AlgoritmaTraversal.py

```python
import os
os.system('cls')

class Node:
  def __init__(self, key):
    self.kiriChild = None
    self.kananChild = None
    self.data = key

# Function untuk Inorder Traversal
def InorderTraversal(root):
  if root:
    InorderTraversal(root.kiriChild)
    print(root.data, end=" ")
    InorderTraversal(root.kananChild)

# Function untuk Preorder Traversal
def PreorderTraversal(root):
  if root:
    print(root.data, end=" ")
    PreorderTraversal(root.kiriChild)
    PreorderTraversal(root.kananChild)

# Function untuk Postorder Traversal
def PostorderTraversal(root):
  if root:
    PostorderTraversal(root.kiriChild)
    PostorderTraversal(root.kananChild)
    print(root.data, end=" ")

if __name__ == '__main__':
  root = Node(1)
  # Subpohon kiri
  root.kiriChild = Node(2)
  root.kiriChild.kiriChild = Node(4)
  root.kiriChild.kananChild = Node(5)
  root.kiriChild.kiriChild.kiriChild = Node(8)
  # Subpohon kanan
  root.kananChild = Node(3)
  root.kananChild.kiriChild = Node(6)
  root.kananChild.kananChild = Node(7)
  root.kananChild.kiriChild.kiriChild = Node(9)
  root.kananChild.kiriChild.kananChild = Node(10)

  print("\nInorder Traversal of binary tree is")
  InorderTraversal(root)
  print()
  print("\nPreorder Traversal of binary tree is")
  PreorderTraversal(root)
  print()
  print("\nPreorder Traversal of binary tree is")
  PostorderTraversal(root)
```