CS 320 - Spring 2023

Instructor: Meenakshi Syamkumar

Exam 3 — 15%

(Last) Surname: _____ (First) Given name: _____

NetID (email): _____ @wisc.edu

Fill in these fields (left to right) on the scantron form (use #2 pencil):

1. LAST NAME (surname) and FIRST NAME (given name), fill in bubbles
2. IDENTIFICATION NUMBER is your Campus ID number, fill in bubbles
3. Under *ABC* of SPECIAL CODES, write your lecture number, fill in bubbles:
   001 - MWF 11:00am
   002 - MWF 1:20pm
4. Under **F** of SPECIAL CODES, write **1** and fill in bubble **1**

If you miss step 4 above (or do it wrong), the system may not grade you against the correct answer key, and your grade will be no better than if you were to randomly guess on each question. So don't forget and double check it's correct!

You may only reference your note sheet. You may not use books, calculators, or other electronic devices during this exam. You may not sit near your friends or look at your neighbors during this exam. Please place your student ID face up on your desk. Turn off and put away portable electronics (including smart watches) now.

Use a #2 pencil to mark all answers. DO NOT USE PEN on the scantron.

When you're done, please hand in the exam and note sheet and your filled-in scantron form. The note sheet will not be returned.

(Blank Page)

1. Considering the following code for `PCA`, which of the following approximately reconstructs the original dataframe `df` using the first three components?

```
p = PCA()
W = p.fit_transform(df)
C = p.components_
```

A. `pd.DataFrame(W[:, :3] @ C[:3, :]  + p.mean_)`

B. `pd.DataFrame(W[:3, :]  @ C[3:, :]  + p.mean_)`

C. `pd.DataFrame(W[:, :3] @ C[:, :3] + p.mean_)`

D. `pd.DataFrame(W[3:, :]  @ C[:, :3] + p.mean_)`

2. Which dictionary corresponds to the query string in the following URL?

```
https://wisc.edu/abc?item=book&time=tomorrow
```

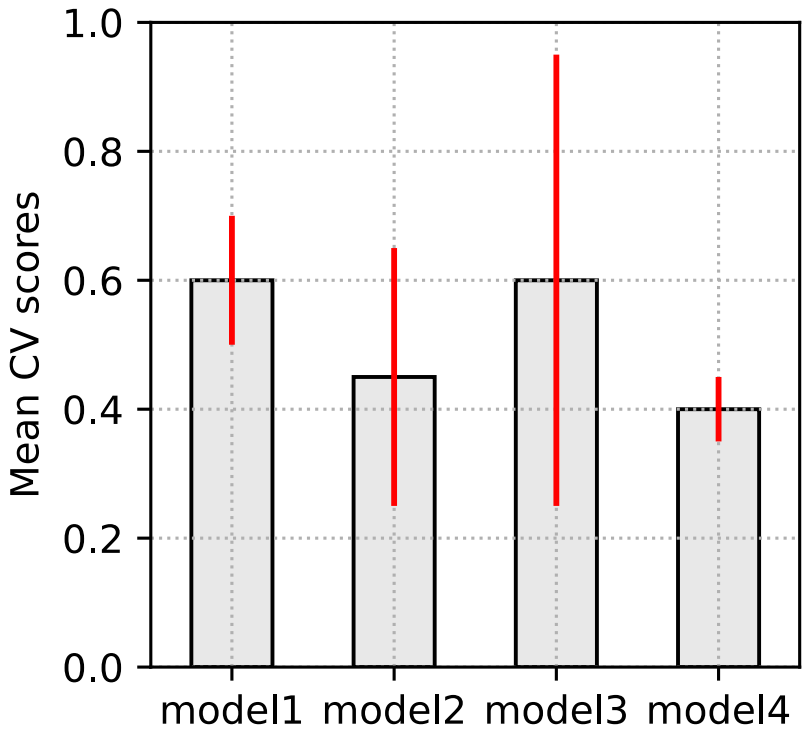
A. `{"item":"book&time=tomorrow"}`

B. `{"abc":"item=book&time=tomorrow" }`

C. `{"item":"book", "time":"tomorrow"}`

D. `{"wisc.edu":"abc", "item":"book", "time":"tomorrow"}`

3. The bars and error bars in the following plot represent the means and standard deviations of the cross validation (CV) scores of four models, respectively. Which model performs the best?



