matrix

Task 2. Over-time Calculator

You will use a matrix, which represents employee work log. Rows of the matrix represent's days, while columns represent individual workers. The employer wants to use this matrix to calculate the extra fee should be paid for overtime. Each worker are obliged to work 7 hours per day, every hour above this is considered as over time. Fortunately for the employer until a given threshold value, no over time fee is paid.

You should write a script which first inputs this matrix and the threshold number (ie. if over time is below 2 hours discard it). Your script should calculate the over-times and filter out over times lower than the threshold. Draw in a figure two plots, first plot should show the total over-time vs. days. Secondly draw a bar chart, where each bar shows each employee's total over-time.

Task 3. Russian Peasant Multiplication

In this task you should code a script, which simply does the multiplication of two integers. There is a very old multiplication technique, used in Russia. This technique keeps on halving number on the left and doubles the other number on the right. We continue the process until the column in which we are halving reaches the number 1. Also, one other little thing must be observed: when we take the half of an odd number, there is naturally a "remainder of 1"; so we just discard it. The result of the multiplication is the sum of doubled numbers, while the halved number is odd. The example below will clarify the algorithm.

22x35 =

```

22 35
11 70 +
 5 140 +
 2 280
 1 560 +

```

$$22 \times 35 = 70 + 140 + 560 = 770$$

Write a script which inputs two numbers and carries out the Russian Multiplication Steps and prints out the result in a similar fashion as above.

For those wondering why this multiplication is interesting, note that you only need multiplication by 2 and division by 2, which are very efficient for computers.

Task 4. Random Maze

In this task you will simulate a bug in a maze which moves randomly to 4 directions (up, down, right, left). For this task you will be using a matrix as the maze. Below is an example maze.

```

1 1 1 1 1 1 1
1 9 0 0 0 0 1
1 0 0 0 0 0 1
1 0 0 0 0 0 1
1 0 0 0 0 0 1
1 0 0 0 0 0 1
1 0 0 0 0 0 1
1 0 0 0 0 0 6
1 1 1 1 1 1 1

```

Note that in this matrix 9 represents your bug, 1 represents an obstacle and 6 is the

exit. You should write a script which moves 9 to a random available direction (you can not move to a 1 square) until it eventually exits the maze (moves to 6 square). Write out the matrix in every move.