

ТЕМА 2 ИСПОЛЬЗОВАНИЕ УКАЗАТЕЛЕЙ В ЯЗЫКЕ C#

2.1 Цель лабораторной работы

Получить практические навыки по объявлению указателей в консольных приложениях и их использованию в языке C#. Освоить операции с указателями при работе с массивами и другими структурами данных.

2.2 Методические указания к лабораторной работе

Изучить материал 2, 3 и 4 лекций дисциплины.

Изучить материал, рекомендуемый в силлабусе: стр. 347- 355 [2] – сокращенный вариант представлен:

«Небезопасный код

Одним из основных достоинств языка C# является его схема работы с памятью: автоматическое выделение памяти под объекты и автоматическая уборка мусора. При этом невозможно обратиться по несуществующему адресу памяти или выйти за границы массива, что делает программы более надежными и безопасными и исключает возможность появления целого класса ошибок, доставляющих массу неудобств при написании программ на других языках.

Однако в некоторых случаях возникает необходимость работать с адресами памяти непосредственно, например, при взаимодействии с операционной системой, написании драйверов или программ, время выполнения которых критично. Такую возможность предоставляет так называемый небезопасный (unsafe) код.

Небезопасным называется код, выполнение которого среда CLR не контролирует. Он работает напрямую с адресами областей памяти посредством указателей и должен быть явным образом помечен с помощью ключевого слова `unsafe`, которое определяет так называемый небезопасный контекст выполнения.

Ключевое слово `unsafe` может использоваться либо как спецификатор, либо как оператор. В первом случае его указывают наряду с другими спецификаторами при описании класса, делегата, структуры, метода, поля и т. д. — везде, где допустимы другие спецификаторы. Это определяет небезопасный контекст для описываемого элемента, например:

```
public unsafe struct Node
{
    public int Value;
    public Node* Left;
    public Node* Right;
}
```

Вся структура `Node` помечается как небезопасная, что делает возможным использование в ней указателей `Left` и `Right`. Можно применить и другой

вариант описания, в котором небезопасными объявляются только соответствующие поля структуры:

```
public struct Node
{
    public int Value;
    public unsafe Node* Left;
    public unsafe Node* Right;
}
```

Оператор unsafe имеет следующий синтаксис:

unsafe блок

Все операторы, входящие в блок, выполняются в небезопасном контексте.

ПРИМЕЧАНИЕ

Компиляция кода, содержащего небезопасные фрагменты, должна производиться с ключом /unsafe. Этот режим можно установить путем настройки среды Visual Studio (Project ► Properties ► Configuration Properties ► Build ► Allow Unsafe Code).»

2.3 Домашнее задание на лабораторную работу

Разместить в динамической памяти компьютера матрицу 5x5 случайных целых чисел в диапазоне минус 60 до 60. Напечатать ее. Предусмотреть оба варианта использования указателей для работы с массивами (один указатель на массив и массив указателей).

2.4 Индивидуальные задания для СРС

2.4.1 Очередь в столовую 100 человек представлена 4 категориями людей: студенты, преподаватели, школьники и сотрудники. Очередь формируется случайным образом с одинаковой вероятностью для каждой категории. Напечатать общую очередь. Разделить общую очередь на две очереди – для преподавателей и сотрудников, студентов и школьников. Напечатать обе очереди.

2.4.2 Создать стек 25 студентов, в каждый элемент которого включать номер студента и его оценку за экзамен. Оценки формировать случайным образом в диапазоне от 2 до 5. Напечатать стек. Разделить стек на два отдельных стека – «отличников» и «хорошистов», «троечников» и «двоечников». Напечатать оба стека.

2.4.3 Создать стек 20 случайных чисел в диапазоне от минус 0 до 100. Напечатать его. Поменять местами первый и последний элементы стека. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.4 Создать очередь 20 случайных чисел в диапазоне от минус 60 до 60. Напечатать ее. «Переписать» числа в новую очередь так, чтобы положительные числа разместились в начале очереди, а отрицательные числа – в конце очереди. Напечатать новую очередь.

2.4.5 Создать две очереди 10 случайных чисел в диапазоне от минус 40 до 40. Напечатать их. Поэлементно объединить их в одну очередь. Напечатать полученную очередь.

