Bài tập Stack Queue

Mục lục

[I. Câu Hỏi Ôn Tập 1](#_Toc514084290)

[Bài 1 1](#_Toc514084291)

[Bài 4 1](#_Toc514084292)

[Bài 5 3](#_Toc514084293)

[Bài 6 4](#_Toc514084294)

[Bài 7a 6](#_Toc514084295)

[Bài 7b 9](#_Toc514084296)

[II. Bài tập 10](#_Toc514084297)

[Bài 1a 10](#_Toc514084298)

[Bài 1b 12](#_Toc514084299)

[Bài 2 14](#_Toc514084300)

[Bài 3 16](#_Toc514084301)

[Bài 4 19](#_Toc514084302)

[Bài 5 20](#_Toc514084303)

1. Câu Hỏi Ôn Tập
   1. Bài 1
      1. Input:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| E | A | ~~S~~ |  |  |  |  |  |  |  |  |
| S |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| E | ~~A~~ | ~~Y~~ |  |  |  |  |  |  |  |
| S | Y | A |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| E | ~~Q~~ | ~~U~~ | ~~E~~ |  |  |  |  |  |  |
| S | Y | A | E | U | Q |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ~~E~~ | ~~S~~ | ~~T~~ |  |  |  |  |  |  |  |
| S | Y | A | E | U | Q | T | S | E |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ~~I~~ |  |  |  |  |  |  |  |  |  |
| S | Y | A | E | U | Q | T | S | E | I |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| O | N |  |  |  |  |  |  |  |  |
| S | Y | A | E | U | Q | T | S | E | I |

* 1. Bài 4

Lưu vị trí phần tử chẵn là phần tử đầu tiên

Duyệt từ đầu tới cuối danh sách:

+ Tại vị trí lẻ bỏ qua

+ Tại vị trí chẵn đưa phần tử về đứng sau vị trí được lưu, lưu thành vị trí mới, tiếp tục duyệt.

Push(5)

|  |
| --- |
| 5 |

Push(3)

|  |  |
| --- | --- |
| 5 | 3 |

Pop()

|  |
| --- |
| 5 |

Push(2)

|  |  |
| --- | --- |
| 5 | 2 |

Push(8)

|  |  |  |
| --- | --- | --- |
| 5 | 2 | 8 |

Pop()

|  |  |
| --- | --- |
| 5 | 2 |

Pop()

|  |
| --- |
| 5 |

Pop()

Push(9)

|  |
| --- |
| 9 |

Push(1)

|  |  |
| --- | --- |
| 9 | 1 |

Pop()

|  |
| --- |
| 9 |

Push(7)

|  |  |
| --- | --- |
| 9 | 7 |

Push(6)

|  |  |  |
| --- | --- | --- |
| 9 | 7 | 6 |

Pop()

|  |  |
| --- | --- |
| 9 | 7 |

Pop()

|  |
| --- |
| 9 |

Push(4)

|  |  |
| --- | --- |
| 9 | 4 |

Pop()

|  |
| --- |
| 9 |

Pop()

Rỗng.

* 1. Bài 5

Enq(5)

|  |
| --- |
| 5 |

Enq(3)

|  |  |
| --- | --- |
| 5 | 3 |

Deq()

|  |
| --- |
| 3 |

Enq(2)

|  |  |
| --- | --- |
| 3 | 2 |

Enq(8)

|  |  |  |
| --- | --- | --- |
| 3 | 2 | 8 |

Deq()

|  |  |
| --- | --- |
| 2 | 8 |

Enq(9)

|  |  |  |
| --- | --- | --- |
| 2 | 8 | 9 |

Enq(1)

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | 8 | 9 | 1 |

Deq()

|  |  |  |
| --- | --- | --- |
| 8 | 9 | 1 |

Enq(7)

|  |  |  |  |
| --- | --- | --- | --- |
| 8 | 9 | 1 | 7 |

Enq(6)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 1 | 7 | 6 |

Deq()

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | 1 | 7 | 6 |

Deq()

|  |  |  |
| --- | --- | --- |
| 1 | 7 | 6 |

Enq(4)

