**Міністерство освіти і науки України**

**Національний університет «Запорізька Політехніка»**

Кафедра програмних засобів

**ЗВІТ**

з контрольної роботи №1

з дисципліни «Моделювання та Аналіз Програмного Забезпечення»

**Виконав:**

Студент групи КНТ-122 О. А. Онищенко

**Прийняли:**

Викладач: Ж. К. Камінська

2024

Опис завдання

На АЗС є три колонки. На колонку #1, одразу ж з відкриттям АЗС, приїжджають вантажні автомобілі кожні 20 хвилин. На колонку #2 та #3 через 1 годину після відкриття легкові кожні 10 хвилин. Вантажні автомобілі обслуговуються - 25 хвилин, легкові - 20 хвилин і заправляють 30 літрів 92 бензини з оплатою 1290грн. Промоделювати роботу АЗС протягом 8 годин робочого дня, якщо легкові автомобілі стають на заправку тільки у випадку, якщо черга менша за 3 автомобілі.

Бо так полюбив Бог світ, що Він віддав Сина Свого Однородженого (Ісуса), аби кожен хто вірує в Нього, не згинув, але мав вічне життя. ([Йоан 3:16](https://www.bible.com/uk/bible/compare/JHN.3.16))

Код програми

#include "../simc/simc.h"

#include <iostream>

using namespace std;

void solve() {

  auto Truck\_Interval=20;

  auto Car\_Interval=10;

  auto Truck\_Delay=25;

  auto Car\_Delay=20;

  auto Car\_Gas\_Amount=30;

  auto Car\_Gas\_Type=92;

  auto Car\_Gas\_Price=1290;

  auto Modeling\_Hours=8;

  auto Modeling\_Time=Modeling\_Hours\*60;

  // cars go in if queue is less than 3

  pfacility Tank\_One;

  pfacility Tank\_Two;

  pfacility Tank\_Three;

  pqueue Queue\_One;

  pqueue Queue\_Two;

  pqueue Queue\_Three;

  initlist(Modeling\_Time);

  auto Truck\_Initial\_Delay=0;

  auto Car\_Initial\_Delay=1\*60;

  initcreate(1, Truck\_Initial\_Delay);

  initcreate(8, Car\_Initial\_Delay);

  initcreate(15, Car\_Initial\_Delay);

  newfac(Tank\_One, "\"Tank One (Truck)\"");

  newfac(Tank\_Two, "\"Tank Two (Car)\"");

  newfac(Tank\_Three, "\"Tank Three (Car)\"");

  newqueue(Queue\_One, "\"Queue One (Truck)\"");

  newqueue(Queue\_Two, "\"Queue Two (Car)\"");

  newqueue(Queue\_Three, "\"Queue Three (Car)\"");

  auto Total\_Liters=0;

  auto Total\_Cost=0;

  while (systime < Modeling\_Time) {

    plan();

    switch (sysevent) {

      case 1: create(Truck\_Interval); break;

      case 2: inqueue(Queue\_One); break;

      case 3: seize(Tank\_One); break;

      case 4: outqueue(Queue\_One); break;

      case 5: delayt(Truck\_Delay); break;

      case 6: outfac(Tank\_One); break;

      case 7: destroy(); break;

      case 8: create(Car\_Interval); break;

      case 9: if (Queue\_Two->lq < 3) inqueue(Queue\_Two); break;

      case 10: seize(Tank\_Two); break;

      case 11: outqueue(Queue\_Two); break;

      case 12: delayt(Car\_Delay); Total\_Liters += Car\_Gas\_Amount; Total\_Cost += Car\_Gas\_Price; break;

      case 13: outfac(Tank\_Two); break;

      case 14: destroy(); break;

      case 15: create(Car\_Interval); break;

      case 16: if (Queue\_Three->lq < 3) inqueue(Queue\_Three); break;

      case 17: seize(Tank\_Three); break;

      case 18: outqueue(Queue\_Three); break;

      case 19: delayt(Car\_Delay); Total\_Liters += Car\_Gas\_Amount; Total\_Cost += Car\_Gas\_Price; break;

      case 20: outfac(Tank\_Three); break;

      case 21: destroy(); break;

    }

  }

  cout << "Total Liters: " << Total\_Liters << " Total Cost: " << Total\_Cost << endl;

  printall();

}

int main() {

  solve();

  return 0;

}

Виконання програми

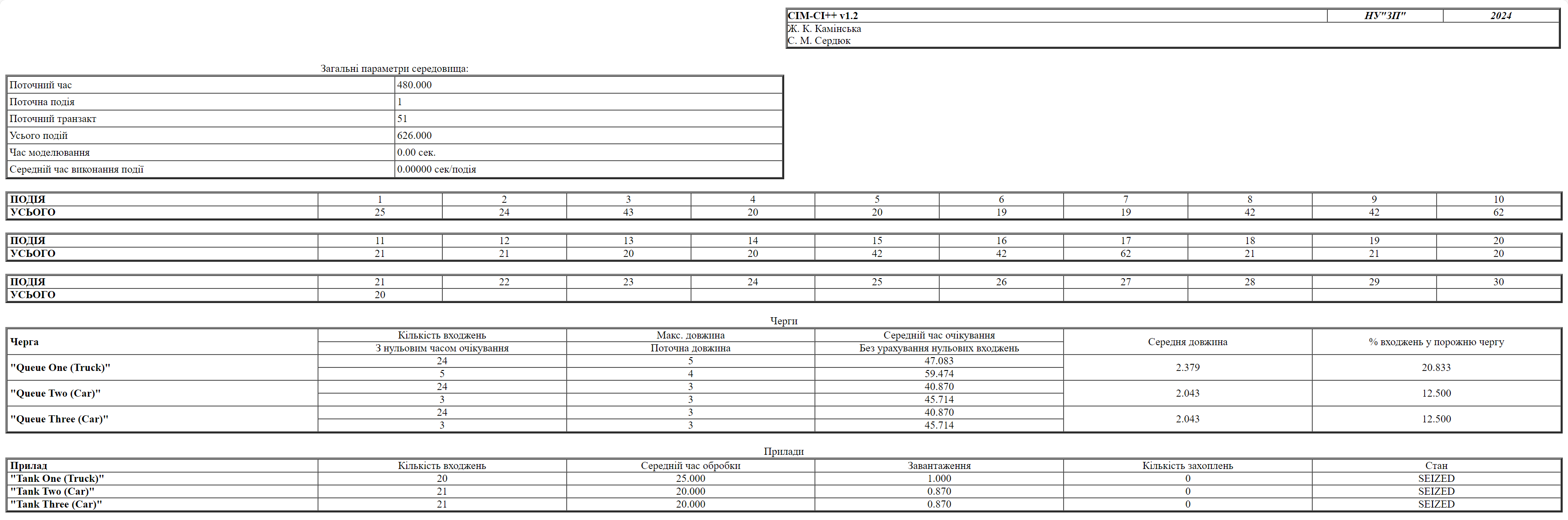


Рисунок 1.1 – Браузер



Рисунок 1.2 – Консолька ♰