## Stack & Queue

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

stack = stream with only ome input/output port

stack = stream with two separated input port, output post

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| stack s |  |  |  |  |  |  |  | 🡨in/out🡪 |  |
|  |  |  |  |  |  |  |  |  |  |
| queue q | out🡨 |  |  |  |  |  |  | 🡨in |  |
|  |  |  |  |  |  |  |  |  |  |

**Methods**

vector<T> v; // khoi tao

v.size();

v.push\_back(x); // nap x vao dau phai

v.push\_back(v.begin(), x); // nap x vao dau trai

v.push\_back(v.begin()+p, x); // xen x vao truoc vi tri p

v.back(x); // xem dau phai

v.front(); // xem dau trai

v.pop\_back(); // xoa dau phai

v.erase(v.end()-1); // xoa dau phai

v.erase(v.begin()); // xoa dau trai

sort(v.begin(), v.end());

Hien thi

for(int i = 0; i < v.size(); ++i) cout << " " << v[i];

VI w(v); // VI w; copy v -> w: Tao moi va gan tri

w.assign(v); // copy v -> w

v.clear(); // xoa noi dung van con bien v

v.empty();

|  |  |  |
| --- | --- | --- |
| Method | Left | Right |
| vector | VI v | |
| clear | v.clear() | |
| size |  | |
| print |  | |
| begin |  | |
| end |  | |
| add |  |  |
| xem |  |  |
| xoa |  |  |
|  |  |  |

## 

## **Demo**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | vector |  |  |  |  |  |  | vector | Library |
| stack s |  |  |  |  |  |  |  | 🡨in: s.push\_back(x) | s.push(x); |
|  |  |  |  |  |  |  |  | 🡪out: x = s.back(); | x = s.pop(); |
|  |  |  |  |  |  |  |  | s.erase(s.end()-1) |  |
|  |  |  |  |  |  |  |  | return x; |  |
|  |  |  |  |  |  |  |  |  |  |
| queue q | 🡪out: x = q.back(); |  |  |  |  |  |  | 🡨in: q.push\_back(x) | q.push(x); |
|  | q.erase(q.begin()) |  |  |  |  |  |  |  | x = q.pop(); |
|  | return x; |  |  |  |  |  |  |  |  |

**Stack demo 1**

/\*

Name: stack

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description:

Tinh tri cua bieu thuc hau to (Lukasiewicz)

\*/

#include <bits/stdc++.h>

using namespace std;

typedef vector<int> VI;

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

void Print(VI v, const char \*msg = "") {

cout << msg;

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

cout << " " << v[i];

}

void Run() {

VI s;

for(int x = 1; x < 10; ++x)

s.push\_back(x);

Print(s);

for(int i = 1; i <= 3; ++i) {

int x = s.back();

cout << "\n i = " << i << ". x = " << x;

s.erase(s.end()-1);

}

Print(s, "\n s: ");

}

main() {

Run();

cout << endl << "\n\n T h e E n d \n";

return 0;

}

**Output 1**

1 2 3 4 5 6 7 8 9

i = 1. x = 9

i = 2. x = 8

i = 3. x = 7

s: 1 2 3 4 5 6

**Stack Demo 2**

/\*

Name: stack

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description:

Dung stack lat mang a[d..c]

\*/

#include <bits/stdc++.h>

using namespace std;

typedef vector<int> Stack;

#define Push(s,x) s.push\_back(x)

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

void Print(Stack v, const char \*msg = "") {

cout << msg;

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

cout << " " << v[i];

}

void Print(int a[], int n, const char \*msg = "") {

cout << msg;

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

cout << " " << a[i];

}

int Pop(Stack &s) {

int x = s.back();

s.erase(s.end()-1);

return x;

}

// lat mang tu a[d..c]

void Rev(int a[], int d, int c) {

Stack s;

for(int i = d; i <= c; ++i)

Push(s,a[i]);

Print(s, "\n stack s: ");

for(int i = d; i <= c; ++i)

a[i] = Pop(s);

Print(s, "\n stack s = empty: ");

}

void Run() {

int a[] = {0,1,2,3,4,5,6,7,8,9};

int n = sizeof(a) / sizeof(int);

Print(a, n, "\n Init a: ");

Rev(a, 3, 6);

Print(a, n, "\n Rev(a[3..6]): ");

Rev(a, 0, n-1);

Print(a, n, "\n Rev(a[0..n-1]): ");

}

main() {

Run();

cout << endl << "\n\n T h e E n d \n";

return 0;