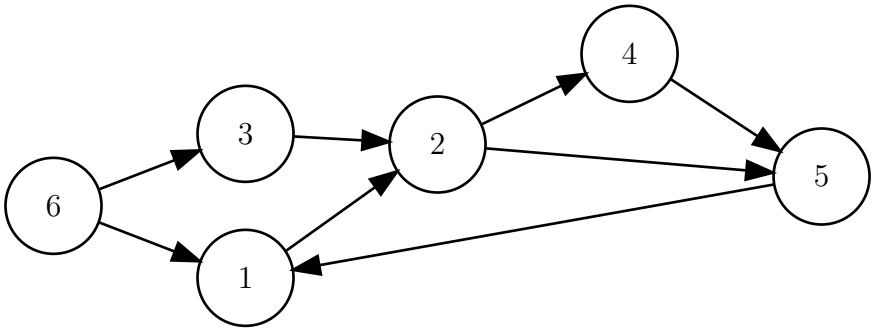
A. model1 B. model2 C. model3 D. model4

4. Assume `Feature 1` is a numerical column and `Feature 2` is a categorical column containing 5 categories. How many output columns will there be after we apply `custom_transformer`?

```
custom_transformer = make_column_transformer(
    (PolynomialFeatures(degree=3, include_bias=False), ["Feature 1"]),
    (OneHotEncoder(), ["Feature 2"]),
)
```

A. 6 B. 7 **C. 8** D. 9 E. 10

5. What can be said about the following graph?



A. It is cyclic and strongly connected

B. It is not cyclic but strongly connected

C. It is cyclic but not strongly connected

D. It is not cyclic and not strongly connected

6. Given points `[(7, 7), (1, 4), (2, 3), (8, 6)]` and starting centroids `[(6, 9), (2, 2)]`, what are the centroids after the first iteration of assigning points and updating centroids, using the iterative K-Means Clustering algorithm discussed in class?

A. `[(4, 5.5), (5, 4.5)]`

B. `[(5, 4.5), (4, 5.5)]`

C. `[(7.5, 6.5), (1.5, 3.5)]`

D. `[(1.5, 3.5), (7.5, 6.5)]`

7. For the regular expression `^[^d-g]*(ab)+.*c$` which of the following strings would match?

A. `"ac"`

B. `"abcd"`

C. `"eabc"`

D. `"gab10c"`

E. `"abababc"`

8. What is the complexity of the below code snippet?

```
some_nums = [6, 36, 64, 84, 47, 6, 31, 28, 57, 75]
selected_nums = []
for num in some_nums:
    if (num > min(some_nums) * 1.5) and (num < max(some_nums) / 1.5):
        selected_nums.append(num)
print(selected_nums)
```

A. O(N) B. O(N log N) **C. O(N**2)** D. O(N**3)

9. What does the following code snippet print?

```
import re

print(re.sub(r"([A-Z])([a-z]*)", "\g<2>.\g<1>", "Hello.World!"))
```

A. `World.Hello!`

B. `World!.Hello`

C. `ello.H.orld.W!`

D. `orld.W.ello.H!`

10. Given the following recursive function, what is `mystery(7)`?

```
def mystery(a):
    if a <= 2:
        return 1
    return mystery(a-1) + mystery(a-3)
```

A. 7 **B. 9** C. 10 D. 15

11. Which of the following can be performed by the process of broadcasting?

A. add dimension of size 1 to the beginning of a shape

B. stretching 2 to N along any dimension to make shapes compatible

C. performing dot product between matrices

D. performing element-wise multiplication between matrices

12. Given the below contingency table, what is B's CTR (click-through rate)?

	click	no-click
A	300	700
B	280	720

A. 0.28 B. 0.3 C. 0.38 D. 0.42

13. What does the following code snippet print?

```
import numpy

arr = numpy.array([
    [1, 8, 6],
    [3, 5, 7],
    [4, 9, 2]
])
print(numpy.argmin(arr, axis=1))
```

