Лабораторная №4

**Работа с очередями**

Заколесник Максим ИУ7-33

* **Цель работы:**

Получить навык работы с типом данных «очередь», представленным в виде одномерного массива и односвязного линейного списка. Провести сравнительный анализ реализации алгоритмов включения и удаления из очереди. Оценить эффективность реализаций.

* **Техническое задание:**

Система массового обслуживания состоит из двух обслуживающих аппаратов (ОА1 и ОА2) и двух очередей заявок. Всего в системе обращается **100** заявок.

Заявки поступают в "хвост" каждой очереди; в ОА они поступают из "головы" очереди по одной и обслуживаются по случайному закону за интервалы времени **Т1** и **Т2,** равномерно распределенные от **0 до 6** и от **1 до 8** единиц времени соответственно. (Все времена – **вещественного** типа). Каждая заявка после ОА1 c вероятностью **Р=0.7** вновь поступает в "хвост" первой очереди, совершая новый цикл обслуживания, а с вероятностью **1-Р** входит во вторую очередь. В начале процесса все заявки находятся в первой очереди.

Смоделировать процесс обслуживания до выхода из ОА2 первых 1000 заявок. Выдавать на экран после обслуживания в ОА2 каждых 100 заявок информацию о текущей и средней длине каждой очереди, а в конце процесса - общее время моделирования, время простоя ОА2, количество срабатываний ОА1, среднее времени пребывания заявок в очереди. Обеспечить по требованию пользователя с на экран адресов элементов очереди при удалении и добавлении элементов. Проследить, возникает ли при этом фрагментация памяти.

* **Входные данные:**

Два файла для выбора реализации очереди списком или массивом.

* **Выходные данные:**

Количество вышедших из ОА1 заявок;

Общее время моделирования;

Общее время простоя;

Среднее время пребывания заявок в очереди;

Погрешность;

Предусмотрено:

Выбор способа реализации

Проверка корректности ввода

Очистка очереди

Просмотр освобожденных адресов в случае работы со списком

Просмотр симуляции очереди

Просмотр сравнения эффективности реализаций

* **Аварийные ситуации:**

Переполнение очереди реализованной массивом

Ошибка выделения памяти

* **Целесообразность работы с очередью-массивом и очередью-списком**

Очередь-список:

+ Память ограничена только размером оперативной памяти, и постоянно перевыделяется столько раз, сколько нужно.

- На добавление/удаление элемента списка необходимо больше времени, причина - постоянное перевыделение памяти.

TIME COMPARISON

Process queue by array queue by list efficiency

push 76 354 466%

pop 32 323 994%

MEMORY COMPARISON

queue by array queue by list Efficiency

4000 32 12500%

Очередь-массив:

+ не затратно по времени

- размер массива ограничен

* **Анализ:**

Реальное время – все время работы моделирования очереди, оно оказалось приблизительно равно теоретическому.

Для данных при полном попадании во 2 очередь среднее время обработки OA2 есть 4.5 и происходит накапливание очереди.

Пои количестве вышедших из OA2 1000, примерное время работы:

1000 \* 4.5 примерно 4500

При иной вероятности попадания во 2 очередь зависимость несколько труднее

Погрешность:

Количество вышедших из 2 аппарата заявок, умноженное на среднее время обработки аппаратом есть суммарное время обработки вторым аппаратом при сравнении с практических и теоретических значений отклонение составляет около 2 процентов.

RESULT

Total time of processing = 4469.588641

Downtime = 5.227180

Number of entered OA1: 1597

ERRORs

Expected number of entered: 1116 (10.31%)

Expected total arrival time(2 way): 4504 (0.89%)

Для вероятности возврата 70%

Расчетное время MAX(1000 \* 3, 1000 \* (100 -70)%/100% \* 4.5) \* 100%/(100-70)% = 10000

RESULT

Total time of processing = 9825.530351

Downtime = 5239.433576

Number of entered OA1: 3307

ERRORs

Expected number of entered: 1147 (14.65%)

Expected total arrival time(2 way): 4500 (1.91%)

* **Анализ фрагментации**

Для реализации списком наличие фрагментации случайно. На одних машинах она есть, а на других ее нет, лично на моем устройстве ее наличие случайно.

* **Вывод:**

Реализации списком и массивом очереди имеют свои плюсы и минусы, выбор между ними определяется тем, что наиболее важно: эффективность по времени или эффективность по памяти. Реализация списком более затратна по времени, а реализация массивом более затратна по памяти.

* **Контрольные вопросы:**

1. **Что такое очередь?**

Очередь – это последовательный список переменной длины, включение элементов в который идет с одной стороны (с «хвоста»), а исключение – с другой стороны (с «головы»).

1. **Каким образом освобождается память при удалении элемента из очереди при ее различной реализации?**

При реализации в виде массива по кольцевому принципу удаляется элемент, находящийся по адресу Pout(голова очереди), а указатель Pout смещается к следующему элементу.

При моделировании очереди в виде списка память освобождается по указателю head(головы) списка, а указатель на новую голову принимает значение адреса следующего элемента списка.

1. **Что происходит с элементами очереди при ее просмотре?**

Ничего. Просмотр.

1. **Каким образом эффективнее реализовывать очередь. От чего это зависит?**

См. вывод.