|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**

**«РАБОТА СО СТЕКОМ»**

Студент Светличная Алина Алексеевна

Группа ИУ7 – 33Б

Проверил Барышникова Марина Юрьевна

*2021 г.*

# **Описание технического задания**

Условие задачи:

Создать программу работы со стеком, выполняющую операции добавление, удаления элементов и вывод текущего состояния стека. Реализовать стек: а) массивом; б) списком. Все стандартные операции со стеком должны быть оформлены подпрограммами. При реализации стека списком в вывод текущего состояния стека добавить просмотр адресов элементов стека и создать свой список или массив свободных областей (адресов освобождаемых элементов) с выводом его на экран.

Пункты меню:

1 - Добавить элементы в стек

2 - Добавить случайные элементы в стек

3 - Удалить элементы из стека

4 - Вывести текущее состояние стека

5 - Проверить правильность расстановки скобок в выражении

0 - Выйти из программы

Входные данные:

* Способ реализации стека (массивом или списком)
* Номер пункта меню
* Входные данные зависят от пункта меню

Выходные данные:

Выходные данные зависят от пункта меню

Действие программы:

* Добавление элемента в стек
* Удаление элемента из стека
* Проверка правильности расстановки скобок в выражении

Обращение к программе:

Запускается через терминал командой ./main.exe в директории с программой.

Возможные аварийные ситуации:

1. Некорректный пункт меню
2. Некорректные символы при вводе
3. Отрицательное количество элементов в пунктах меню
4. Переполнение стека при добавлении
5. Выход за нижнюю границу стека при удалении
6. Невыделенная память

# **Описание структур данных**

**stack\_type** – переменная для выбора реализации стека

**action\_type** – переменная для выбора пункта меню

Поля структуры, описанной ниже, для реализации стека с помощью массива:

\*ptr – указатель на текущий элемент стека

size – количество заполненных элементов стека

**typedef struct stack\_array**

**{**

**char \*ptr;**

**int size;**

**} stack\_array\_t**

Поля структуры, описанной ниже, для реализации стека с помощью списка:

el – текущий элемент стека

num\_el – количество заполненных элементов стека

\*next – указатель на следующий элемент стека

**typedef struct el\_stack\_list**

**{**

**char el;**

**int num\_el;**

**struct el\_stack\_list \*next;**

**} el\_stack\_list\_t**

Поля структуры, описанной ниже, для хранения свободных областей стека:

\*\*array – динамическая матрица для хранения адресов свободных областей стека

size – количество свободных областей стека

**typedef struct array\_free\_areas**

**{**

**el\_stack\_list\_t \*\*array;**

**int size;**

**} array\_free\_areas\_t**

# **Описание алгоритма**

Стек-массив: указатель на начало массива, который передвигается вперед или назад при добавлении и удалении соответственно (также увеличивается или уменьшается количество элементов).

Стек-список: происходит работа с указателем на следующий элемент, который при добавлении сначала инициализируется, а потом двигается, а при удалении сначала очищается, а потом двигается.

Алгоритм задания по варианту: проверка правильности расстановки скобок

Изначально три флага, соответствующие форме скобок, выставляются в ноль, что означает правильность расставления таких скобок (если скобки не были встречены вообще, то они все равно расставлены верно). Далее обрабатывается каждый элемент, если он является “закрывающей” скобкой (стек обрабатывается с конца), то соответствующий флаг выставляется в открытую позицию (1), далее ожидается “закрытие” флага “открывающей” скобкой, но, если встречена “открывающая” скобка при “закрытом” флаге, это обозначает неправильное расставление скобок – цикл прекращается.

После цикла проверяются все флаги на “закрытость”.

**Тесты**

|  |  |  |
| --- | --- | --- |
| Аварийная ситуация | Код ошибки | Уведомление |
| Тип стека - k | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Тип стека - 5 | ERROR\_STACK\_TYPE | Ошибка: введен некорректный номер команды |
| Пункт меню - № | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Пункт меню - -1 | ERROR\_ACTION\_TYPE | Ошибка: введен некорректный номер команды |
| При добавлении элемента в стек: | | |
| Размер стека - 1000 | ERROR\_STACK\_OVERLOW | Ошибка: стек переполнен |
| Количество добавляемых элементов - we | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Количество добавляемых элементов - -3 | ERROR\_NUM\_ADD | Ошибка: введенное число не является натуральным |
| Количество добавляемых элементов - 1001 | ERROR\_NUM\_ADD | Ошибка: при добавлении такого числа элементов произойдет переполнение стека |
| Добавляемый элемент - \n | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| При удалении элемента из стека: | | |
| Размер стека - 0 | ERROR\_STACK\_EMPTY | Ошибка: cтек пуст |
| Количество удаляемых элементов - we | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Количество удаляемых элементов - -3 | ERROR\_NUM\_REMOVE | Ошибка: введенное число не является натуральным |
| Количество удаляемых элементов - 1001 | ERROR\_NUM\_ REMOVE | Ошибка: при удалении такого числа элементов произойдет выход за нижнюю границу стека |

**Обработка результатов**

**Методика исследования:** программно проводится 100 замеров одного действия для определения среднего значения выполнения действия.

