Quiz 2 - Results

Attempt 1 of 1

Written Oct 2, 2025 2:18 PM - Oct 2, 2025 2:39 PM

Attempt Score **22 / 25 - B+**

Overall Grade (Highest Attempt) **22 / 25 - B+**

Question 1 1 / 1 point

In gradient descent, the learning rate is a parameter that controls the speed of the algorithm. What is the effect of decreasing learning rate on the speed of the algorithm?

- Gradient descent will converge more quickly to a solution
- ✓ Gradient descent will require more iterations to converge to a solution
- There is no effect of decreasing learning rate on the speed of the algorithm
- Gradient descent will oscillate between multiple solutions

Question 2 2 / 2 points

Which of the following holds true for Learning Rate?

- [x] A large learning rate may allow the algorithm to come close to a good solution but will then oscillate around the point or even diverge
- [x] The optimal solution for learning rates is to start with a higher value to find a good starting point for the weights and then move to a smaller value to converge to a minima
- [ ] None of the above
- [x] A low learning rate may cause the algorithm to take a too long time to come even close to an optimal solution

Question 3 1 / 1 point

Which of the following is true for the Computation Graph?

- [x] Computation Graph allows scope for optimization as we can parallelize the computations present in it.
- [ ] Computation Graph is the same as a neural network.
- [ ] The computation graph represents the components of a function and can have cycles in it.
- [x] In Pytorch, Autograd builds the Computation Graph dynamically.
- [ ] Generally, in Tensorflow, the computation graph is constructed at run-time whereas in PyTorch it is built at compile-time.

Question 4 1 / 1 point

Pytorch Dataloader can perform which of the following functions. Select all that apply.

- [ ] Update the model weights

- [x] Speed-up data loading with parallelization
- [x] Slice datasets into batches for training

Question 5 1 / 1 point

You will need to create a separate DataLoaders for the training, testing and validation datasets.

- [ ] False
- [x] True

Question 6 2 / 2 points

Select which of the following is true for start methods for a new process.

- [x] The fork method starts a new fresh process with minimal resources inherited.
- [x] The fork-Server method creates a new server process that forks new child processes. This helps in keeping the original process safer.
- [x] Usually, Spawn and fork-Server methods are slower but they are a safer option.
- [x] In the Spawn method, a new process is created which inherits all resources.

Question 7 1 / 1 point

Shared memory is a memory area that the OS (eg Linux) maps on the address space of the processes, allowing in this way to be simultaneously accessed by multiple programs with an intent to provide communication among them or avoid redundant copies.

