

```

class D  composition {

    /*
     * Ant  d  dent : n>0 et cand>=2
     */
    static String myst  re(int n, int cand) {
        assert cand>=2;
        if (n==1) return "1";
        if (n%cand == 0)
            return cand + " x " + myst  re(n/cand, cand);
        return myst  re(n, cand+1);
    }

    /**
     * Ant  d  dent : n>0
     * R  le : renvoie la d  composition de n en facteurs premiers
     */
    static String facteursPremiersR(int n) {
        assert n>0;
        return myst  re(n, 2);
    }

    /**
     * Ant  d  dent : n>0
     * R  le : renvoie la d  composition de n en facteurs premiers
     */
    static String facteursPremiers(int n) {
        String res="";
        int nbPremier=2;
        while (n>1) {
            if (n%nbPremier == 0) {
                // nbPremier est un facteur premier
                res+= nbPremier + " x ";
                n/=nbPremier;
            }
            else
                nbPremier++;
        }
        // nbPremier == 1;
        return res + "1";
    }

    public static void main(String [] args) {
        int n = 24;
        System.out.println(n + " = " + facteursPremiersR(n));
        System.out.println(n + " = " + facteursPremiers(n));
    }
}

```