При добавлении элемента: массивом тратится время только на создание элемента (для этого пункта написана отдельная функция добавления рандомных элементов, чтобы исключить человеческую погрешность), передвижение указателя и увеличение размерности стека. Списком тратится время еще и на выделение памяти под каждый элемент (остальные этапы остаются так же).

При удалении элемента: массивом тратится время только на передвижение указателя и уменьшение размерности стека. Списком тратится время еще и на освобождение памяти из-под каждого элемента (остальные этапы остаются так же).

При проверки расстановок скобок: обе реализациии используют одинаковый алгоритм проверки. Массивом тратится время на алгоритм, передвижение указателя (изменение размерности происходит единожды после цикла). Списком тратится время еще и на освобождение памяти из-под каждого элемента (остальные этапы остаются так же).

**Добавление элементов (в тактах)**

|  |  |  |  |
| --- | --- | --- | --- |
| Размер | Массив | Список | Отношение массива к списку (в разах) |
| 10 | 976 | 8686 | 8,9 |
| 100 | 17251 | 61744 | 3,6 |
| 500 | 36941 | 240336 | 6,5 |
| 1000 | 70553 | 321419 | 4,6 |

**Удаление элементов (в тактах)**

|  |  |  |  |
| --- | --- | --- | --- |
| Размер | Массив | Список | Отношение массива к списку (в разах) |
| 10 | 177 | 1159 | 6,5 |
| 100 | 817 | 22549 | 27,6 |
| 500 | 4354 | 48537 | 11,1 |
| 1000 | 9152 | 135412 | 14,8 |

**Проверка расставления скобок (в тактах)**

**\***результаты различных замеров могут сильно отличаться из-за сравнения до первой «ошибки»

|  |  |  |  |
| --- | --- | --- | --- |
| Размер | Массив | Список | Отношение массива к списку (в разах) |
| 10 | 20664 | 47182 | 2,3 |
| 100 | 30856 | 45458 | 1,5 |
| 500 | 23982 | 54376 | 2,3 |
| 1000 | 19304 | 54974 | 2,8 |

**Сравнение памяти (в байтах)**

|  |  |  |  |
| --- | --- | --- | --- |
| Размер | Массив | Список | Отношение списка к массиву (в разах) |
| 10 | 40016 | 160 | 250 |
| 100 | 40016 | 1600 | 25 |
| 500 | 40016 | 8000 | 5 |
| 1000 | 40016 | 16000 | 2,5 |

**Контрольные вопросы**

1. Что такое стек?

Стек – это последовательный список с переменной длиной, в котором включение и

исключение элементов происходит только с одной стороны – с его вершины. Стек

функционирует по принципу: последним пришел – первым ушел, Last In – First Out (LIFO).

2. Каким образом и сколько памяти выделяется под хранение стека при различной его реализации?

Если хранить стек как список, то память выделяется в куче, если хранить как массив — либо в куче, либо на стеке (зависит от того, динамически или статический массив используется).

Для каждого элемента стека-списка выделяется дополнительные 4 или 8 байт для хранения указателя на следующий элемент, однако хранения статического массива под стек изначально занимает гораздо больше памяти.

3. Каким образом освобождается память при удалении элемента стека при различной реализации стека?

Если хранить стек как список, то верхний элемент удаляется при помощи операции освобождения памяти для него и смещением указателя, который указывает на начало стека.

Если хранить стек как массив, то смещается только указатель на начало стека.

4. Что происходит с элементами стека при его просмотре?

Классическая реализация стека предполагает, что просмотреть содержимое стека без извлечения (удаления) его элементов невозможно.

5. Каким образом эффективнее реализовывать стек? От чего это зависит?

Основной причиной выбора типа стека является размерность стека, если размер стека известен заранее, то статический массив будет идеальным сочетанием и быстро действенности, и затрат по памяти, но если же размер стека абсолютно неизвестен, то статический массив обязывает нас выделять память со слишком большим излишком (до 250 раз), а выигрыш во времени оказывается менее существенным (максимум в 25 раз) и в таком случае используется список.

# **Вывод**

Если стек реализован статическим массивом, то добавление в него новых элементов будет происходить быстрее (от 3 до 8 раз), чем в стек, реализованный списком. Удаление элементов из массива происходит в от 6 до 20 раз быстрее, чем из списка. Все это связано с тем, что стеку-списку помимо значения элемента надо еще и обрабатывать указатель на следующий элемент (при добавлении создавать, а при удалении очищать).

Однако вариант хранения стека в виде списка выигрывает пред статическим массивом по занимаемой памяти, также стоит использовать стек-список, если не известен даже приблизительный размер стека.