```java
import java.util.Scanner;

/**
 * 完全背包问题,基本思路，时间复杂度较高，使用二维数组
 */
public class Test2 {
    // N表示物体的个数，V表示背包的载重
    int N,V;
    //用于存储每个物体的重量，下标从1开始
    private int[] weight;
    //存储每个物体的收益，下标从1开始
    private int[] value;
    //降成二维数组，用来保存每种状态下的最大收益
    private int[][] F;

    public void CompletePackNonRecursive() {
        //对二维数组F进行初始化
        for(int i = 0; i <= V; i++) {
            F[0][i] = 0;
        }

        //注意边界问题，i是从1开始的
        for(int i = 1; i <= N; i++) {
            //j是正序、降序没影响
            for (int j = 0; j <= V; j++) {
                for (int k = 0; k <= V / weight[i]; k++) {
                    if (j >= k * weight[i]) {
                        //注意：状态转移方程是F[i][j]，而不是F[i - 1][j]
                        //因为这时放k个第i个物品，之后还可能继续放这个物体，所以应是F[i]
[j]
                        F[i][j] = Math.max(F[i - 1][j - k * weight[i]] + k *
value[i], F[i][j]);
                    } else {
                        //可以省略，这里为什么不是F[i - 1][j]
                        //因为刚开始k=0，j >= 0 * weight[i]肯定成立，此时F[i][j] =
F[i - 1][j]。
                        F[i][j] = F[i][j];
                    }
                }
            }
        }

        //打印所有结果，我们要求的是F[N][V]
        int res = 0;
        for(int i = 0; i <= N; i++) {
            for(int j = 0; j <= V; j++) {
//                System.out.print(F[i][j] + " ");
                res = Math.max(res,F[i][j]);
            }
//            System.out.println();
        }
        System.out.println(res);
    }

    /**
```

```
     * 输入格式：
     5 10
     2 2 6 5 4
     6 3 5 4 6
     result:30
     * 第一行是物体个数、背包总空间；
     * 第二行是每个物体的空间；
     * 第三行是每个物体的收益。
     */
    public void init() {
        Scanner sc = new Scanner(System.in);
        N = sc.nextInt();
        V = sc.nextInt();

        //下标从1开始，表示第1个物品
        weight = new int[N + 1];
        value = new int[N + 1];
        F= new int[N + 1][V + 1];

        for(int i = 1; i <= N; i++) {
            weight[i] = sc.nextInt();
        }

        for(int i = 1; i <= N; i++) {
            value[i] = sc.nextInt();
        }
    }

    public static void main(String[] args) {
        Test2 cpe2 = new Test2();
        cpe2.init();
        cpe2.CompletePackNonRecursive();
    }
}
```