

BÁO CÁO ĐỒ ÁN MÔN HỌC
(Đồ án tìm hiểu CTDL/Giải thuật)
Lớp: IT003.O21.CNTT

SINH VIÊN THỰC HIỆN

Mã sinh viên: 23520742

Họ và tên: Lê Nguyễn Anh Khoa

TÊN ĐỀ TÀI: Trie

CÁC NỘI DUNG CẦN BÁO CÁO: kết quả tìm hiểu được về CTDL/GT đã chọn

1. Giới thiệu đồ án:

- Trie là một cấu trúc dữ liệu dạng cây dùng để lưu trữ các xâu kí tự.
- Trie có nhiều ứng dụng trong Khoa Học Máy Tính.
- Giải quyết các bài toán thực tế cụ thể

2. Quá trình thực hiện:

- a. Tuần 1: Tìm hiểu các khái niệm, cách cài đặt và những ứng dụng thường gặp
- b. Tuần 2: Áp dụng Trie để giải quyết các vấn đề thực tế hiện tại

3. Kết quả đạt được

- a. Các định nghĩa/khái niệm cơ bản
 - **Trie** là một cấu trúc dữ liệu dạng cây được dùng để lưu trữ và xử lý một tập các xâu. Do tính chất dễ hiểu và dễ cài đặt nên **Trie** được sử dụng rất phổ biến hiện nay.
 - Mỗi đường đi từ gốc đến một nút bất kì sẽ là tiền tố của một xâu. Như vậy, việc lưu trữ các xâu có tiền tố giống nhau trở nên rất hiệu quả.
- b. Các đặc trưng nổi bật
 - Các thao tác trên cây được thực hiện trong thời gian tuyến tính

- Có thể lưu các xâu có tiền tố giống nhau một cách hiệu quả
- c. Các thao tác cơ sở (nếu là CTDL)
- Thêm 1 xâu vào tập hợp
 - Xóa 1 xâu khỏi tập hợp
 - Kiểm tra xâu có trong tập hợp hay không
- d. Vận dụng thực tế
- Sử dụng các thao tác cơ bản để giải các bài toán phổ thông hiện nay
 - Áp dụng Trie để giải các bài toán phức tạp hơn
4. Phụ lục 1: [Link github](#)
5. Phụ lục 2: Demo chương trình minh họa
- Bước 1: tải repo github và mở file trie.cpp, trie.inp, trie.out .
 - Bước 2: Điền input vào file trie.inp theo dạng:
 - + “Insert s”: Trong đó s là xâu muốn thêm vào tập hợp .
 - + “Delete s”: Trong đó s là xâu muốn xóa khỏi tập hợp
 - + “Find s”: Trong đó s là xâu muốn kiểm tra
 - Bước 3: Chạy file trie.cpp, kết quả được hiển thị qua file trie.out .

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 #define SZ(x) (int)((x).size())
4 struct Trie{
5     static Node*
6     Node* child[26];
7     int leaf, cnt;
8     Node() {
9         for(int i = 0; i < 26; i++) child[i] = NULL;
10        leaf = cnt = 0;
11    }
12    ~Node() {}
13    Node* root;
14    Trie() {
15        root = new Node();
16    }
17    void add_string(string s) {
18        // Parameters
19        // s: string
20        // s: string
21        Node* p = root;
22        for (auto f : s) {
23            int c = f - 'a';
24            if(p->child[c] == NULL) p->child[c] = new Node();
25        }
26    }
27    // ...
28    // ...
29    // ...
30    // ...
31    // ...
32    // ...
33    // ...
34    // ...
35    // ...
36    // ...
37    // ...
38    // ...
39    // ...
40    // ...
41    // ...
42    // ...
43    // ...
44    // ...
45    // ...
46    // ...
47    // ...
48    // ...
49    // ...
50    // ...
51    // ...
52    // ...
53    // ...
54    // ...
55    // ...
56    // ...
57    // ...
58    // ...
59    // ...
60    // ...
61    // ...
62    // ...
63    // ...
64    // ...
65    // ...
66    // ...
67    // ...
68    // ...
69    // ...
70    // ...
71    // ...
72    // ...
73    // ...
74    // ...
75    // ...
76    // ...
77    // ...
78    // ...
79    // ...
80    // ...
81    // ...
82    // ...
83    // ...
84    // ...
85    // ...
86    // ...
87    // ...
88    // ...
89    // ...
90    // ...
91    // ...
92    // ...
93    // ...
94    // ...
95    // ...
96    // ...
97    // ...
98    // ...
99    // ...
100   // ...

```

