

MÔN : CÔNG NGHỆ JAVA

Bài thực hành 2.2 : Viết chương trình tạo từ

I. Mục tiêu :

- Giúp SV làm quen với môi trường lập trình trực quan NetBeans (version 7.3.1) để viết các ứng dụng thuật giải back-tracking và chạy ở chế độ text.

II. Nội dung :

- Dùng NetBeans viết chương trình chạy ở chế độ textmode (console), thực hiện các yêu cầu sau :

1. chờ user nhập vào chuỗi có n ký tự khác nhau.
2. tìm tất cả phương án để tạo từ dài n ký tự, mỗi ký tự là ký tự do user nhập, mỗi ký tự được dùng 1 lần trong từ.
3. in các từ tạo được để người dùng xem dễ dàng.

III. Chuẩn đầu ra :

- Sinh viên làm quen môi trường NetBeans, biết dùng nó để viết chương trình đơn giản chạy ở chế độ text.
- Sinh viên nắm vững cách viết đoạn lệnh thực hiện thuật giải back-tracking và biết cách dùng các kiểu dữ liệu như array và class.

IV. Phân tích :

1. Ta dùng thuật giải backtracking để tìm các phương án tạo từ như sau :

- + Lập tìm chỉ số gốc của ký tự được dùng cho từng vị trí (cột) trong từ, từ trái sang phải :
 - Nếu sắp được cho cột c thì tăng c 1 đơn vị để tiếp tục cột kế tiếp, nếu cột c đã vượt qua MAX thì đã tìm được 1 từ mới.
 - Nếu không sắp được cho cột c thì giảm c 1 đơn vị để thử tiếp các khả năng còn lại của cột cũ này. Nếu c < 0 thì hết cách.

2. Để lưu vết xử lý trên từng cột, ta dùng biến array chiso[MAX], mỗi phần tử chiso[c] chứa chỉ số ký tự gốc đã xử lý lần cuối trước đây (-1 : chưa, MAX : hết).

Từ thuật giải được viết bằng ngôn ngữ tự nhiên ở trên, ta dịch nó thành chương trình được viết bằng Java như trong mã nguồn dưới đây.

V. Qui trình :

1. Chạy NetBeans, nếu cửa sổ Project bên trái màn hình còn hiển thị Project nào đó (của lần cuối cùng trước đó), ấn phải chuột trên phần từ gốc của cây Project để hiển thị menu các chức năng, rồi chọn option Close để đóng Project tương ứng lại.
2. Chọn menu File.New Project để máy hiển thị cửa sổ "New Project", chọn mục "Java" trong Listbox Categories, chọn mục "Java Application" trong Listbox Projects rồi click button Next để hiển thị cửa sổ "New Application".
3. Xác định thư mục chứa Project ở textbox "Project Location", nhập "NBTaoTu" vào textbox "Project Name", click button Finish để máy tạo thực sự Project. Cửa sổ soạn mã nguồn của class chương trình có tên là NBTaoTu hiển thị.
4. Viết code cho thân của class NBTaoTu như sau :

```
public class NBTaoTu {  
    //định nghĩa các thuộc tính dữ liệu  
    static final int MAXLEN = 20; //số ký tự max cần dùng để tạo từ  
    static String srcstr; //chuỗi các ký tự khác nhau để tạo từ
```

```

static int sotu; //số từ đã tìm được
static int[] chiso = new int[MAXLEN + 1]; //danh sách chỉ số ký tự đã thử trong từng cột
static int sokytu; //số ký tự gốc được dùng để tạo từ
static int c; //vị trí cột đang sắp

//Điểm nhập của chương trình
public static void main(String[] args)
{
    //tạo đối tượng phục vụ đọc chuỗi từ bàn phím
    BufferedReader bufferRead = new BufferedReader(new
InputStreamReader(System.in));
    //chờ nhập chuỗi ký tự dùng để tạo từ
    do
    {
        try {
            System.out.print("Hãy nhập chuỗi các ký tự cần dùng : ");
            srcstr = bufferRead.readLine();
            //xác định số ký tự được dùng
            sokytu = srcstr.length();
        } catch (Exception e) {}
    } while (sokytu >= MAXLEN);

    // khởi động trạng thái xuất phát
    for (c = 0; c < sokytu; c++) chiso[c] = -1;
    //bắt đầu đặt ký tự chỉ số 0 vào cột đầu (cột 0) của từ cần xây dựng
    chiso[0] = 0;
    //bắt đầu tìm ký tự cho cột thứ 2 (chỉ số là 1) của từ cần xây dựng
    c = 1;
    while (Tim1cach())
    {
        // tìm được 1 từ mới
        InKetqua();
        //lùi lại cột trước để tiếp tục tìm cách sắp khác
        c--;
    }
    // hết cách --> dừng chương trình
    System.out.println("Số từ xây dựng được là " + sotu);
    return;
}

//cố gắng tìm 1 phương án sắp các ký tự thành 1 từ mới theo yêu cầu
static boolean Tim1cach() {
    int h;
    while (c < sokytu)
    {
        // tìm chỉ số ký tự cho ký tự cột c
        h = timkytu(c);
        if (h >= 0)
        { //tìm được ==> lưu chỉ số vào danh sách kết quả rồi tăng chỉ số cột để tiếp tục
            chiso[c++] = h;
        }
    }
}

