**Axborot tizimlari va texnologiylari yo’nalishi**

**122-20-guruh talabasi**

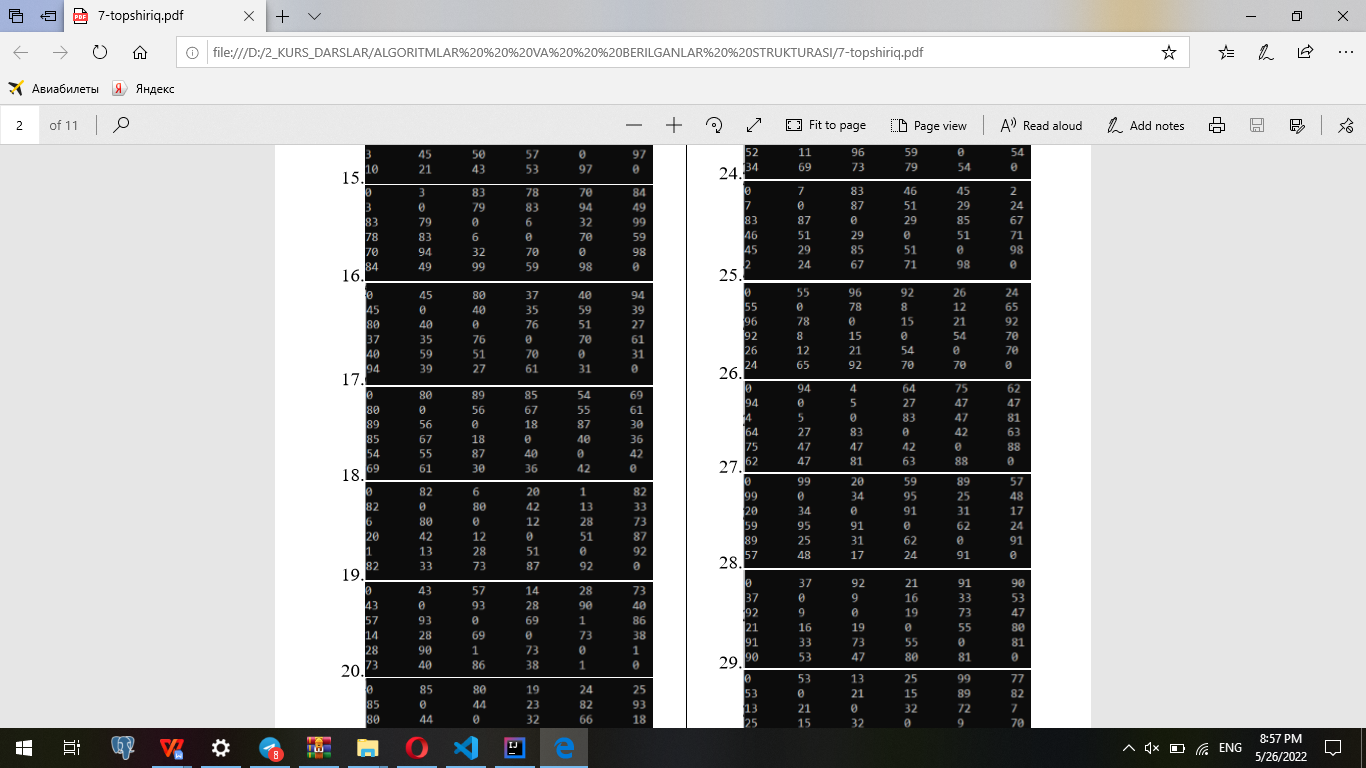