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 7 | 6 | 4 |

Deq()

|  |  |  |
| --- | --- | --- |
| 7 | 6 | 4 |

Deq()

|  |  |
| --- | --- |
| 6 | 4 |

* 1. Bài 6

|  |  |  |
| --- | --- | --- |
| Biểu thức trung tố | Cây biểu diễn | Biểu thức hậu tố |
| a – b \* c ^ d + f |  | a b c d ^ \* - f + |
| 1 + 2 +3 \* 4 + 5 – 6 \* 7 + 8 |  | 1 2 + 3 4 \* + 5 + 6 7 \* - 8 + |

**1. Nhận xét:**

Việc chuyển từ biểu thức trung tố sang hậu tố, thực chất là duyệt theo thứ tự Left-Right-Mid đệ quy trên cây biểu diễn.

**2. Dùng stack chuyển:**  
***Bước 1:*** Khởi tạo độ ưu tiên cho các toán tử:

|  |  |
| --- | --- |
| Toán tử | Độ ưu tiên |
| + | 1 |
| - | 1 |
| \* | 2 |
| / | 2 |
| ^ | 3 |

***Bước 2:*** duyệt biểu thức trung tố:

Với kí tự thứ i của biểu thức trung tố (giả sử đã loại hết kí tự trắng thừa):

* Nếu thuộc [0..9] hoặc [a..z], đẩy vào vector kết quả:
* Nếu là các toán tử thì:
  + B1: **Trong khi** stack khác rỗng **và** toán tử đỉnh stack có độ ưu tiên lớn hơn hoặc bằng toán tử đang xét thì:
    - Đẩy 1 phần tử ở đỉnh stack vào vector kết quả;
    - Lấy 1 phần tử ở đỉnh stack ra;
  + B2: Đẩy toán tử đang xét vào stack;

***Bước 3:*** Kiểm tra stack nếu khác rỗng thì đẩy tất cả những phần tử còn lại vào vector kết quả;

**[>** Đáp án biểu thức hậu tố chính là vector kết quả <**]**

**3. Ví dụ:**

|  |  |  |
| --- | --- | --- |
| a  -  b  \*  c  ^  d  +  f  0  1  2  3  4  5  6  7  8  **Biểu thức trung tố:** | | |
| Lấy phần tử: | Stack | Vector kết quả |
| A[0] = ‘a’ | Top  **STACK (blank):** | a |
| A[1] = ‘-‘ | (-, 1)  Top  **STACK (push):** | a |
| A[2] = ‘b’ | (-, 1)  Top  **STACK:** | a b |
| A[3] = ‘\*’ | (\*, 2)  Top  **STACK (push):**  (-, 1) | a b |
| A[4] = ‘c’ | (\*, 2)  Top  **STACK:**  (-, 1) | a b c |
| A[5] = ‘^’ | (\*, 2)  Top  **STACK (push):**  (^, 3)  (-, 1) | a b c |
| A[6] = ‘d’ | (\*, 2)  Top  **STACK:**  (^, 3)  (-, 1) | a b c d |
| A[7] = ‘+’ | (\*, 2)  Top  **STACK (pop):**  (-, 1) | a b c d ^ |
| (-, 1)  Top  **STACK (pop):** | a b c d ^ \* |
| Top  **STACK (pop, blank):** | a b c d ^ \* - |
| (+, 1)  Top  **STACK (push):** | a b c d ^ \* - |
| A[8] = ‘f’ | (+, 1)  Top  **STACK:** | a b c d ^ \* - f |
|  | Top  **STACK (pop, blank):** | a b c d ^ \* - f + |

* 1. Bài 7a

1 2 + 3 1 + \* 1 1 + 1 + /

+ Bước 1:

1 2 + 3 1 + \* 1 1 + 1 + /

Khi gặp 1 và 2 đẩy 1 và 2 vào statck

|  |  |
| --- | --- |
| 1 | 2 |

+ Bước 2:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu + thì ta pop để lấy 2 số 1 và 2
* Dùng phép cộng lấy 1+2=3
* Đẩy 3 vào stack

|  |
| --- |
| 3 |

+ Bước 3:

