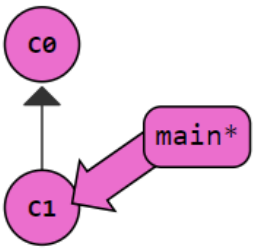
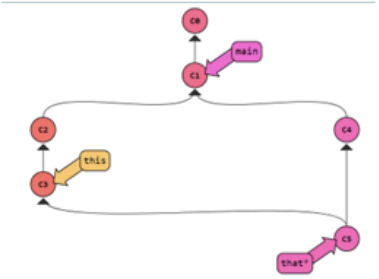
Q1

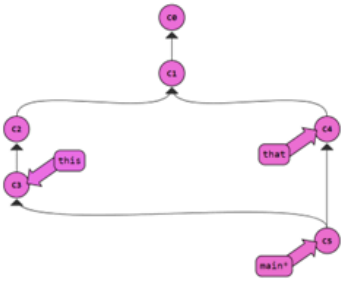
Consider a Git tree that currently looks like this:

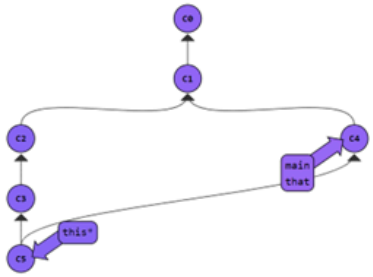


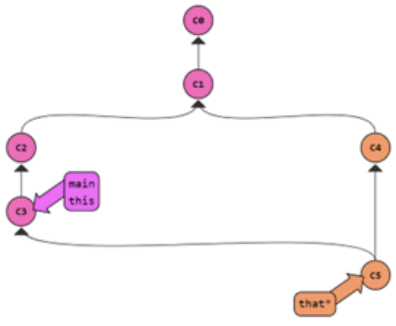
Which of the following trees would be generated by the following set of commands?

```
$ git branch this
$ git checkout this
$ git commit
$ git commit
$ git checkout main
$ git branch that
$ git checkout that
$ git commit
$ git merge this
$ git merge main
```









Q2

If numpy array `x` has a shape of `(20,6)` and numpy array `y` has a shape of `(20,30,6)` , how can we reshape `x` such that `x*y` can be performed?

- x.reshape(20, -1, 6)
- x.reshape(-1,20)
- x.reshape(-1)
- x.reshape(20,-1)

Q3

```
@app.route("/")
def home():
    return """
    <html>
    <head>Page</head>
    <body>
    <a href="A.html">A page</a>
    <a href="B.html">B page</a>
    </body>
    </html>
    """

@app.route("/A.html")
def B():
    return """
    <html>
    <head>Page</head>
    <body>
    <a href="A.html">A page</a>
    </body>
    </html>
    """

@app.route("/B.html")
def A():
    return 'D'
```

How many total handlers will be invoked every time the homepage is visited?

- 4
- 1
- 2
- 3

Q4

There are two polygons with the coordinates -

```
polygon1 = Polygon([(0, 0), (2, 0), (2, 2), (0, 2)])
polygon2 = Polygon([(1, 1), (3, 1), (3, 3), (1, 3)])
```

Upon conducting the following operations -

```
polygon1_gs = gpd.GeoSeries([polygon1])
polygon2_gs = gpd.GeoSeries([polygon2])

intersection = polygon1_gs.intersects(polygon2_gs)
```

Which of the following will be returned?

- (1, 0), (3, 2), (3, 2), (1, 0)
- (0, 1), (2, 3), (2, 3), (0, 1)
- (0, 1), (2, 1), (2, 3), (0, 3)
- (1,0), (3, 0), (3,2), (1, 2)
- True
- False

Q5

Consider the following code

```
arr = np.array([0.2, 0.4, 0.6, 0.1, 0.3, 0.2, ])

arr = arr.reshape(2,3)
```

What is displayed by `arr.mean(axis=1)`?

- array([0.4, 0.2])
- array([0.15, 0.35, 0.4])
- 0.3
- array([[0.2],
  0.4]])

Q6

Which of the following is **not** a benefit of process level parallelism?

- Computer resources can be used effectively
- Processes do not occupy the CPU if they are blocked
- There are too many tasks for the CPU

Q7

Consider the following code.

```
mod =LinearRegression()
df = pd.DataFrame([{"a":3, "b":5},
                   {"a":4, "b":7},
                   {"a":1, "b":3},
                  {"a":2, "b":5}])

Xcols = "a"
Ycol = "b"
```

Which of the following will **not** give an error?

- mod.fit(df[Xcols],df[Ycol])
- mod.fit(df[Xcols],df[["b"]])
- mod.fit(df[["a"]],df[Ycol])
- mod.fit(df[Xcols],df["b"])
- mod.fit(df["a"],df[Ycol])

Q8

Which of the following is **false** about the projection matrix?