A. `[0, 0, 2]`

B. `[0, 1, 2]`

C. `[1, 5, 2]`

D. `[1, 3, 2]`

14. If `A = np.array([[3, 3], [1, 5]])` and `b = np.array([[4], [2]])`, what is `b * A`?

A. `np.array([18, 14])`

B. `np.array([[18], [14]])`

C. `np.array([[12, 6], [4, 10]])`

D. `np.array([[12, 12], [2, 10]])`

15. Which of the following ML implementations enables us to predict catgeorical labels?

A. `LinearRegression`

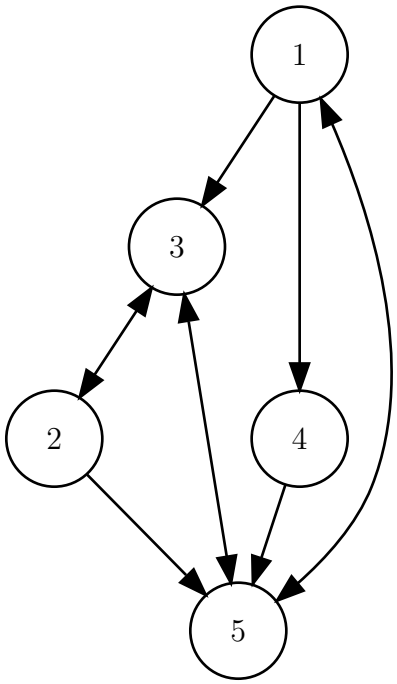
B. `LogisticRegression`

C. `KMeans`

D. `AgglomerativeClustering`

E. `PCA`

16. What is the order in which the nodes of the below directed graph are visited in a DFS starting from node 5? When you have the choice of two or more nodes, break ties by choosing the node with smaller value.



A. [5, 3, 1, 2, 4]

B. [5, 1, 3, 4, 2]

C. [5, 1, 3, 2, 4]

D. [5, 1, 4, 3, 2]

17. When implementing an object to be used like a Python `dict`, what special method is necessary to enable lookup / subscription operation?

A. `__contains__` B. `__eq__` C. `__repr__` D. `__enter__` **E.** `__getitem__`

18. Which of the following is true about threads and processes?

A. Threads share the same data, whereas processes have their own data.

B. Processes share the same data, whereas threads have their own data.

C. Both threads and processes have their own data.

D. Both threads and processes share the same data.

19. Which of the following **best** describes a dendrogram?

A. graph B. tree C. BST **D. binary tree** E. DAG

20. (**This question is dropped due to typo**) Which of the following vectors is **NOT** in the column space of `X`?

```
X = np.array([
    [2, 4, 0],
    [1, 2, 0],
    [3, 6, 5]
])
```

A. `np.array([[0], [0], [0]])`

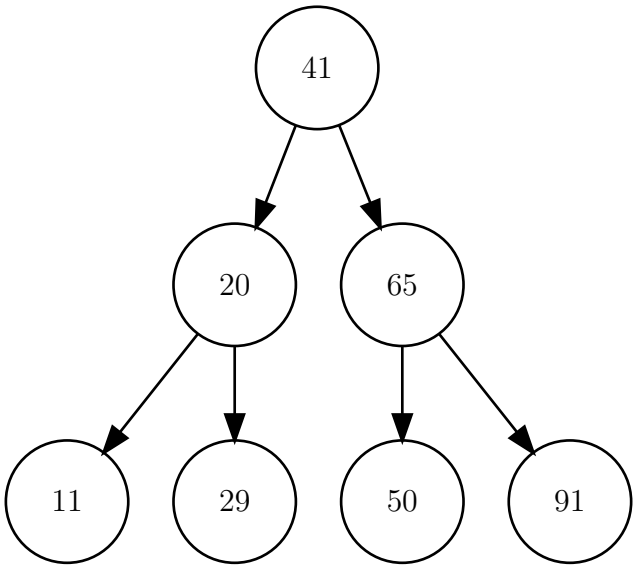
B. `np.array([[10], [5], [15]])`

C. `np.array([[-8], [-4], [-12]])`

D. `np.array([[18], [9], [0]])`

E. `np.array([[6], [3], [-11]])`

21. Consider the BST insertion algorithm we learned in class. Given the below BST, which of the following **cannot** be the insertion order? For every node, consider first child as left and second child as right.