1 2 + 3 1 + \* 1 1 + 1 + /

Khi gặp 3 và 1 đẩy 3 và 1 vào stack

|  |  |  |
| --- | --- | --- |
| 3 | 3 | 1 |

+ Bước 4:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu + thì ta pop để lấy 2 số 3 và 1
* Dùng phép cộng lấy 3+1=4
* Đẩy 4 vào stack

|  |  |
| --- | --- |
| 3 | 4 |

+ Bước 5:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu \* thì ta pop để lấy 2 số 3 và 4
* Dùng phép nhân lấy 3\*4=12
* Đẩy 12 vào stack

|  |
| --- |
| 12 |

+ Bước 6:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp 1 và 1 ta đẩy 2 số vào stack

|  |  |  |
| --- | --- | --- |
| 12 | 1 | 1 |

+ Bước 7:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu + thì ta pop để lấy 2 số 1 và 1
* Dùng phép cộng lấy 1+1=2
* Đẩy 2 vào stack

|  |  |
| --- | --- |
| 12 | 2 |

+ Bước 8:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp 1 ta đẩy 1 vào stack

|  |  |  |
| --- | --- | --- |
| 12 | 2 | 1 |

+ Bước 9:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu + thì ta pop để lấy 2 số 2 và 1
* Dùng phép cộng lấy 2+1=3
* Đẩy 3 vào stack

|  |  |
| --- | --- |
| 12 | 3 |

+ Bước 10:

1 2 + 3 1 + \* 1 1 + 1 + /

* Khi gặp dấu / thì ta pop để lấy 2 số 12 và 3
* Dùng phép chia lấy 12/3=4
* Đẩy 4 vào stack

Kết quả bài toán: Ta pop phần tử cuối cùng ra là 4

* 1. Bài 7b

3 4 + 3 5 + \* 7 + 8 \*

+ Bước 1:

3 4 + 3 5 + \* 7 + 8 \*

Khi gặp 3 và 4 đẩy 3 và 4 vào stack

|  |  |
| --- | --- |
| 3 | 4 |

+ Bước 2:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp dấu + thì ta pop để lấy 2 số 3 và 4
* Dùng phép cộng lấy 3+4=7
* Đẩy 7 vào stack

|  |
| --- |
| 7 |

+ Bước 3:

3 4 + 3 5 + \* 7 + 8 \*

Khi gặp 3 và 5 đẩy 3 và 5 vào stack

|  |  |  |
| --- | --- | --- |
| 7 | 3 | 5 |

+ Bước 4:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp dấu + thì ta pop để lấy 2 số 3 và 5
* Dùng phép cộng lấy 3+5=8
* Đẩy 8 vào stack

|  |  |
| --- | --- |
| 7 | 8 |

+ Bước 5:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp dấu \* thì ta pop để lấy 2 số 7 và 8
* Dùng phép nhân lấy 7\*8=56
* Đẩy 56 vào stack

|  |
| --- |
| 56 |

+ Bước 6:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp 7 ta đẩy 7 vào stack

|  |  |
| --- | --- |
| 56 | 7 |

+ Bước 7:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp dấu + thì ta pop để lấy 2 số 56 và 7
* Dùng phép cộng lấy 56+7=63
* Đẩy 63 vào stack

|  |
| --- |
| 63 |

+ Bước 8:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp 8 ta đẩy 8 vào stack

|  |  |
| --- | --- |
| 63 | 8 |

+ Bước 9:

3 4 + 3 5 + \* 7 + 8 \*

* Khi gặp dấu \* thì ta pop để lấy 2 số 63 và 8
* Dùng phép nhân lấy 63\*8=504
* Đẩy 504 vào stack

|  |
| --- |
| 504 |

Kết quả bài toán: Ta pop phần tử cuối cùng ra 504

1. Bài tập
   1. Bài 1a

#include<iostream>

#include<stdio.h>

#include<algorithm>

using namespace std;

//----Khai bao stack

struct stack {

int n;

int a[10009];

};

typedef struct stack STACK;

STACK st;

