The answers are for mid preparation. Also, these are just examples. Some questions can have alternative solutions.

1. How is a constructor defined in a Python class? Show an example. **(2 marks)**

   Answer:

   ```
   def __init__(self, id, name):
           self.id = id
           self.name = name
   ```

2. What is ADT? Give an example. **(2 marks)**

   Answer:

   Abstract Data type (ADT) is a type for objects whose behavior is defined by a set of values and a set of operations.

   Example: list, set, dictionary, stack, queue

3. What is the purpose of using DefaultDict in python? **(2 marks)**

   Answer:

   One type of Dictionary that handles missing values.

4. Suppose, you have implemented your stack class. You have 5 functions: isempty(), push(), pop(), peek() and size(). Write a function named '**remove_duplicates(input_string)**' using this stack class to remove all the adjacent (পাশাপাশি) duplicate characters from a given string. **(4 marks)**

   Example: remove_duplicates("aabcbtt") will return bcb.

   Answer:

   ```
   def remove_duplicates(input_string):
     s = Stack()
     for x in input_string:
       if not s.isempty() and s.data[-1] == x:
         s.pop()
       else:
         s.push(x)
     return s.data
   ```

1. Suppose you have to store the monthly income (Integer) of every person in Bangladesh. Which data structure will you use between python array and list and why? **(2 marks)**

   Answer:

   Python array.

   We are storing just integer here. **memory consumption** of arrays is more efficient than lists

2. Give an example of a Student class. (2 attributes will be enough) **(2 marks)**

   Answer:

   ```
   class Student:
           def __init__(self, id, name):
                   self.id = id
                   self.name = name
   ```

3. What will be the output of this given code? **(2 marks)**

   ```
   from collections import namedtuple
   Employee = namedtuple('abc', ['salary','employee_id'])
   e = Employee (100, 250)
   print(e.employee_id)
   print(e.abc)
   ```

   Answer:

   250

   It will give an error

4. Suppose, you have implemented your stack class. You have 5 functions: isempty(), push(), pop(), peek() and size(). Modify your pop function so that it can remove only the odd integers from the stack. We are assuming this stack has only integer elements. **(4 marks)**

   Example: If the stack has these elements: 20, 10, 25, 6, 7

   pop()

   pop()

   The first pop will remove 7. The second pop will not remove any element.

   Answer:

   ```
   def pop_odd(self):
         if self.isempty():
           print("Stack is empty")
         else:
           if self.peek()%2 != 0:
             self.data.pop()
   ```