- [ ] False

True

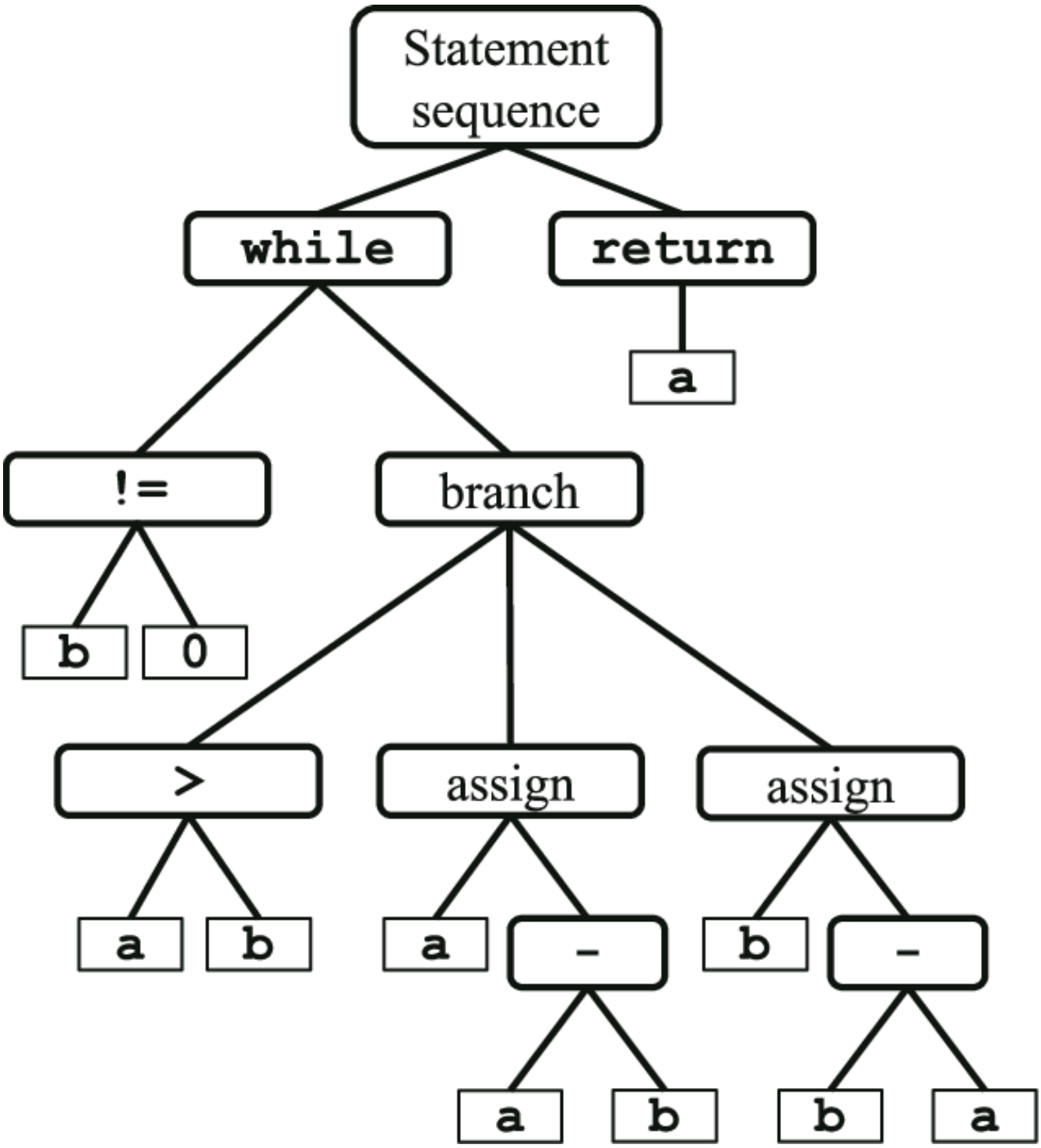
Question 8 0 / 2 points

Which of the following statements are true about optimizers? Select all that apply.

- Adam combines the advantages of RMSProp and momentum
- When working with sparse data, adaptive gradient algorithms should be preferred.
- Adam should be used with mini-batch gradient computations, not with batch gradient descent.
- For selecting the initial learning rate in AdaDelta one can perform search using validation set.

Question 9 1 / 1 point

Which of the following represents the given Abstract Syntax Tree in Python?


Statement sequence
while
return
a
!=
branch
b
0
>
assign
assign
a
b
a
-
b
-
a
b
b
a

○

c.
```
while b!=0:
        if a>b:
            a=a-b
        else:
            b=b-a
        return a
```

b.
```
while b!=0 and a>b:
        a=a-b
    b=b-a
    return a
```

a.
```
while b!=0:
        if a>b:
            a=a-b
        else:
            b=b-a
    return a
```

d.
```
while b!=0:
        if a>b:
            b=b-a
        else:
            a=a-b
    return a
```

Question 10 1 / 1 point

Which of the following statements are not true for Python bytecode?