**Sattorova Mohiraning**

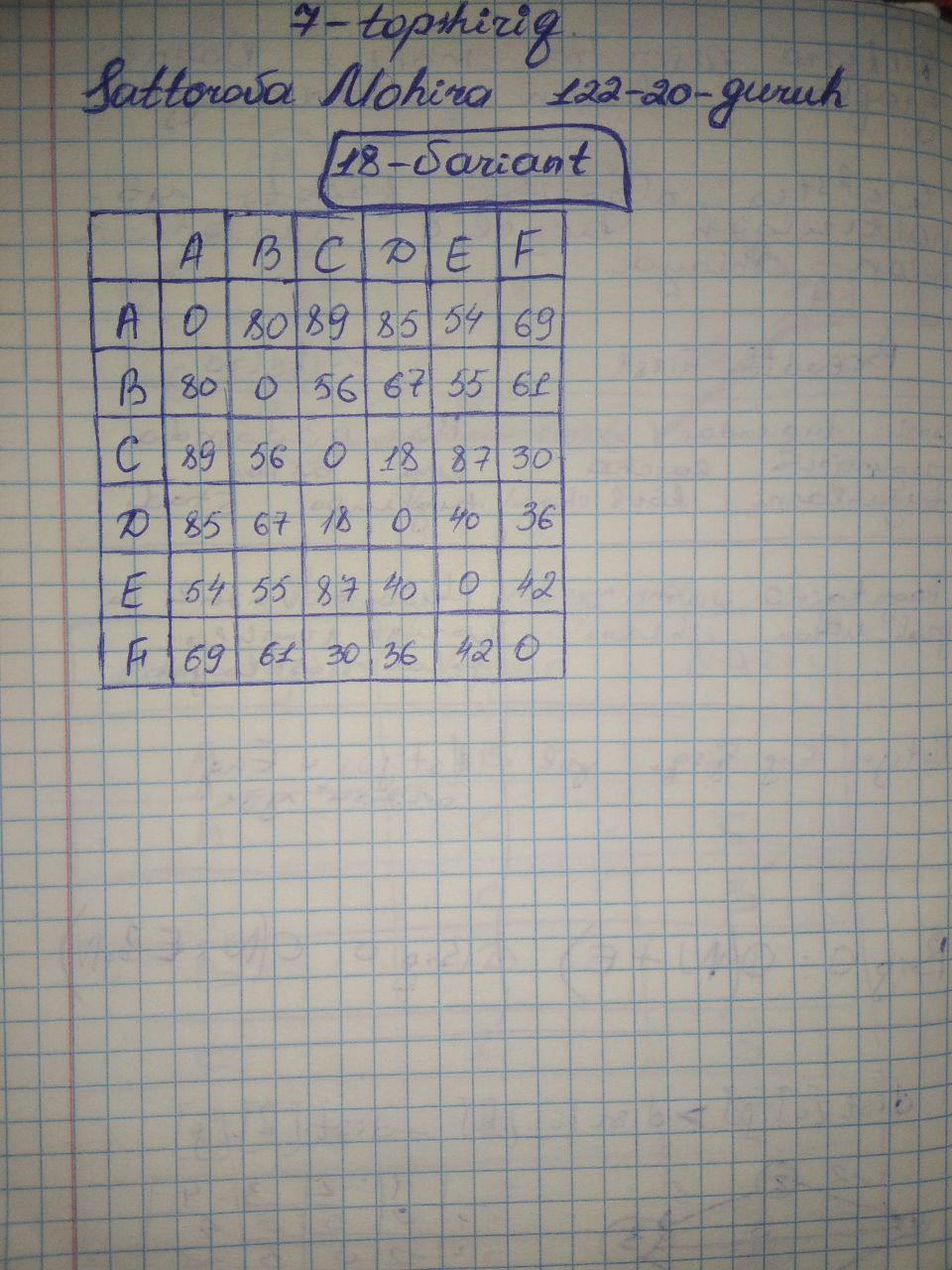
**Algoritmlar va berilganlar strukturasi fanidan**

