

Билеты к экзамену по информатике

20.Б09-мм

26 декабря 2021 г.

Содержание

- 1 История создания методов структуризации данных. Цели и принципы структурной методологии. Сложность, присущая программному обеспечению. Оценки сложности алгоритмов и представления данных.

TEST

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2 Представление полиномов вектором коэффициентов и вектором значений. Оценки сложности основных операций. Связь двух видов представлений полиномов. Быстрое умножение полиномов — основные идеи и проблемы.

TEST2

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

3 Абстракция типа стек. Определение новых типов. Абстракция типа очередь. Обобщение типа стек и очередь. Пример реализации стека. Вычисление выражений в ОПЗ.

Предметная область:

Стек - одномерная однородная структура данных с "обычными" операциями над стеком (или абстрактный тип данных, реализованный по принципу LIFO (last-in-first-out)).

Примеры:

- Магазин оружия
- В старых деревенских магазинах была дощечка с гвоздем на который накалывались чеки (для удобства и для проверок)
- Детская игра "пирамидка"

Формальная спецификация:

Определение АД (абстрактного типа данных) **стек** состоит из трех частей:

- **Множества данных:**

Множество базовых данных - X

Дополнительные множества: $bool = \{true, false\}$

Множество стеков - обозначается $stack$ или, для краткости, буквой S .

- **Множество операций над стеками:**

$push: X \times S \rightarrow S$

$pop: S \rightarrow S$

$top: S \rightarrow X$

$clear: S \rightarrow S$

или

$clear: S \rightarrow \{nil\}$

$is_empty: S \rightarrow bool$

Возм и др опер - придумать – упр (извл группу эл, пометить все, ...)

- **Аксиомы:**

(огр целостности)

– не всегда легко сформулир

$\forall x \in X \forall s \in S \ top(push(x, s)) = x$

$\forall x \in X \ pop(push(x, s)) = s$

$\forall x \in X \forall s \in S \ is_empty(push(x, s)) = false$

$is_empty(nil) = true$

$\forall s \in S \ is_empty(clear(s)) = true$

Упр – доп опер (напр, взять верхние k элементов)

Прикладная программа:

Массивы (статические или динамические), динамические структуры данных и прочее.

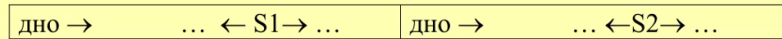
Определение новых типов данных на основе более простых (иерархия типов)

Имея некоторый базовый тип можно строить производные типы и реализовывать их различным образом.

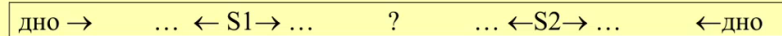
Производные типы

Два стека.

1) память



2) память



Предметная область:

Очередь - одномерная однородная структура данных с "обычными" операциями над очередью (или абстрактный тип данных, реализованный по принципу FIFO (first-in-first-out)).

Примеры:

- Очередь в кассу (идеализированная)
- Труба

Применение:

- В системах массового обслуживания для моделирования доступа к ресурсу
- В операционных системах для моделирования доступа к ресурсу
- Для организации каналов передачи данных

Определение АТД (абстрактного типа данных) **очередь** состоит из трех частей:

• Множества данных:

Множество простых/базовых данных обозначим X

Множество всех очередей обозначим $queue$ или Q

• Множество операций над стеками:

a. $push: X \times Q \rightarrow Q$ //...

b. $pop: Q \rightarrow Q$ //...

c. $top: Q \rightarrow X$ //...

d. $clear: Q \rightarrow Q$ или $clear: Q \rightarrow \{nil\}$ //...

e. $is_empty: Q \rightarrow bool$ //...

f. возм и др опер - придумать – упр
(извл группу эл, пометить все, ...)

- **Аксиомы:**

$$\text{Is_empty}(\text{nil}) = \text{true}$$

$$\forall q \in Q \text{ is_empty}(\text{clear}(Q)) = \text{true}$$

$$\forall x \in X, \forall q \in Q \text{ is_empty}(\text{push}(x, q)) = \text{false}$$

$$\forall x \in X, \forall q \in Q \text{ pop}(\text{push}(x, q)) \neq q$$

$$\forall x_1, x_2, x_3 \in X, q = (x_1, x_2)$$

$$\text{push}(x_3, q) = (x_1, x_2, x_3)$$

$$\text{top}(q) = x_1$$

$$\text{pop}(q) = (x_2, x_3)$$

Дек – абстрактный тип данных, являющийся обобщение очереди и стека – последовательность, открытая для изменения с двух сторон.

Пример реализации стека на языке программирования Java:

```
public interface Stack
// «верхний» ур абстр
{
    // int size = 100;           // часто с огр размера;
    public void push (int x); // м также опр метод, контролир размер стека
    public void pop();
    public int top();
}
// Stack
public class ПетинStack implements Stack // один вариант реализации
{
    public void push (int x) { ... }
    public void pop() { ... }
    public int top() { ... }
    ...
}
// ПетинStack
public class ВасинStack implements Stack // другой вариант реализации
{
    public void push (int x) { ... }
    public void pop() { ... }
    public int top() { ... }
    ...
}
// ВасинStack
```

Вычисление выражений в ОПЗ

<https://habr.com/ru/post/100869/>

2. Вычисление значения арифм выр, заданного в обратной польской записи, на уровне абстракции ТД стек

Пример.

$$(2-3) * (7+6) \rightarrow 2\ 3 - 7\ 6 + *$$

I

Выр м сод знаки бин ар опер + , - , * , / .

Вычисление производится за один просмотр выражения
слева направо с использованием АД стек.

Решаем задачу на уровне специфик стека и
задачи вычисл арифм выр.
без упомин особенн реализации

Пусть выражение также хран в виде стека $f \in \text{stack}$
(м и список).

Предп, что выр. правильно построено.

К спецификации стека добавим спецификацию алгоритма.

Переменные и типы данных

Data –
мн базовых данных, напр цел или вещ. (пока не уточн)

Oper –
мн (кодов) операций (напр, бин ар опер + , - , * , /) – тоже над базовыми дан

$x, a, b, c \in \text{Data} \cup \text{Oper}$ – перем,
способные принимать зн эл-тов стека/списка (базовые данные).

$s \in \text{stack}$ –
исп для хран промеж данных

$f \in \text{stack}$ –
хранит исх выр

Операции:

$\text{apply}(x, a, b) = \text{«значение 'x'(a,b)»}$ -

$$\text{apply}(' + ', 2, 3) = \text{значение ' + '(2,3)} = 2 + 3 = 5.$$

Здесь реализ идею нумерации счетной посл-ти пр-м (моделей ф-й),
поскольку программы –

«Программа»

```
1: if is_empty(f) then exit;  
x := top(f): pop(f);  
                                     // top+pop vs. pop  
if   x ∈ Data  
then push(x,s);  
else (a := top(s); pop(s); b := top(s); pop(s);  
      c := apply(x, a, b); push(c, s);  
      );  
goto 1;
```

Зам. Как использовался АДД стек

- 4 Деревья, их представление, примеры, сложность представления. Представление деревьев массивами. Бинарные деревья. Основные операции над ними. Применение стека.