None of the above

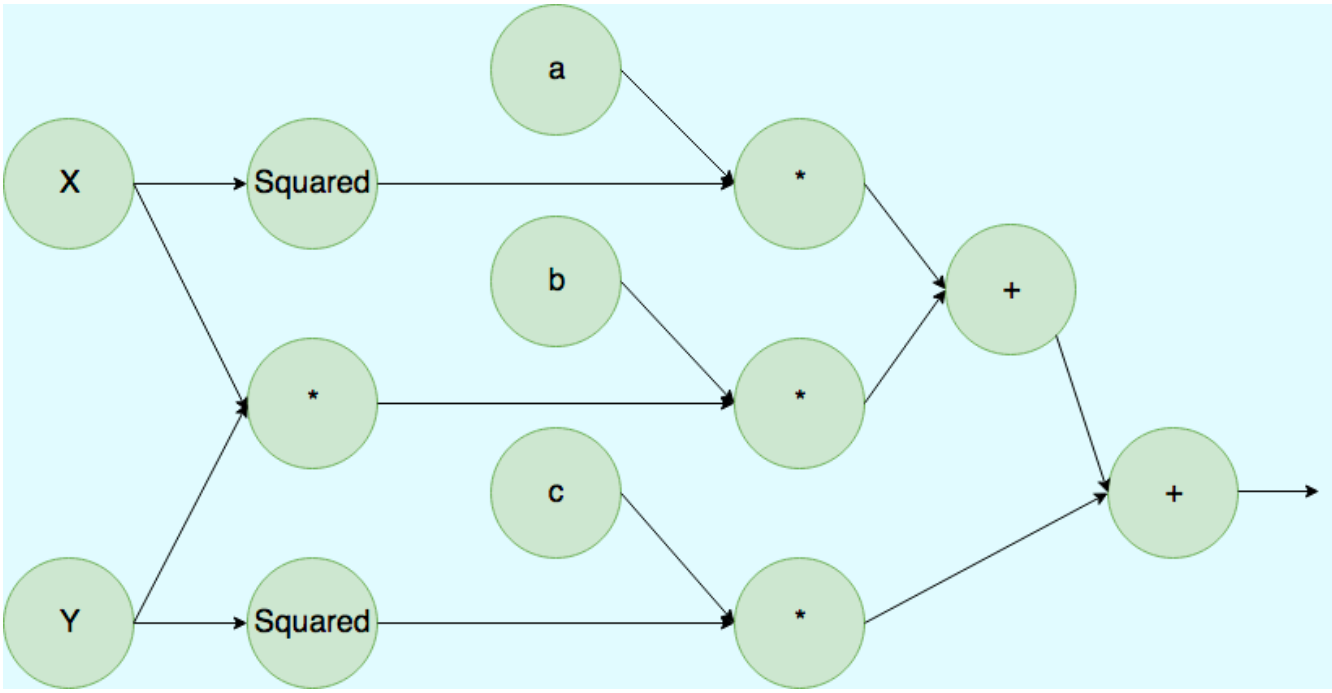
✔ The bytecode generated for any user function can not be inspected in human readable form

Run on a VM that executes the machine code corresponding to each byte code

Caches in pyc file for faster execution the second time the same file is executed

Question 11 2 / 2 points

Given the following computation graph, which of the following expressions does it evaluate to?



✔ ax^2+bxy+cy^2

(xy)^2+abc+ax+by

None of the above

(x)+abc+ax^2+by^2

Question 12 1 / 1 point

What is a primary benefit of using Python extension modules in Python development?

- ✓ ◉ They enhance Python's ability to interact with C/C++ libraries, allowing Python code to call C/C++ functions and manage memory more efficiently.
- ○ They enable Python programs to run entirely without the Python interpreter.
- ○ They provide a means to execute Python code at the speed of compiled C/C++ code by bypassing the Python Virtual Machine.
- ○ They allow for direct integration of Python code with JavaScript-based web applications.

Question 13 1 / 1 point

Which of the following are examples of Static Typed languages?

- ✓ ☑ C++
- ✓ ☐ Python
- ✓ ☑ C
- ✓ ☐ JavaScript

Question 14 1 / 1 point

Order the steps in which they occur in the Python Execution stages.

- ✓ __4__ Input is transformed into bytecode
- ✓ __3__ Input is transformed to Control Flow Graph (CFG)
- ✓ __2__ Input is transformed into Abstract Syntax Tree (AST)

✔ __1__ Input is transformed into parse tree

✔ __5__ Input is passed to peephole optimization

Question 15 2 / 2 points

Which of the following is a property of **Declarative** approach for creating computation graphs and which is a property of **Imperative** approach?

1. Not waiting for the full graph declaration - ____
2. Full graph declaration is required - ____
3. Better for optimization - ____
4. Keep adding components after initial declaration - ____
5. Computed using the computing engine - ____

Answer for blank # 1: Imperative ✔(20 %)

Answer for blank # 2: Declarative ✔(20 %)

Answer for blank # 3: Declarative ✔(20 %)

Answer for blank # 4: Imperative ✔(20 %)

Answer for blank # 5: Declarative ✔(20 %)

Question 16 1 / 1 point

Write **low**, **high**, and **medium** against the profling techniques based on their overhead

