

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. Промодельовати 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](#))*

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 (systemtime < 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;
    }
}

```

Виконання програми

Рисунок 1.1 – Браузер

Рисунок 1.2 – Консолька †