}

Output 2

Init a: 0 1 2 3 4 5 6 7 8 9

stack s: 3 4 5 6

stack s = empty:

Rev(a[3..6]): 0 1 2 6 5 4 3 7 8 9

stack s: 0 1 2 6 5 4 3 7 8 9

stack s = empty:

Rev(a[0..n-1]): 9 8 7 3 4 5 6 2 1 0

T h e E n d

**Stack Demo3**

/\*

Name: stack

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description:

Tinh tri cua bieu thuc hau to (Lukasiewicz)

\*/

#include <bits/stdc++.h>

using namespace std;

typedef vector<int> Stack;

#define Push(s,x) s.push\_back(x)

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

void Print(Stack v, const char \*msg = "") {

cout << msg;

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

cout << " " << v[i];

}

void Print(int a[], int n, const char \*msg = "") {

cout << msg;

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

cout << " " << a[i];

}

int Pop(Stack &s) {

int x = s.back();

s.erase(s.end()-1);

return x;

}

void Run() {

// a bcdef ghijk lmnop qrstu vwxyz

// e = "(k+f)\*(j+b)"; // (10+5)\*(9+1) = 15\*10 = 150

string h = "kf+jb+\*"; // Hau to

cout << "\n (k+f)\*(j+b) -> " << h;

Stack s;

int a, b;

for(int i = 0; i < h.length(); ++i) {

char c = h[i];

if(c >= 'a' && c <= 'z') {

Push(s,c-'a');

continue;

}

switch(c) {

case '+': a = Pop(s); b = Pop(s);

Push(s,b+a);

break;

case '-': a = Pop(s); b = Pop(s);

Push(s,b-a);

break;

case '\*': a = Pop(s); b = Pop(s);

Push(s,b\*a);

break;

case '/': a = Pop(s); b = Pop(s);

Push(s,b/a);

break;

} // switch

} // for

cout << "\n Final result: " << Pop(s);

}

main() {

Run();

cout << endl << "\n\n T h e E n d \n";

return 0;

}

**Output 3**

(k+f)\*(j+b) -> kf+jb+\*

Final result: 150

T h e E n d

## **Joshephus Problem (game)**

Rounding of n elements, count k elem. Who is the rest ?

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A green circle with numbers  Description automatically generated | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |   m = 14, k = 3, res = 2 |

Algorithm? Data Structure?

5 approaches

Version 1. rounding scan (rai gianh) int array

Version 2. rounding scan (rai gianh) bit array

Version 2. rounding scan (rai gianh) bit array

/\*

Name: JP.CPP

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description: Josephus Problem

'''

\*/

#include <bits/stdc++.h>

using namespace std;

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

void Print(int a[], int d, int c, const char \* msg = "") {

cout << msg;

for (int i = d; i <= c; ++i) {

cout << " " << a[i];

}

}

void Print(char b[], int d, int c, const char \* msg = "") {

cout << msg;

for (int i = d; i <= c; ++i) {

cout << " " << (int)b[i];

}

}

// int array, mark 0/1 O(n)

int JP1(int n, int k) {

int a[n];

for(int i = 0; i < n; ++i) a[i] = i+1;

int n1 = n-1;

Print(a, 0, n1, "\n Init");

int d, j = -1;

for(int i = 1; i < n; ++i) {

d = 0;

while(d < k) {

j = (j + 1) % n;

if(a[j] > 0) ++d;

}

a[j] = 0;

Print(a, 0, n1, "\n a:");

} // i

Print(a, 0, n-1, "\n Rest:");

int res;

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

if (a[i] > 0) {

res = a[i];

break;

}

cout << "\n Result: " << res;

return res;

}

// int array, counting O(n)

int JP2(int n, int k) {

int a[n];

for(int i = 0; i < n; ++i) a[i] = 1;

int n1 = n-1;

Print(a, 0, n1, "\n Init");

int d, j = -1;

for(int i = 1; i < n; ++i) {

d = 0;

while(d < k) {

j = (j + 1) % n;

d += a[j];

}

a[j] = 0;

Print(a, 0, n1, "\n a:");

} // i

Print(a, 0, n-1, "\n Rest:");

int res;

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

if (a[i] > 0) {

res = i+1;

break;

}

cout << "\n Result: " << res;

return res;

}

// bit array, O(n)

int JP3(int n, int k) {

char a[n];

for(int i = 0; i < n; ++i) a[i] = 1;

int n1 = n-1;