2.4.6 Список студентов группы (25 человек) организовать в виде стека. В записях элементов стека предусмотреть пол студента (М или Ж) – формировать случайным образом. Напечатать стек. Преобразовать этот стек в два по полу студентов. Напечатать новые стеки.

2.4.7 Создать стек 10 случайных чисел в диапазоне от минус 10 до 10. Напечатать его. Найти элемент стека с максимальным значением и поменять его с первым элементом. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.8 Создать очередь 20 случайных чисел в диапазоне от 0 до 100. Напечатать ее. «Переписать» числа в новую очередь так, чтобы все четные числа разместились в начале очереди, а нечетные числа – в конце очереди. Напечатать новую очередь.

2.4.9 Алгебраическое выражение (без скобок) типа $A*B+C*E+X$, введенное в режиме диалога, посимвольно записывается в очередь. При очистке очереди определить число слагаемых в выражении.

2.4.10 Алгебраическое выражение (без скобок) типа $A*B+C*E$, введенное в режиме диалога, посимвольно записать в стек. При очистке стека определить число сомножителей в выражении.

2.4.11 Создать стек 20 случайных чисел в диапазоне от минус 20 до 20. Напечатать его. Найти элемент стека с минимальным значением и поменять его с последним элементом. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.12 Создать стек 20 случайных чисел в диапазоне от минус 50 до 50. Напечатать его. Поменять местами 1 элемент с 20 и 10 элемент с 11. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.13 Магазин за 1 день посещает 100 человек. Обслуживание покупателей предполагает продажу им : хлеба – число от 0 до 24; сыра – число от 25 до 49; печенья – число от 50 до 74; пиво – число от 75 до 99 из случайного числа в диапазоне от 0 до 100. Организовать очередь покупателей, а при ее обслуживании определить и напечатать какой продукт покупается чаще.

2.4.14 Произвольный текст, введенный в режиме диалога, посимвольно записать в стек. При очистке стека определить и напечатать сколько раз встречалась каждая буква.

2.4.15 Создать стек 20 случайных чисел в диапазоне от минус 100 до 100. Напечатать его. Найти элемент стека с максимальным значением и поменять его с элементом, имеющим минимальное значение. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.16 Создать стек 20 случайных чисел в диапазоне от 0 до 99. Напечатать его. «Переписать» числа в новый стек так, что все числа из одной цифры

разместились в начале стека, а числа из двух цифр – в конце стека. Напечатать новый стек.

2.4.17 В автомобильной «пробке» застряло 200 авто – очередь. Марки авто: ВАЗ, Москвич и иномарка формируются случайным образом с равной вероятностью. При обслуживании очереди (ликвидации «пробки») определить марка каких авто представлена в очереди максимально.

2.4.18 Создать стек 20 студентов, в каждый элемент которого включить номер студента и его баллы за экзамен. Баллы формировать случайным образом в диапазоне от 0 до 100. Напечатать стек. При очистке стека определить и напечатать сколько студентов сдали экзамен на «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно».

2.4.19 Создать стек 30 случайных чисел в диапазоне от 0 до 100. Напечатать его. Найти элемент стека с максимальным значением и поменять его с элементом, имеющим следующее максимальное значение. Обмен выполнить с помощью указателей, а не «перезаписью» содержимого элементов. Напечатать новый стек.

2.4.20 Создать очередь 10 случайных чисел в диапазоне от 0 до 100. Напечатать ее. Выполнить сортировку чисел в очереди в порядке убывания. Напечатать новую очередь.

2.5 Контрольные вопросы для защиты отчета на СРСП

2.5.1 Существующие формы представления информации? Примеры.

2.5.2 Какая форма представления информации используется в компьютерах?

2.5.3 Какая система счисления используется в компьютерах?

2.5.4 Какая форма представления чисел используется в памяти компьютера?

Примеры.

2.5.5 Как записать шестнадцатеричное число ВС в двоичной и десятичной формах?

2.5.6 Понятие поразрядных логических операций? Примеры.

2.5.7 Операции побитового сдвига? Примеры.

2.5.8 Понятие указателя? Примеры.

2.5.9 Работа с небезопасным кодом. Примеры.

2.5.10 Операции с указателями. Примеры.