

```

public class Ex2Baril {
    // variables globales
    // stockage des mots de Dyck pour tester la distance
        static String[] ts;
        static int m; //taille des mots
    static int z = 0; //indice dans le tableau ts

    // calcule le nb de mots de Dyck de taille n
        static int count108(int n) {
            if (n < 0)
                return -1;
            else if (n == 0)
                return 1;
            else
                return 2*(2*n-1)*count108(n-1)/(n+1);
        }

    // génère et stocke dans ts les codages des mots de Dyck de taille m
    // sous forme de chaines de caractères
    // affiche les mots de parenthèses correspondants, et leur rang
    // d = num appel rec, on récupère la dernière lettre ajoutée
    public static void dyck(int d, String chaine){
        if(d == m+1) {
            ts[z++] = chaine;
            System.out.println(chaine);
            System.out.println(d2p(chaine)+" rang: "+rang(chaine));
        }
        else for (int i = d; i <= m; i++)
            if (d == 1 || (d > 1 && i >= Integer.parseInt(chaine.substring(d-
2))))
                dyck(d+1, chaine+i);
    }

    // d2p donne le mot constitué de parenthèses p correspondant
    // au nb de parenthèses ouvrantes a
    // a[j]+j donne la position de ')'
    public static String d2p(String a) {
        int i, j = 1;
        String p = "";
        for (i = 1; i <= 2*a.length(); i++) {
            if (i == Integer.parseInt(a.substring(j-1, j))+j) { p += ')'; j++; }
            else p += '(';
        }
        return p;
    }

    public static String inf(String a, String b) {
        String p = "";
        for (int i = 0 ; i < a.length() ; i++)
            p += (a.charAt(i) < b.charAt(i)) ? a.charAt(i) : b.charAt(i);
        return p;
    }

    public static String sup(String a, String b) {
        String p = "";
        for (int i = 0 ; i < a.length() ; i++)
            p += (a.charAt(i) > b.charAt(i)) ? a.charAt(i) : b.charAt(i);
    }
}

```

```

    return p;
}

public static int rang(String a) {
    int c = 0, n = a.length();
    for (int i = 0 ; i < n ; i++)
        c += Integer.parseInt(a.substring(i,i+1));
    return c - (n*(n+1)/2);
}

public static int d(String a, String b) {
    int n = a.length();
    return rang(a) + rang(b) - 2*rang(inf(a,b));
}

public static void main(String[] args){

    System.out.println("Donner la taille des mots");
    m = Lire.i();
    int n = count108(m);
    ts = new String [n];
    dyck(1,"");
    System.out.println("distance entre "+
    d2p(ts[0])+" et "+d2p(ts[n-1])+" : "+d(ts[0],ts[n-1]));

}
}

```