№5 лаб

1-Код

pages = [1, 2, 3, 1, 4, 5, 1, 2, 3, 4, 5]

frames = []

frame\_count = 4

page\_faults = 0

print("Сұраныс | Жадтағы беттер | Page Fault")

for p in pages:

if p not in frames:

if len(frames) < frame\_count:

frames.append(p)

else:

frames.pop(0)

frames.append(p)

page\_faults += 1

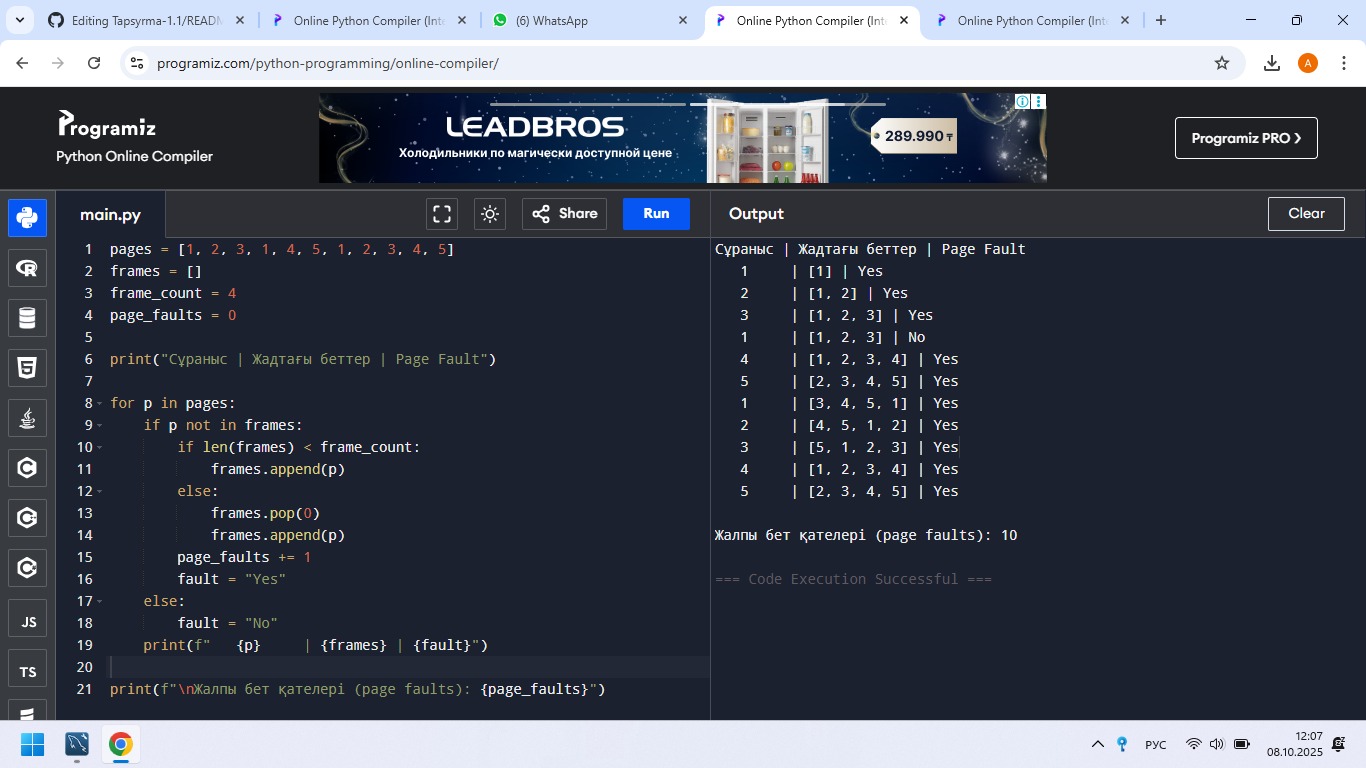
fault = "Yes"

else:

fault = "No"

print(f" {p} | {frames} | {fault}")

print(f"\nЖалпы бет қателері (page faults): {page\_faults}")



2-код

processes = ['P1', 'P2', 'P3', 'P4']

arrival = [0, 1, 2, 3]

burst = [8, 4, 9, 5]

quantum = 3

n = len(processes)

remaining = burst[:]

time = 0

waiting = [0]\*n

turnaround = [0]\*n

queue = []

while True:

done = True

for i in range(n):

if arrival[i] <= time and remaining[i] > 0:

done = False

if remaining[i] > quantum:

time += quantum

remaining[i] -= quantum

else:

time += remaining[i]

waiting[i] = time - burst[i] - arrival[i]

remaining[i] = 0

if done:

break

for i in range(n):

turnaround[i] = burst[i] + waiting[i]

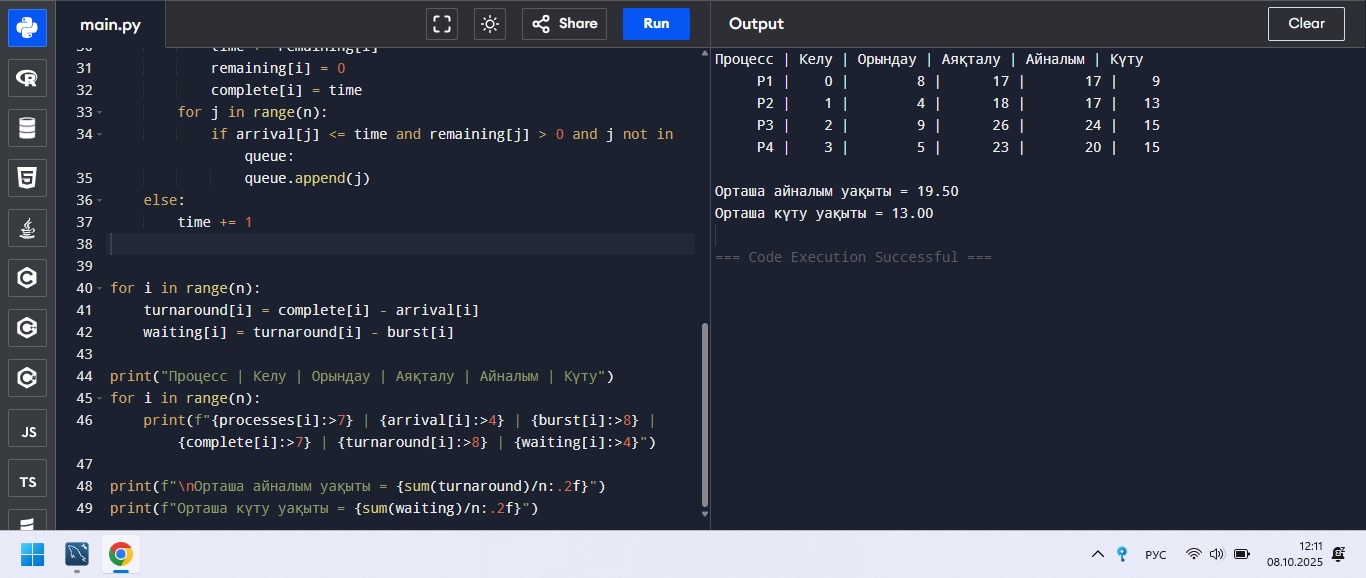
print("Процесс | Күту уақыты | Айналым уақыты")

for i in range(n):

print(f"{processes[i]} | {waiting[i]:>10} | {turnaround[i]:>14}")

print("\nОрташа күту уақыты =", sum(waiting)/n)

print("Орташа айналым уақыты =", sum(turnaround)/n)



2-код нәтижесі