Print(a, 0, n1, "\n Init");

int d, j = -1;

for(int i = 1; i < n; ++i) {

d = 0;

while(d < k) {

j = (j + 1) % n;

d += a[j];

}

a[j] = 0;

Print(a, 0, n1, "\n a:");

} // i

Print(a, 0, n-1, "\n Rest:");

int res;

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

if (a[i] > 0) {

res = i+1;

break;

}

cout << "\n Result: " << res;

return res;

}

void Print(vector<int> a, const char \* msg = "") {

cout << msg;

for (int i = 0; i < a.size(); ++i) {

cout << " " << a[i];

}

}

// Queue O(n)

int JP4(int n, int k) {

vector<int> q;

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

q.push\_back(i);

Print(q, "\n Init: ");

for(int i = 1; i < n; ++i) { // lap n-1 lan

// chuyen k-1 phan tu dau ve cuoi

for(int j = 1; j < k; ++j) {

q.push\_back(q.front()); // lay dau

q.erase(q.begin()); // bo dau

} // j

q.erase(q.begin()); // bo phan tu thu k

Print(q, "\n q: "); //Go();

} // i

int res = q.front();

cout << "\n Result: " << res;

return res;

}

void Rev(int a[], int d, int c) {

while(d < c) {

int x = a[d]; a[d] = a[c]; a[c] = x;

++d; --c;

}

}

// 123|4567 -> 4567|123

// 321|7654 -> 4567|123

// chuyen m phan tu ve cuoi

// 3n

void Move(int a[], int n, int m) {

Rev(a, 0, m-1);

Rev(a, m, n-1);

Rev(a, 0, n-1);

}

// Myqeueu, O(n) Tu cai queue

int JP5(int n, int k) {

int a[n];

for(int i = 0; i < n; ++i) a[i] = i+1;

Print(a, 0, n-1, "\n Init: ");

int nn = n;

for(int i = 1; i < nn; ++i) { // lap n-1 lan

Move(a, n, k);

--n;

Print(a, 0, n-1, "\n a: ");

}

return a[0];

}

main() {

int n = 14, k = 3;

JP5(n,k);

cout << endl << "\n\n T h e E n d \n";

return 0;

}

### If you want...Ver 6

/\*

Name: JP.CPP (Ver. 6)

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description: Josephus Problem

'''

\*/

#include <bits/stdc++.h>

using namespace std;

/\*

vector<T> q;

q.push\_back(x); // nap x vao dau phai

x = q.front(); // xem dau trai

q.erase(q.begin()); // xoa dau trai

v.size();

v.clear();

v.back();

v.front();

v.empty();

\*/

#define Add Push

#define AddRight Push

#define GetLeft Get

#define Del Erase

#define Delete Erase

#define Front Get

class MyQueue {

public:

vector<int> Data;

MyQueue() { Data.clear(); }

inline void Push(int x) { // add x to the right

Data.push\_back(x);

}

inline Erase() { // delete elem in the left

Data.erase(Data.begin());

}

inline int TakeOut() { // takout lay phan tu tu dau trai

int x = Data.front(); // lay tri

Erase(); // xoa

}

inline int Get() { // xem dau trai

return Data.front();

}

inline void Move() { // lay trai -> nap phai

int x = Data.front();

Push(x);

Data.erase(Data.begin());

//Push(Take());

}

void Print(const char \* msg) {

cout << msg;

for (int i = 0; i < Data.size(); ++i) {

cout << " " << Data[i];

}

}

};

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

void Print(int a[], int d, int c, const char \* msg = "") {

cout << msg;

for (int i = d; i <= c; ++i) {

cout << " " << a[i];

}

}

void Print(char b[], int d, int c, const char \* msg = "") {

cout << msg;

for (int i = d; i <= c; ++i) {

cout << " " << (int)b[i];

}

}

void Print(vector<int> a, const char \* msg = "") {

cout << msg;

for (int i = 0; i < a.size(); ++i) {

cout << " " << a[i];

}

}

// Using Queue

int JP4(int n, int k) {

MyQueue q;

for(int i = 1; i <= n; ++i) q.Push(i);

q.Print("\n Init: ");

for(int i = 1; i < n; ++i) { // lap n-1 lan

// chuyen k-1 phan tu dau ve cuoi

for(int j = 1; j < k; ++j) q.Move();

q.Erase(); // bo phan tu thu k

q.Print("\n q: "); //Go();

} // i

int res = q.Front();

cout << "\n Result: " << res;

return res;

