

CS 324 Homework Assignment 3

Due: 11:59pm, Thursday, October 19th

This assignment is scored out of 65. It consists of 6 questions. When you submit, you are required to create a folder with your name (Last name first, then First name), CS324, HW3, e.g., LastName_FirstName_CS324_HW3. Type your answers into a text file (**only .txt, .doc, and .pdf file formats are accepted**) and save it in this folder. Put all your Java programs (***.java**) as well as output files in the same folder. Zip this folder, and submit it as one file to Desire2Learn. Do not hand in any printouts. Triple check your assignment before you submit. **If you submit multiple times, only your latest version will be graded and its timestamp will be used to determine whether a late penalty should be applied.**

Short Answers

P1. (6pts) The following path matrix P is passed back to you from Floyd's algorithm on a directed graph. Include the endpoints in your path.

P	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9
v_1	0	7	7	0	7	7	0	0	7
v_2	0	0	7	0	7	9	0	1	0
v_3	9	0	0	0	7	0	2	9	6
v_4	9	8	8	0	8	8	8	0	8
v_5	9	3	0	0	0	3	3	9	6
v_6	9	9	9	0	9	0	9	9	0
v_7	9	5	5	0	0	5	0	9	6
v_8	9	7	7	0	7	7	0	0	7
v_9	0	7	7	0	7	0	1	1	0

- Give a path starting from v_8 ending at v_9 of minimum cost.
- Give a path starting from v_6 ending at v_3 of minimum cost.

P2. (8pts) You are provided with four path matrices for directed graphs but not all of them are valid. Which of the following P's are valid, and which are invalid? Explain your answers.

P	v_1	v_2	v_3
v_1	0	3	0
v_2	1	0	2
v_3	2	0	0

(A)

P	v_1	v_2	v_3
v_1	0	0	2
v_2	3	0	0
v_3	0	0	0

(B)

P	v_1	v_2	v_3
v_1	0	3	2
v_2	0	0	0
v_3	0	1	0

(C)

P	v_1	v_2	v_3
v_1	0	0	2
v_2	0	0	0
v_3	0	0	2

(D)

P3. (6pts) The following order matrix P is passed back to you from the Minimum Multiplication algorithm.

P	1	2	3	4	5	6	7	8
1		1	2	1	4	4	4	4
2			2	2	4	2	3	2
3				3	4	4	4	4
4					4	4	5	4
5						5	6	7
6							6	6
7								7
8								

- What is the optimal order in which to multiply matrices A_1 through A_8 ?
- What is the optimal order in which to multiply matrices A_2 through A_7 ?

P4. (4pts) Finish computing the optimal way to multiply matrices A_1 through A_5 , where their dimensions are as follows (You can assume that the array index begins at 1).

$$A_1: 6 \times 5 \quad A_2: 5 \times 4 \quad A_3: 4 \times 6 \quad A_4: 6 \times 5 \quad A_5: 5 \times 8$$

M	1	2	3	4	5
1	0	120	264	360	
2		0	120	220	420
3			0	120	280
4				0	240
5					0

What are the values of $M[1][5]$ and $P[1][5]$?

P5. (6pts) Give the denominations of three pretend coins (one must be a 1 cent coin, the other two must be between 2 cents and 100 cents) and an amount such that the greedy change algorithm does not yield the fewest number of coins. How many coins does the greedy change algorithm use? What is the optimal solution?

See next page for programming questions.

Programming Questions

P6. (35pts)

a. Completing the Homework3 class

You are provided with two files "Homework3.java" and "TestHomework3.java". You are required complete the following three methods in the former:

```
void booleanFloyd(int n, boolean[][] W, boolean[][] D)
```

This method takes as parameters an `int n` as well as two `boolean` matrices `W` and `D`. A graph (that has `n` vertices) is represented by `W`, where `W[i][j] = true` if and only if $i = j$ or there is an **edge** from v_i to v_j . The method finds the connectivity for every pair of vertices in the graph and save this information in `D`, so that `D[i][j] = true` if and only if $i = j$ or there is a **path** from v_i to v_j . Note that you should assume that the indices of both matrices start at 1. **You should NOT change the content of `w` and you are NOT supposed to create any additional arrays in the implementation.**

```
int numOrders(int n)
```

This method takes an `int n` and returns the number of orders to multiply `n` chained matrices. For example, there is only one order to multiply two matrices: (A_1A_2) , two orders to multiply three matrices: $(A_1A_2)A_3$, $A_1(A_2A_3)$, and five orders to multiply four matrices (see lecture slides). You are not required to list all the combinations, just return the number of orders for a given `n`.

```
int minMulti(int n, int[] d, int[][] M, int[][] P)
```

This method takes as parameters the number of matrices, the matrix dimensions, and two matrices `M` and `P`. It builds an order matrix (saved in the two-dimensional `int` array `P`) and returns the minimum number of element multiplications when multiplying matrices A_1 through A_n . Note that you should assume that the indices of both matrices start at 1.

Note that you are only supposed to touch the above methods. You are NOT allowed to create any other methods, instance variables, or make any changes to methods other than the above methods or files other than "Homework3.java". Points will be taken off if you fail to follow this rule.

b. Code Testing

You are provided with a test driver implemented by "TestHomework3.java" (**Do not make any changes to this file!**) so there is no need to write your own.

Once you have completed the above method, you can run the test. You should create a plain text file named "output.txt", copy and paste the output (if your code crashes or does not compile, copy and paste the error messages) to this file and save it.

Grading Rubrics:

Code does not compile: -10

Code compiles but crashes when executed: -5

Changes were made to things other than the required methods: -5

Has output file: 5

`booleanFloyd` was correctly implemented: 10

booleanFloyd changes the content of array **w**: -5
booleanFloyd creates additional array(s): -5
numOrders was correctly implemented: 10
minMulti was correctly implemented: 10

Sample Output:

Test 1: booleanFloyd[Passed]

W =

T	T	F	T	T
T	T	T	T	F
F	F	T	T	F
F	F	T	T	T
T	F	F	F	T

Expected:

D =

T	T	T	T	T
T	T	T	T	T
T	T	T	T	T
T	T	T	T	T
T	T	T	T	T

Yours:

D =

T	T	T	T	T
T	T	T	T	T
T	T	T	T	T
T	T	T	T	T
T	T	T	T	T

...

Test 4: numOrders(2) - [Passed]

Expected: 1

Yours: 1

Test 5: numOrders(4) - [Passed]

Expected: 5

Yours: 5

...

Test 10: d = [9, 8, 7, 6, 5, 4, 3, 2, 1], buildOrder(1, 8) - [Passed]

Expected: min # multiplications = 238 and order is A1(A2(A3(A4(A5(A6(A7A8))))))

Yours: min # multiplications = 238 and order is A1(A2(A3(A4(A5(A6(A7A8))))))

Total test cases: 10

Correct: 10

Wrong: 0