- By hand, it's is intensive to calculate
- Multiplying a vector by the projection matrix gives the same vector
- It is used to a obtain a solvable system of equations

Q9

Rows represent the actual labels and columns represent the predicted labels, that is, consider the following as a confusion matrix as explained in the class.

	pink	blue	green	yellow
pink	2	0	2	3
blue	4	2	0	0
green	1	0	2	0
yellow	0	0	0	2

What is the accuracy? Note: answers are rounded to two decimal places.

- 0.33
- 0.44
- 0.50
- 0.67

Q10

```
arr = np.array([[6,7], [1,9], [4,8]])
arr = arr.reshape(2,3)
```

Choose the correct answer.

Note: in the following options, the equal sign is not an assignment operator, but it means that the left hand side of the equal sign is equal to its right hand si

- `arr.argmin(axis=0) = array([0, 0, 0])`
- `arr.argmin(axis=1) = array([2,1])`
- `arr.argmin(axis=0) = array([0,1])`
- `arr.argmin(axis=1) = array([0, 1, 0])`

Q11

Given the following the mean and variance of the cross-validation scores for 4 different Linear Regression models, select the best model

- Mean = 0.789, Var = 0.08
- Mean = 0.689, Var = 0.05
- Mean = 0.823, Var = 0.04
- Mean = 0.725, Var = 0.09

Q12

What score does cross_val_score() calculate for a Logistic Regression model?

- R^2
- explained variance score
- accuracy
- F1-score

Q13

```
df = pd.DataFrame([{"a":3, "b":5, "c":8, "d":"Badger", "e":"tomato"},
                   {"a":4, "b":7, "c":16,  "d":"Zebra", "e":"potato"},
                   {"a":1, "b":3, "c":1, "d":"Elephant", "e":"potato"},
                   {"a":2, "b":5, "c":4, "d":"Badger","e":"potato"}])

Xcols = ["a","c"]
Ycol = "b"
pf = PolynomialFeatures(degree=4, include_bias = var)
xcols = ["d", "e"]

oh = OneHotEncoder()
```

Which of the following is correct?

- If var = True, pf.fit_transform(df[["a"]]) will return 4 columns
- If var = False, pf.fit_transform(df[["c"]]) will return 5 columns
- oh.fit_transform(df[xcols]).toarray() has shape (5,4)
- oh.fit_transform(df[xcols]).toarray() has shape (4,5)

Q14

PCA is applied to 6 columns of a dataset containing 32 rows. Using p = PCA(2). What is the shape of the p.components_?

- (6, 6)
- (32, 6)
- (2, 6)
- (32, 2)

Q15

Consider the system of points: (0,2) (1,1) (2,2) (3,5) (3,4) (4,4)
Using (1,4) and (5,2) as the centroids, predict the next set of centroids using KMeans.

- (1.67,1) (4.33,3.33)
- (1,1.67) (3.33,4.33)
- (2.8,1.8) (4,4)
- (1.8,2.8) (4,4)

Q16

Consider the points in an Euclidean plane: (6, 0), (7,0), (12,0), (15,0), (17,0).
Suppose agglomerative clustering is being used. What will be the cluster distances for point (12,0) under complete linkage?

Note: in the following options, [(a,0),(b,0)]:c means a cluster containing the points (a,0) and (b,0) is at a distance c from point (12,0), where (a, 0) and (b,0) represent the points (6, 0), (7,0), (12,0), (15,0), (17,0).

- [(6,0), (7,0)]: 5 and [(15,0), (17,0)]: 3
- [(6,0), (7,0)]: 6 and [(15,0), (17,0)]: 5
- [(6,0), (7,0)]: 5.5 and [(15,0), (17,0)]: 4
- (6,0), (7,0)]: 0.5 and [(15,0), (17,0)]: 2

[(6,0), (7,0)]: 5 and [(15,0), (17,0)]: 3.

Q17

Consider the following arrays

```
a = np.array([1, 2, 3]).reshape(-1, 1)
```

```
b = np.array([4, 5, 6, 7, 8, 9]).reshape(2, -1)
```

```
c = np.array([[[50, 70, 31], [ 0, 13, 44], [ 3, 37, 42]],
              [[70, 32, 63], [69, 71, 58], [11, 25, 34]]])
```

```
d = np.array([9,3,2,8,6,1]).reshape(6,1)
```

Which of the following will give a valid output?

- `a*b`
- `b*c`
- `b*a`
- `a*c`

Q18

Consider the numpy arrays `a` and `b` as defined in question 17 above.

What is the value of `b @ a`?

- array([[32], [50]])
- array([[32, 50]])
- array([[40], [46]])
- array([[40, 46]])

Q19

Consider the numpy arrays `a` and `b` as defined in question 17.

Which of the following operations will give an error?

b @ a.

- `a.T @ b.T`
- `a @ c.T`
- `c @ a`
- `b @ a`