Започнат на	Tuesday, 28 December 2021, 16:46
Състояние	Завършен
Приключен на	Tuesday, 28 December 2021, 16:55
Изминало време	8 мин. 11 сек.

Въпрос **1**

Правилен отговор

От максимално 1,00

Искаме да реализираме опашка (queue) чрез свързано представяне, използващо двойни кутии от вида:

```
struct Element {
   int Data;
   Element* pNext;
};
```

Ако поддържаме само указател към първия елемент в опашката, но нямаме указател към последния ѝ елемент, вярно ли е, че можем да я реализираме така, че както добавянето на елемент (enqueue), така и изваждането на елемент (dequeue) да бъдат със сложност O(1)?

Изберете едно:

- Истина
- Лъжа ✔

Въпрос **2**

Правилен отговор

От максимално 1,00

Нека е даден **двусвързан** списък, който съдържа N-елемента. Считаме, че представянето е такова, че разполагаме с указатели към първата и последната кутия в списъка.

Каква ще бъде сложността на всяка от дадените по-долу операции със списъка?

Забележка: В отговорите със знак ^ е обозначена операцията степенуване. По-конкретно:

- O(N^2) *обозначава сложността* $O(N^2)$
- O(2^N) *обозначава сложността* $O(2^N)$

Изтриване на първия елемент на списъка	O(1) ✔
Вмъкване на елемент на произволна позиция в списъка	O(N) ✔
Изтриване на последния елемент на списъка	O(1) ✔
Вмъкване на елемент на първа позиция в списъка	O(1) ✔
Проверка дали даден елемент се съдържа в списъка	O(N) ✔
Намиране на максималния елемент в списъка	O(N) ✔
Конкатениране на два списъка	O(1) ✔
Вмъкване на елемент на последна позиция в списъка	O(1) ✔
Сливане на два сортирани списъка	O(N) ✔

Въпрос **3**

Правилен отговор

От максимално 1,00

Нека е дадено следното свързано представяне на стек:

```
// двойна кутия в стека
struct Box {
   int Data;   // текущ елемент
   Box* pNext; // следващ елемент
};

// представяне на стека
struct Stack {
   Box* pTop;   // указател към върха на стека
   size_t Size; // брой на елементите в стека
};
```

Вярно ли е, че при това представяне можем да реализираме всички стандартни операции на стека (Push, Pop, Peek, GetSize) със сложност O(1)?

Изберете едно:

- (●) Истина ✔
- ( ) Лъжа

Въпрос **4**

Частично правилен отговор

От максимално 1,00

Кои от операциите **не е необходимо** да се имплементират за пълнофункционален стек?

Изберете едно или повече:

- a. isEmpty
- b. isFull ✖
- c. getSize ✔
- d. print ✔
- e. pop
- f. push
- g. top

Въпрос **5**

Правилен отговор

От максимално 1,00

Нека в опашка (queue) поред сме добавили числата 10, 20, 30, 40 и накрая 50. Можем ли да извадим числото 30 от опашката?

Изберете едно

- a. Да, но най-напред трябва да извадим 10 и след него 20. ✔
- b. Не, няма начин да извадим 30 от опашката.
- c. Да, но най-напред трябва да извадим 50 и след него 40.
- d. Да, можем директно да го извадим.

Въпрос **6**

Правилен отговор

От максимално 1,00

Нека е дадена опашка, която съхранява елементите си в динамичен масив. Тя е реализирана по стандартната схема, при която ако запълним масива до неговия край, а в началото му има свободно място, "превъртаме" края на опашката така, че запълването да продължи използвайки свободните позиции. Знаем, че при тази схема е възможно индексът на края на опашката да е по-малък от индекса на нейното начало.

Такава опашка е показана на дадената по-долу схема. Тя съдържа числата от 10 до 60. Над числата с 'H' (head) е обозначена главата, а с 'T' (tail) - опашката. Със символ '--' са указани празните клетки в масива.

	T			H			
50	60	--	--	10	20	30	40

Ако такава опашка се препълни, можем ли да преоразмерим масива? Ако това е възможно, каква ще бъде сложността на добавянето на елемент в общия случай?

Изберете едно

a. Да, може. Сложността на добавянето ще бъде O(N).

b. Не може, защото няма как да се направи копиране на елементите, без да се изгуби техният ред в опашката.

c. Да, може. Сложността на добавянето ще бъде O(1).

d. Да, може. Сложността на добавянето ще бъде амортизирано O(1). ✔

Въпрос **7**

Правилен отговор

От максимално 1,00

Динамичният масив (вектор) винаги изисква по-малко памет от едносвързания списък.

Изберете едно:

Истина

Лъжа ✔

Въпрос **8**

Правилен отговор

От максимално 1,00

Коя от структурите данни ще използва повече памет, за да съхрани N елемента от тип int?

Изберете едно

a. Динамичен масив

b. Свързан списък ✔

c. И двете ще използват еднакво количество памет

Въпрос **9**

Правилен отговор

От максимално 1,00

Какво прави операцията Peek, наричана още Top, за стек (stack)?