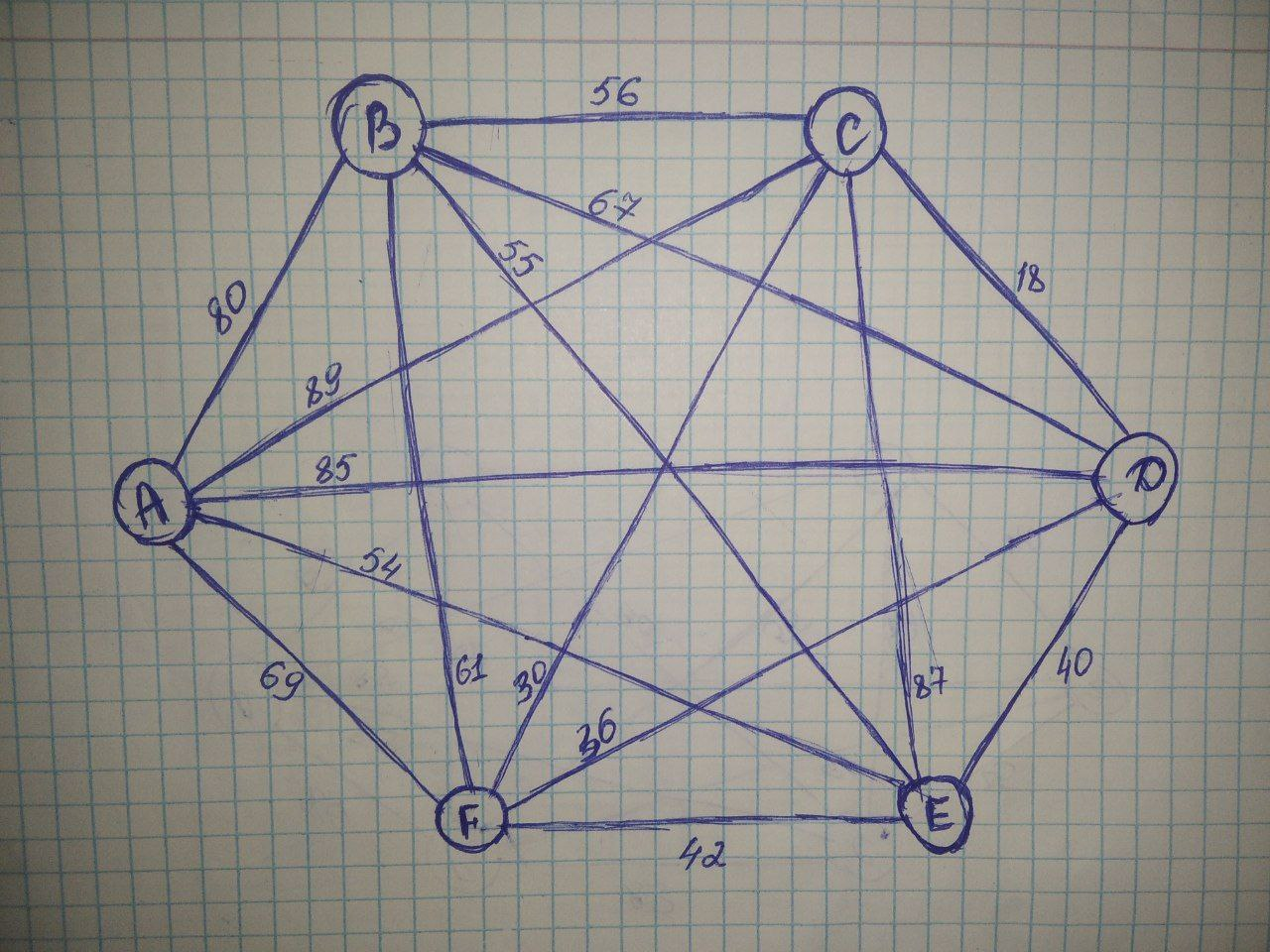
1. **topshirig’I**

**18-variant**

**7-topshiriq:Berilgan vaznli graf uchun Prim algoritmini qo’llab MST(Minimum Spanning Tree)ni hisoblang.**







class Graph:  
 def \_\_init\_\_(self, vertex):  
 self.V = vertex  
 self.graph = []  
  
 def add\_edge(self, u, v, w):  
 self.graph.append([u, v, w])  
  
 def search(self, parent, i):  
 if parent[i] == i:  
 return i  
 return self.search(parent, parent[i])  
  
 def apply\_union(self, parent, rank, x, y):  
 xroot = self.search(parent, x)  
 yroot = self.search(parent, y)  
 if rank[xroot] < rank[yroot]:  
 parent[xroot] = yroot  
 elif rank[xroot] > rank[yroot]:  
 parent[yroot] = xroot  
 else:  
 parent[yroot] = xroot  
 rank[xroot] += 1  
  
 def kruskal(self):  
 result = []  
 i, e = 0, 0  
 self.graph = sorted(self.graph, key=lambda item: item[2])  
 parent = []  
 rank = []  
 for node in range(self.V):  
 parent.append(node)  
 rank.append(0)  
 while e < self.V - 1:  
 u, v, w = self.graph[i]  
 i = i + 1  
 x = self.search(parent, u)  
 y = self.search(parent, v)  
 if x != y:  
 e = e + 1  
 result.append([u, v, w])  
 self.apply\_union(parent, rank, x, y)  
 for u, v, weight in result:  
 print("Edge:", u, v, end=" ")  
 print("-", weight)  
  
  
g = Graph(8)  
g.add\_edge(0, 1, 547)  
g.add\_edge(0, 2, 64)  
g.add\_edge(0, 3, 951)  
g.add\_edge(0, 4,818)  
g.add\_edge(0, 5,494)  
g.add\_edge(0, 6,219)  
g.add\_edge(0, 7,653)  
g.add\_edge(1, 2,548)  
g.add\_edge(1, 3,713)  
g.add\_edge(1, 4,632)  
g.add\_edge(1, 5,252)  
g.add\_edge(1, 6,283)  
g.add\_edge(1, 7,917)  
g.add\_edge(2, 3,206)  
g.add\_edge(2, 4,179)  
g.add\_edge(2, 5,272)  
g.add\_edge(2, 6,436)  
g.add\_edge(2, 7,208)  
g.add\_edge(3, 4,17)  
g.add\_edge(3, 5,540)  
g.add\_edge(3, 6,992)  
g.add\_edge(3,7,412)  
g.add\_edge(4,5,323)  
g.add\_edge(4,6,172)  
g.add\_edge(4,7,68)  
g.add\_edge(5,6,224)  
g.add\_edge(5,7,65)  
g.add\_edge(6,7,381)  
g.kruskal()