```

```

        continue;
    }
    //trường hợp không tìm được ký tự ở cột c
    if (c > 0)
    {
        //cột c chưa phải là cột đầu tiên => xét lại cột c từ ký tự đầu rồi lùi cột c
        chiso[c--] = -1;
        continue;
    }
    //hết cách
    return false;
}
return true;
} //hết hàm Tim1cach

//hàm tìm chỉ số hàng cho con hậu ở cột c
static int timkytu(int c)
{
    int h, hmin = chiso[c] + 1;
    for (h = hmin; h < sokytu; h++) // thử ký tự chỉ số h
        if (duocpheap(h, c)) return h;
    return -1;
}

// Kiểm tra xem có thể xếp ký tự thứ h vào vị trí thứ c trong từ cần xây dựng
static boolean duocpheap(int h, int c)
{
    int k;
    for (k = 0; k < c; k++)
        if (chiso[k] == h) return false;
    return true;
}

//Kiểm tra từ, nếu có nghĩa thì trả về TRUE, nếu không trả về FALSE
static boolean conghia(String str)
{
    // Bạn viết lại hàm này theo yêu cầu riêng
    //ở đây ta coi mọi từ đều có nghĩa
    return true;
}

//hàm in từ mới tìm theo yêu cầu
static void InKetqua()
{
    String desstr;
    int k;
    //Xây dựng từ mới tìm được
    desstr = "";
    for (k = 0; k < sokytu; k++)
        desstr += Character.toString(srcstr.charAt(chiso[k]));
}

```

```

        if (conghia(desstr))    //kiểm tra từ có nghĩa trước khi in
        {
            System.out.println("Từ thứ " + (++sotu) + " : " + desstr);
        }
    }
} //hết class

```

5. Dời về đầu file mã nguồn, thêm các lệnh import sau đây vào sau lệnh package :

```
//import các package cần dùng
```

```
import java.io.*;
```

```
import java.util.*;
```

6. Chọn menu Run.Run Project để dịch và chạy thử chương trình. Nếu có lỗi từ vựng và cú pháp thì sửa, nếu có lỗi run-time thì debug (thông qua các chức năng trong menu Debug) để xác định lỗi rồi sửa lỗi.
7. Nếu chương trình hết lỗi, chương trình sẽ chạy, cửa sổ output sẽ hiển thị yêu cầu nhập danh sách các ký tự cần dùng rồi hiển thị tuần tự các từ tạo được theo yêu cầu. Hãy chạy lại chương trình nhiều lần, mỗi lần nhập danh sách các ký tự khác nhau và xem kết quả để đánh giá tính đúng đắn của chương trình.