A. [41, 65, 29, 20, 11, 50, 91]

B. [41, 65, 91, 20, 29, 50, 11]

C. [41, 20, 29, 11, 65, 50, 91]

D. [41, 20, 11, 65, 50, 91, 29]

22. Given the following confusion matrix, what is the recall for Versicolour? Row dimension represents actual value and column dimension represents predicted value.

	Setosa	Versicolour	Virginica
Setosa	30	20	0
Versicolour	5	15	10
Virginica	15	15	30

A. 0.25 B. 0.3 **C. 0.5** D. 0.6 E. 0.75

23. What does the following code snippet print?

```
import heapq

items = []
heapq.heappush(items, 2)
print(heapq.heappop(items))

heapq.heappush(items, -3)
heapq.heappush(items, 0)
print(heapq.heappop(items))

heapq.heappush(items, 4)
heapq.heappush(items, 1)
print(heapq.heappop(items))
```

A. 2 0 4 **B. 2 -3 0** C. 2 -3 1 D. -3 0 1 E. -3 0 4

24. Given two shapely shapes `X` and `Y`, which of the following enables us to determine the overlapping part (another shapely shape) between them?

A. `X.difference(Y)`

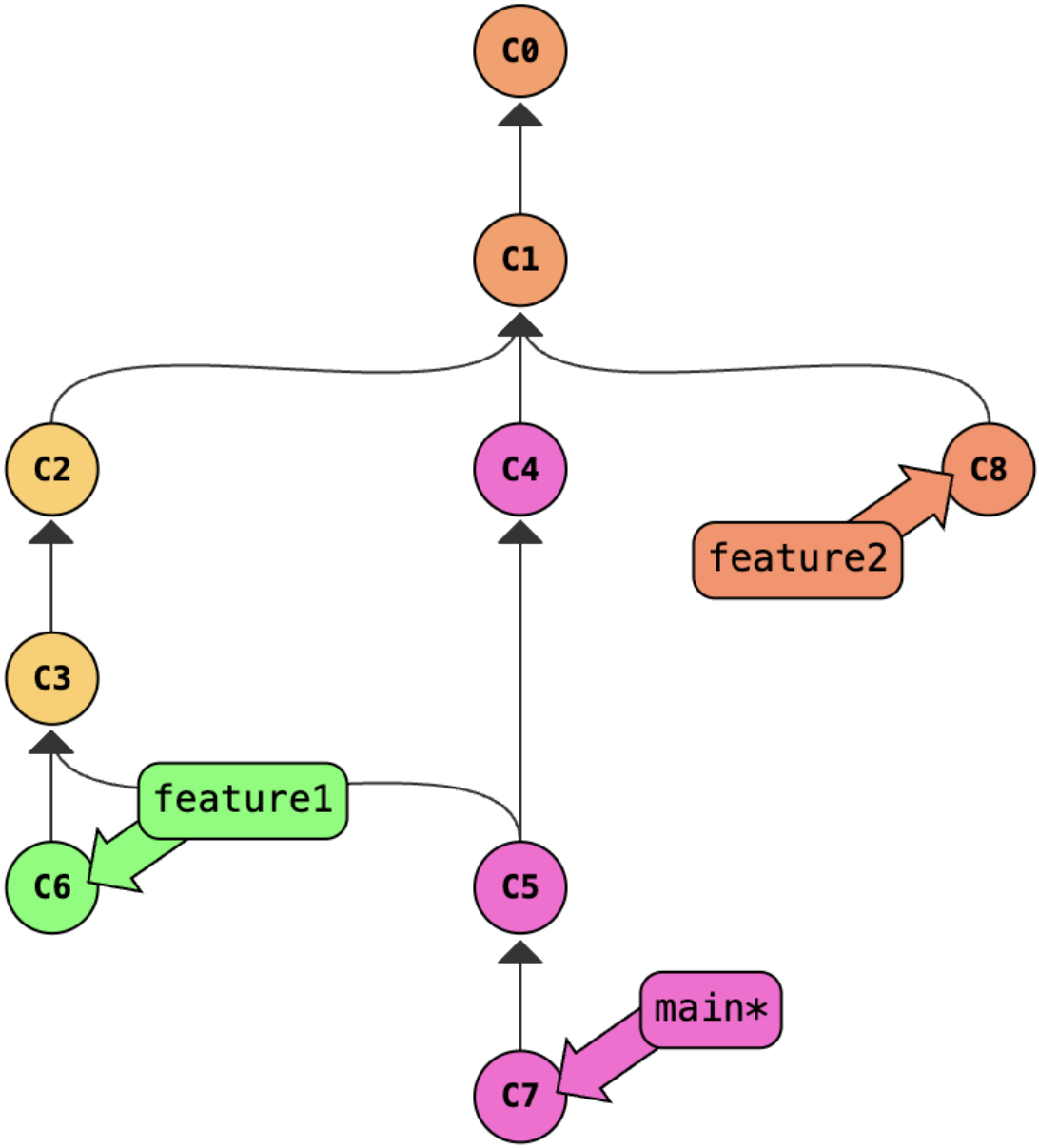
B. `Y.difference(X)`

C. `X.intersects(Y)`

D. `X.intersection(Y)`

E. `X.union(Y)`

25. Given the below git commit graph, which of the following commits should be part of the `main` branch?



A. C0, C1, C4, C5, C7

B. C0, C1, C4, C2, C3, C5, C7

C. C0, C1, C4, C2, C3, C5, C6, C7

D. C0, C1, C4, C2, C3, C5, C6, C7, C8

26. If `A = np.array([[3, 3], [1, 5]])` and `b = np.array([[4], [2]])`, what is `A @ b`?

A. `np.array([18, 14])`

B. `np.array([[18], [14]])`

C. `np.array([[12, 6], [4, 10]])`

D. `np.array([[12, 12], [2, 10]])`

27. What will the following code print?

```
class Calculator:
    def __init__(self):
        self.num1 = 2
        self.num2 = 3

    def multiplication(self):
        return self.num1 * self.num2

    def square(self):
        return self.num2 * self.num2

class AdvancedCalculator(Calculator):
