**Початок**

1. В методі Main() ініціалізується магазин та інтерфейс додатку. В конструкторі магазину ініціалізуються його стенди. в конструкторах кожного стенда ініціалізуються їхні продукти та продавці.
2. Користувач запускає роботу магазину натиснувши кнопку ЕНТЕР. Після цього запускається новий потік, в якому викликається метод магазину Open(). Після цього запускається цикл, який генерує покупців і працює поки не виникне подія isFinishOfWorkEventWaitHandle.

**Послідовність роботи продавця**

1. В методі Shop.Open() для кожного стенда викликається метод Stand.Open(). B Stand.Open() генерується подія StandOpen, яку обробляє кожний продавець в методі stand\_OnStandOpen.
2. B методі stand\_OnStandOpen в кожному продавцеві викликається метод DoWork, в якому запускається потік продавця і створюється помічник продавця, який також працює в новому потоці.
3. Продавець в циклі обробляє чергу стенда, якому він належить. При отриманні покупця з черги, для нього продавцем виконується метод ServeBuyer(). Після цього записується кількість проданих товарів, які зберігаються в стенді. Потім продавець передає даного покупця в чергу свого помічника, який послідовно для кожного покупця своєї черги викликає метод покупця DoWork(), який працює вже в потоці помічника.

**Послідовність роботи покупця**

1. Після запуску всіх стендів і продавців, починаються генеруватися покупці.
2. В конструкторі кожного покупця виконується його ініціалізація та викликається метод DoWork(), що спочатку запускається в новому потоці .
3. В цьому методі покупець шукає стенд, де найменша черга (обирає серед тих стендів, які ще він не відвідав), обирає кількість товару і стає в чергу поточного стенда. Після цього потік покупця завершує роботу. Далі роботу з покупцем буде продовжувати потік помічника.

**Закінчення**

1. Користувач завершує роботу другим натисненням кнопки ЕНТЕР. Після цього сигналізується подія isFinishOfWorkEventWaitHandle, після якої завершується цикл генерації покупців і викликає для магазину метод Close().
2. Тим часом головний потік програми чекає, поки завершить роботу магазин (поки магазин не просигналізує, що завершив роботу).
3. В методі shop.Close() магазин чекає, поки не залишиться активних покупців. Після цього для кожного стенда викликається метод Close(), де генерується подія CloseStand. Після цієї події, для покупців встановлюється прапор завершення роботи. Згідно з цього завершує роботу помічник покупця, а далі сам покупець.
4. Коли завершить роботу останній покупець стенда, тоді виводиться статистика продаж даного стенда. Як завершить роботу останній стенд, тоді магазин сигналізує про завершення своєї роботи.
5. Після цього сигналу, додаток виводить кількість усіх покупців і загальний прибуток магазину.

**Beginning**

1. The Main() method initializes a Shop instance and an application interface. The Shop constructor initializes its Stands. Each Stand initializes its Product and Sellers in constructor.
2. The User starts program work by Enter button. After that, the application runs the new thread, that calls method Shop.Open(). Then the application creates a loop, that is generating new Buyers until isFinishOfWorkEventWaitHandle is generated.

**Seller’s flow**

1. In Shop.Open() method, the application calls method Stand.Open() for each Stand. In this method, the StandOpen event generates. Each Seller of this stand handles this by stand\_OnStandOpen() method.
2. In stand\_OnStandOpen() method, each Seller calls a DoWork() method, in which application runs Seller’s thread and creates sellerHelper instance, that works in new thread.
3. Then the Seller gets some Buyer from queue. It serve Buyer (it calls method ServeBuyer()). After that, Seller writes information about number of sold products (this information is stored in Stand). Then the Seller sends a Buyer to helper’s queue. The sellerHelper gets this Buyer and calls for it a DoWork() method, that works in helper thread.

**Buyer’s flow**

1. After Shop opening, the application generates new Buyers in new thread.
2. The constructor of each Buyer does initialization of instance and calls a DoWork() method, that works in a new thread.
3. In this method, the Buyer chooses a Stand where is a smallest queue (it chooses among those Stand that it has not visited yet), chooses the number of products and takes the queue of selected Stand. After that, the Buyer’s thread ends work. Later the sellerHelper will continue Buyer’s work.

**Ending**

1. The User stops Shop’s work by Enter button. After that, isFinishOfWorkEventWaitHandle event is generated, generate loop is finished and shop.Close() method is called.
2. The application is waiting until Shop is finished work.
3. In the shop.Close() method, Shop is waiting until there are active buyers. After that, each Stand calls the Close() method, where the CloseStand event is generated. This event sets flag about finishing of work for all Sellers.
4. When the last Seller will finish a work, then the application shows statistic of Stand. If the last Stand will finish a work, then the Shop generates a WorkComplete event.
5. In the end, the application shows statistic about visitors and total profit.