}

main() {

int n = 14, k = 3;

JP4(n,k);

cout << endl << "\n\n T h e E n d \n";

return 0;

}

### One more...Ver 7: Simplest queue

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *0* | *1* | *2* | *3* | *4* | *5* | *6* | *7* | *8* | *9* | *10* | *11* | *12* | *12* |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| **L** |  |  |  |  |  |  |  |  |  |  |  |  | **R** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

/\*

Name: JP.CPP (Ver 7)

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description: Josephus Problem

simplest queue

'''

\*/

#include <bits/stdc++.h>

using namespace std;

int L, R, N;

int \*a;

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

inline void Inc(int &i) { i = (i+1) % N; }

void Print(int a[], int d, int c, const char \* msg = "") {

cout << msg;

while(true) {

cout << " " << a[d];

if (d == c) return;

Inc(d);

}

}

int TakeOut() { // TakeOut the left

int x = a[L];

Inc(L);

return x;

}

void Push(int x) { // add x to the Rigjht

Inc(R);

a[R] = x;

}

void Move() { // Move L -> R

int x = TakeOut();

Push(x);

}

// Using simplest Queue

int JP7(int n, int k) {

N = n;

a = new int[N];

//fill(a, a+N, 0);

L = 0; R = N-1;

for(int i = 0; i < n; ++i) a[i] = i + 1;

Print(a, L, R, "\n Init: ");

for(int i = 1; i < N; ++i) { // loop n-1 times

// move k-1 elem front L to R

for(int j = 1; j < k; ++j) Move();

TakeOut(); // del

Print(a, L, R, "\n a: ");

}

int res = a[L];

cout << "\n Result: " << res;

return res;

}

main() {

int n = 14, k = 3;

JP7(n,k);

cout << endl << "\n\n T h e E n d \n";

return 0;

}

### MyQueue Again

/\*

Name: JP.CPP (Ver 8)

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description: Josephus Problem

simplest queue

'''

\*/

#include <bits/stdc++.h>

using namespace std;

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

class MyQueue {

public:

// Data

int \* Data;

int Len;

int L, R;

// Methods

inline MyQueue(int n) {

Len = n;

Data = new int[Len];

fill(Data, Data + Len, 0);

L = 0; R = Len-1;

}

inline Size() { return Len; }

inline void Inc(int &i) { i = (i+1) % Len; }

inline void Print(const char \* msg = "") {

int d = L, c = R;

cout << msg;

while(true) {

cout << " " << Data[d];

if (d == c) return;

Inc(d);

}

}

inline int TakeOut() { // TakeOut the left

int x = Data[L];

Inc(L);

return x;

}

inline void Push(int x) { // add x to the Rigjht

Inc(R);

Data[R] = x;

}

inline void Move() { // Move L -> R

Push(TakeOut());

}

inline int Left() {

return Data[L];

}

inline int Right() {

return Data[R];

}

};

// Using simplest Queue

int JP7(int n, int k) {

MyQueue q(n);

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

q.Push(i+1);

q.Print("\n Init: "); Go();

for(int i = 1; i < n; ++i) { // loop n-1 times

// move k-1 elem front L to R

for(int j = 1; j < k; ++j) q.Move();

q.TakeOut(); // del

q.Print("\n q: ");

}

int res = q.Left();

cout << "\n Result: " << res;

return res;

}

main() {

int n = 14, k = 3;

JP7(n,k);

cout << endl << "\n\n T h e E n d \n";

return 0;

}

### Using stringstream

/\*

Name: JP.CPP (Ver 9)

Copyright: (C) 2023

Author: Devcpp Fan

Date: 22-06-23 11:50

Description: Josephus Problem

stringstream

'''

\*/

#include <bits/stdc++.h>

using namespace std;

void Go() {

cout << " ? ";

fflush(stdin);

if (cin.get()=='.') exit(0);

}

// Using simplest Queue

int JP9(int n, int k) {

stringstream ss;

char BL = ' '; // 32

for(int i = 1; i <= n; ++i) {

ss << i; ss << BL;

}

int x;

for(int i = 1; i < n; ++i) { // loop n-1 times

for(int j = 1; j < k; ++j) {

ss >> x;

ss << BL << x;

}

ss >> x;

cout << "\n " << i << ": " << ss.str();

}

int res;

ss >> res;

cout << "\n Result: " << res;

return res;

}

main() {

int n = 14, k = 3;

JP9(n,k);

cout << endl << "\n\n T h e E n d \n";

return 0;

}