```

1 Insert khon
2 Find khon
3 Delete khon
4 Find khon
5

```

```

1 Xon nam trong tap hop
2 Xon khong nam trong tap hop
3

```

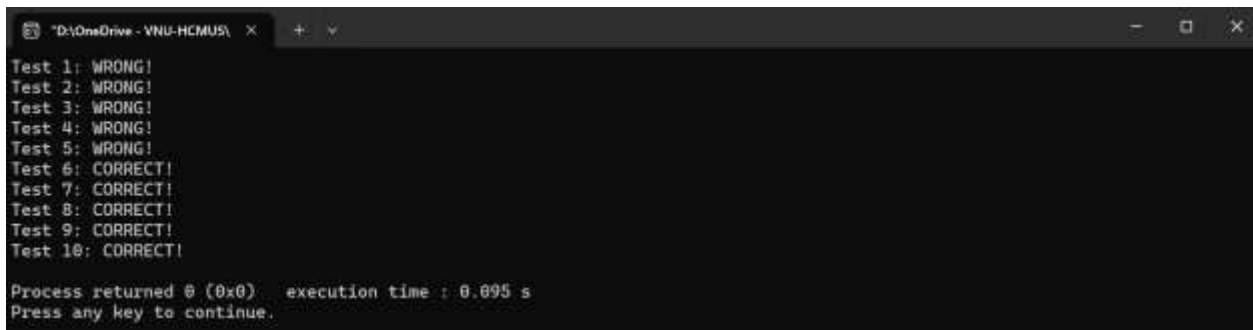
6. Phụ lục 3: Tự làm bài toán 1 và bài toán 2

- Bước 1: Mở file “Bai_Toan_1/main.cpp” và làm vào file



```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int n, k;
5 string s;
6
7 signed main()
8 {
9     for(int i = 1; i <= 10; ++i)
10     {
11         Setup for input and output file
12         string file_in = "TEST/test", file_out = "TEST/test";
13         file_in = file_in + to_string(i) + ".inp";
14         file_out = file_out + to_string(i) + ".out";
15         const char* file_in_cstr = file_in.c_str();
16         const char* file_out_cstr = file_out.c_str();
17         freopen(file_in_cstr, "r", stdin);
18         freopen(file_out_cstr, "w", stdout);
19
20         cin >> n >> k;
21         for(int j = 1; j <= n; ++j)
22             cin >> s;
23     }
24 }
```

- Bước 2: Mở và chạy file “checker.cpp” để xem kết quả



```
Test 1: WRONG!
Test 2: WRONG!
Test 3: WRONG!
Test 4: WRONG!
Test 5: WRONG!
Test 6: CORRECT!
Test 7: CORRECT!
Test 8: CORRECT!
Test 9: CORRECT!
Test 10: CORRECT!

Process returned 0 (0x0)   execution time : 0.095 s
Press any key to continue.
```

- Chú ý:

- + Nếu muốn sinh thêm test có thể mở và chạy file “Bai_Toan_1/Test_generator.cpp” để sinh thêm test sau đó chạy file “Bai_Toan_1/solution.cpp” để chạy ra output chuẩn.
- + Toàn bộ code chuẩn trong phần “Bai_Toan_1/solution.cpp”.
- + Có thể làm tương tự với Bài toán 2.