1. Counting - ____
2. Sampling - ____
3. Tracing - ____

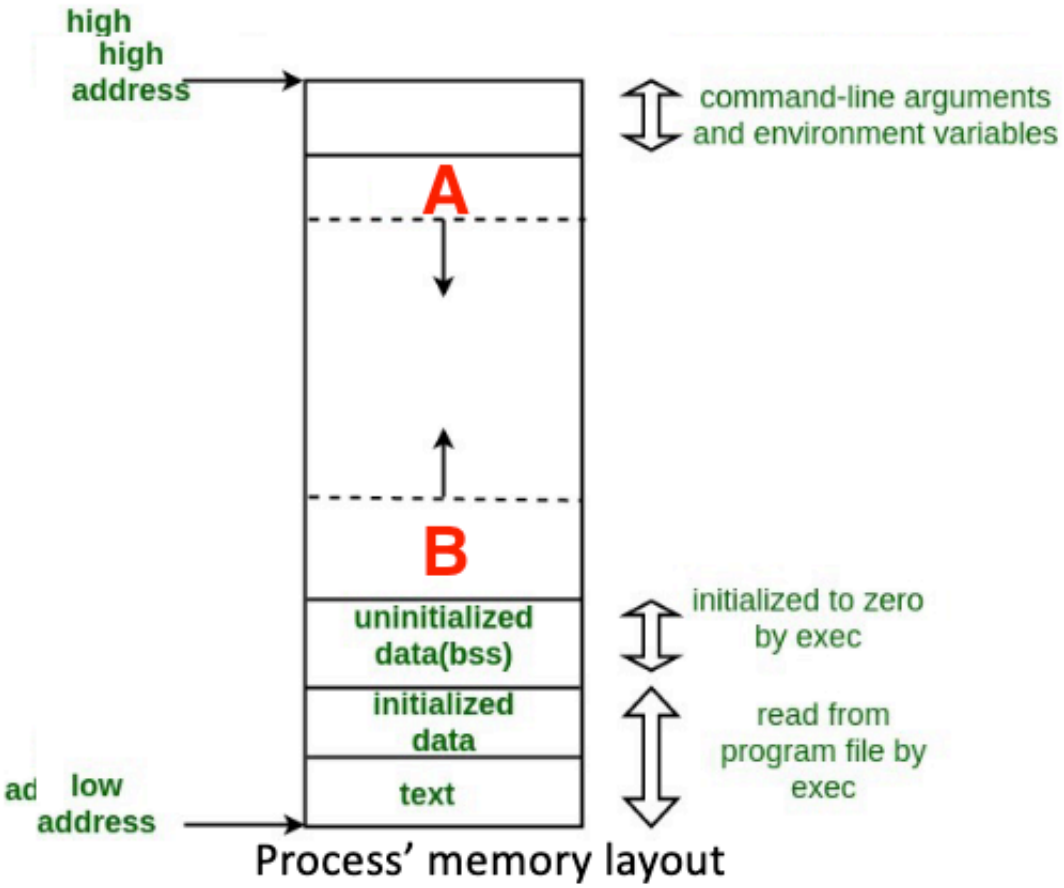
Answer for blank # 1: low ✔(33.33 %)

Answer for blank # 2: medium ✔(33.33 %)

Answer for blank # 3: high ✔(33.33 %)

Question 17 1 / 1 point

Refer the following figure and write down Stack or Heap in front of the blank:



A - ____

B - ____

Manually managed by C programs but automatically managed by Python - ____

Automatically/Externally managed for both C and Python Programs - ____

Answer for blank # 1: Stack ✔(25 %)

Answer for blank # 2: Heap ✔(25 %)

Answer for blank # 3: Heap ✔(25 %)

Answer for blank # 4: Stack ✔(25 %)

Question 18 1 / 2 points

You are evaluating the following expression: x - 2 * (m+n). Fill in the blanks with any subset made using the expression variables given (eg m+n or 2°m) and instructions used in the lecture slides [BINARY_MULTIPLY,BINARY _ADD].

COMPILED BYTECODE
LOAD_FAST 0
LOAD_FAST 1
LOAD_FAST 2
A
B
STORE_FAST 3

A - ____

B - ____

TOP of Stack State when IP is at A - ____

TOP of Stack State when IP is at B - ____

Answer for blank # 1:	BINARY_ADD	✔(25 %)
Answer for blank # 2:	BINARY_MULTIPLY	✔(25 %)
Answer for blank # 3:	x, m, n	✖ **(BINARY_ADD, m+n)**
Answer for blank # 4:	x, m+n	✖ **(BINARY_MULTIPLY, 2*(m+n))**

Question 19 1 / 1 point

Which of the following are advantages of Manual memory management?

- ✔☐ Code simplicity
- ✔☑ High Performance
- ✔☐ Lower risk of bugs
- ✔☑ Better code understanding by programmer

Done