//--- Kiem tra stack rong---

int isEmpty(STACK st) {

if (st.n == 0) return 1;

return 0;

}

//--- Day vao stack--

void push(STACK &st, int x) {

st.n++;

st.a[st.n - 1] = x;

}

//------Lay ra stack -----

int get(STACK &st) {

int x = st.a[st.n - 1];

st.n--;

return x;

}

//----- Dem phan tu---

int countl(int x) {

int d = 0;

while (x) {

x /= 10;

++d;

}

return d;

}

int main() {

//----nhap phan tu---

int pt;

cin >> pt;

int len = countl(pt);

int pdiv = 1;

for (int i = 1; i < len; i++)

pdiv \*= 10;

while (pt) {

push(st, pt / pdiv);

pt %= pdiv;

pdiv /= 10;

}

while (!isEmpty(st))

cout << get(st);

system("pause");

cout << endl;

return 0;

}

* 1. Bài 1b

#include<iostream>

#include<stdio.h>

#include<algorithm>

using namespace std;

//---- khai bao queue;

struct queue {

int n;

int a[10009];

};

typedef struct queue QUEUE;

QUEUE que;

//--------kiem tra rong----

int isEmpty(QUEUE que) {

if (que.n == 0) return 1;

return 0;

}

//-------day vao queue ----

void push(QUEUE &que, int x) {

que.n++;

que.a[que.n - 1] = x;

}

//---- lay ra queue ----

int get(QUEUE &que) {

if (isEmpty(que)) return -1;

else {

int x = que.a[0];

que.n--;

for (int i = 0; i < que.n; i++)

que.a[i] = que.a[i + 1];

return x;

}

}

int main() {

que.n = 0;

int pt;

cin >> pt;

while (pt) {

int du = pt % 10;

pt /= 10;

push(que, du);

}

while (!isEmpty(que)) {

cout << get(que);

}

system("pause");

return 0;

}

* 1. Bài 2

#include "stdafx.h"

#include <iostream>

#include <vector>

using namespace std;

struct stack

{

vector<char>stk;

int n = 0;

};

void output(stack);

stack converter(int, int);

void push(stack &, char);

char pop(stack &);

int main()

{

int x, base;

cout << "Nhap x: ";

cin >> x;

do

{

cout << "Nhap base: ";

cin >> base;

} while (base > 16 || base < 2);

output(converter(x, base));

return 0;

}

void push(stack & a, char x)

{

++a.n;

a.stk.push\_back(x);

}

char pop(stack & a)

{

if (!a.n)

return '\0';

char temp = a.stk[--a.n];

a.stk.pop\_back();

return temp;

}

stack converter(int x, int base)

{

stack a;

char str[] = {'0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F' };

for (int n = x; n; n /= base)

push(a, str[n%base]);

return a;

}

void output(stack a)

{

for (; a.n > 0;)

cout << pop(a) << " ";

cout << endl;

}

* 1. Bài 3

#include "stdafx.h"

#include <string>

#include <stack>

#include <cmath>

#include <iostream>

using namespace std;

void solve(stack<int> source, stack<int> aux, stack<int> dest, int num\_of\_disks);

void moveDisk(string fromPeg, string toPeg, int disk);

void moveDisksBetweenTwoPoles(stack<int> &source, stack<int> &dest, string s, string d);

int main()

{

stack<int> source, aux, dest;

int number\_of\_disks;

cout << "Nhap so dia: "; cin >> number\_of\_disks;

solve(source, aux, dest, number\_of\_disks);

return 0;

}

void solve(stack<int> source, stack<int> aux, stack<int> dest, int num\_of\_disks)

{

int i, total\_num\_of\_moves;

string s = "cot 1", d = "cot 3", a = "cot 2";

// Nếu số đĩa là chẵn, đổi chỗ của cột 2 và cột 3

if (num\_of\_disks % 2 == 0)

d.swap(a);

total\_num\_of\_moves = pow(2, num\_of\_disks) - 1;

//Khởi tạo stack với số đĩa cần chuyển

for (i = num\_of\_disks; i >= 1; i--)

source.push(i);

for (i = 1; i <= total\_num\_of\_moves; i++)