Изберете едно

a. Връща масив от всички елементи в стека, подредени от най-горния към най-долния.

b. За стековете няма такава операция

c. Връща стойността на или дава достъп до елемента, който се намира най-отдолу на стека.

d. Връща масив от всички елементи в стека, подредени от най-долния към най-горния.

e. Връща стойността на или дава достъп до елемента, който се намира най-отгоре на стека. ✔

Въпрос **10**

Правилен отговор

От максимално 1,00

Нека е даден **сортиран** масив, който съдържа N-елемента.

Каква ще бъде сложността на всяка от дадените по-долу операции с масива?

Забележка: В списъка с отговорите със знак ^ е обозначена операцията степенуване. По-конкретно:

- O(N^2) *обозначава сложността* $O(N^2)$
- O(2^N) *обозначава сложността* $O(2^N)$

Извличане на стойността на елемент намиращ се на даден индекс *k* в масива	O(1) ✔
Вмъкване на елемент на произволна позиция	O(N) ✔
Проверка дали даден елемент се съдържа в масива (с изчерпващо търсене)	O(N) ✔
Намиране на максималния елемент в масива	O(1) ✔
Проверка дали даден елемент се съдържа в масива (с двоично търсене)	O(log(N)) ✔
Добавяне на елемент на първа позиция в масива	O(N) ✔

Въпрос **11**

Правилен отговор

От максимално 1,00

При реализация на структура от данни, която ще изисква често добавяне и премахване на елементи на произволна позиция, двусвързания списък е за предпочитане пред динамичен масив при голям размер.

Изберете едно:

Истина ✔

Лъжа

Въпрос **12**

Правилен отговор

От максимално 1,00

Извличането на елемент от свързана опашка е идентично като реализация с извличането от свързан стек.

Изберете едно:

- Истина ✔
- Лъжа

Въпрос **13**

Правилен отговор

От максимално 1,00

Кои от операциите **не е необходимо** да се имплементират за пълнофункционална опашка?

Изберете едно или повече:

- a. operator == ✔
- b. dequeue
- c. isFull ✔
- d. getSize ✔
- e. print ✔
- f. getFront
- g. enqueue
- h. contains ✔

Въпрос **14**

Правилен отговор

От максимално 1,00

Разширяването на динамична, последователна опашка при добавяне е идентично с разширяването на динамичен масив?

Изберете едно:

- Истина
- Лъжа ✔

Въпрос **15**

Правилен отговор

От максимално 1,00

Ако при имплементиране на свързана опашка използваме само един указател (към началото на свързаната последователност) имплементирането на коя операция не може да стане според дефиницията на опашка?

Изберете едно

- a. isEmpty
- b. dequeue
- c. enqueue ✔
- d. Не може да се имплементира нито една операция
- e. Няма такъв проблем
- f. print
- g. getFront

Въпрос **16**

Правилен отговор

От максимално 1,00

При кои от следните операции свързаният списък има предимство пред динамичния масив:

- a. Търсене на елемент
- b. Изтриване на произволен елемент ✔
- c. Конкатенация на два списъка / масива ✔
- d. Добавяне на елемент в края
- e. Премахване на елемент от началото ✔
- f. Премахване на определени елементи (операция filter)
- g. Премахване на елемент в края
- h. Пространствена локалност
- i. Добавяне на елемент в началото ✔

Въпрос **17**

Правилен отговор

От максимално 1,00

Кое от следните понякога се реализира като адаптер чрез останалите две?

Изберете едно

- a. Стек ✔
- b. Свързан списък
- c. Динамичен масив
- d. Нито едно от посочените не може да се реализира така

Въпрос **18**

Правилен отговор

От максимално 1,00

Искаме да реализираме стек (stack) чрез свързано представяне, използващо структури от вида:

```
struct Element {
   int Data;
   Element* pNext;
};
```

Вярно ли е, че можем да реализираме стека така, че както добавянето на елемент (push), така и изваждането на елемент (pop) да бъдат със сложност O(1)?

Изберете едно:

- Истина ✔
- Лъжа

Въпрос **19**

Правилен отговор

От максимално 1,00

Нека е даден **едносвързан** списък, който съдържа N-елемента. Считаме, че представянето е такова, че разполагаме с указатели към първата и последната кутия в списъка.

Каква ще бъде сложността на всяка от дадените по-долу операции със списъка?

Забележка: В отговорите със знак ^ е обозначена операцията степенуване. По-конкретно:

- O(N^2) *обозначава сложността* $O(N^2)$
- O(2^N) *обозначава сложността* $O(2^N)$

Вмъкване на елемент на първа позиция в списъка	O(1) ✔
Проверка дали даден елемент се съдържа в списъка	O(N) ✔
Изтриване на последния елемент на списъка	O(N) ✔
Намиране на максималния елемент в списъка	O(N) ✔
Вмъкване на елемент на последна позиция в списъка	O(1) ✔
Вмъкване на елемент на произволна позиция в списъка	O(N) ✔
Изтриване на първия елемент на списъка	O(1) ✔

◄ Избор на проект

Отиди на ...

Записи от лекциите на ИС ►