    def __init__(self):
        super().__init__()
        self.num2 = 5
        self.const = -1

    def multiplication(self):
        return self.num1 * self.num2 * self.const

calc = AdvancedCalculator()
print(calc.multiplication(), calc.square())
```

A. -10 9 **B. -10 25** C. -6 9 D. -6 25 E. 6 9

28. A flask application has three handlers. Combined, how many times will these handlers be invoked each time somebody visits the home page in a browser?

```
@app.route("/")
def home():
    return """
    <html>
    <body>
    <img src="animal.svg?type=bear">
    <img src="animal.svg?type=bird">
    <img src="bird.svg">
    </body>
    </html>
    """
@app.route("/animal.svg")
def animal():
    # HIDDEN

@app.route("/bird.svg")
def bird():
    # HIDDEN
```

A. 1 B. 2 C. 3 **D. 4**

29. Given the following HTML snippet:

```
<div id="data-programming">
    <h2>Data Programming Courses</h2>
    <ul>
        <li>CS 220</li>
        <li>CS 320</li>
    </ul>
</div>
<div id="programming">
    <h2>Programming Courses</h2>
    <ul>
        <li>CS 200</li>
        <li>CS 300</li>
        <li>CS 400</li>
    </ul>
</div>
```

Assuming `driver` is correctly initialized, what does the following code snippet print?

```
element = driver.find_element("id", "programming")
elements = element.find_elements("tag name", "li")
print(len(elements))
```

A. 2 **B. 3** C. 4 D. 5

30. You encountered the following warning when fitting a `LogisticRegression` model.

```
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Which of the following best resolves this issue?

A. Apply `StandardScalar` on feature columns before `LogisticRegression`

B. Apply `OneHotEncoder` on feature columns before `LogisticRegression`

C. Apply `PolynomialFeatures` on feature columns before `LogisticRegression`

D. Ignore the warning

(Blank Page)