{

if (i % 3 == 1)

moveDisksBetweenTwoPoles(source, dest, s, d);

else if (i % 3 == 2)

moveDisksBetweenTwoPoles(source, aux, s, a);

else if (i % 3 == 0)

moveDisksBetweenTwoPoles(aux, dest, a, d);

}

}

void moveDisksBetweenTwoPoles(stack<int> &source, stack<int> &dest, string s, string d)

{

// Nếu cột 1 rỗng

if (source.empty() == true)

{

int pole2TopDisk = dest.top();

dest.pop();

source.push(pole2TopDisk);

moveDisk(d, s, pole2TopDisk);

}

// Nếu cột 2 rỗng

else if (dest.empty() == true)

{

int pole1TopDisk = source.top();

source.pop();

dest.push(pole1TopDisk);

moveDisk(s, d, pole1TopDisk);

}

// Nếu đĩa ở đầu stack 1 > đĩa ở đầu stack 2

else if (source.top() > dest.top())

{

int temp = dest.top();

source.push(temp);

dest.pop();

moveDisk(d, s, temp);

}

// Nếu đĩa ở đầu stack 2 > đĩa ở đầu stack 1

else

{

int temp = source.top();

dest.push(temp);

source.pop();

moveDisk(s, d, temp);

}

}

void moveDisk(string fromPeg, string toPeg, int disk)

{

cout << "Di chuyen dia " << disk << " tu " << fromPeg << " toi " << toPeg << endl;

}

* 1. Bài 4

char priotity[200];

priority[‘+’] = 1;

priotity[‘-‘] = 1;

priority[‘\*’] = 2;

priority[‘/’] = 2;

priority[‘^’] = 3;

vector<char> answer;

stack <char> st;

for (int i = 0; I < A.size(); ++i)

if (‘0’ <= a[i] && a[i] <= ‘9’ || ‘a’ <= a[i] && a[i] <= ‘z’)

answer.push\_back(a[i]);

else {

while (!st.empty() && priority[a[i]] <= st.top()) {

answer.push\_back(st.top());

st.pop();

}

st.push();

}

While (!st.empty()) {

answer.push\_back(st.top());

st.pop();

}

* 1. Bài 5

#include <iostream>

#include <string>

using namespace std;

#define Max 100

struct Stack

{

int n;

char Storage[Max];

};

void Push (Stack &s , char x)

{

s.Storage[s.n++] = x;

}

char Pop (Stack &s)

{

char c = s.Storage[s.n-1];

s.n--;

return c;

}

void Init (Stack &s)

{

s.n = 0;

}

bool IsEmpty (Stack s)

{

if ( s.n == 0)

return true;

return false;

}

bool IsFull (Stack s)

{

if ( s.n == Max )

return true;

return false;

}

bool CheckPair (char a , char b)

{

//cout<<a<<" "<<b<<endl;

if ( (a == '(') && (b != ')'))

return false;

else if ( (a == '[') && (b != ']'))

return false;

else if ( (a == '{') && (b != '}'))

return false;

return true;

}

void PrintStack(Stack s)

{

for (int i = 0 ; i < s.n ; i++)

cout<<s.Storage[i];

cout<<endl;

}

bool Input (Stack &s, string S)

{

bool flag = false;

for (unsigned int i = 0 ; i < S.size() ; i++)

{

if (S[i] == '(' || S[i] == '[' || S[i] == '{')

Push(s,S[i]);

else if (S[i] == ')' || S[i] == ']' || S[i] == '}')

{

flag = true;

if ( !CheckPair(Pop(s),S[i]) )

return false;

}

}

if (!flag || IsEmpty(s) == false)

return false;

return true;

}

int main()

{

string S;

cout<<"Nhap vao van ban can kiem tra:"<<endl;

getline (cin,S);

Stack s;

Init(s);

Input(s,S);

//PrintStack(s);

if (Input(s,S) == true)

cout<<"Cac dau ngoac hop le"<<endl;

else

cout<<"Cac dau ngoac khong hop le"<<endl;

return 0;

}