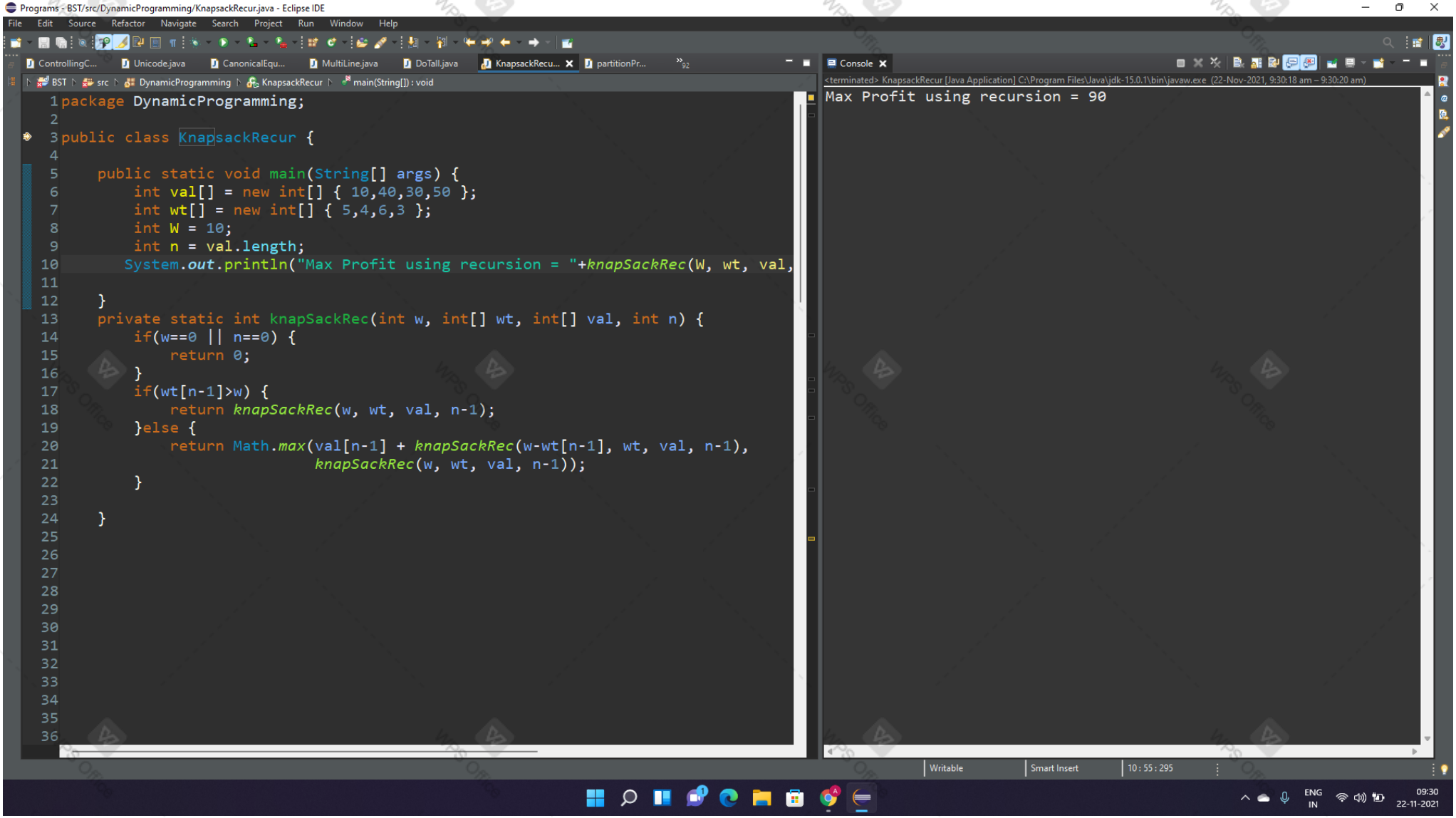
Programs - BST/src/DynamicProgramming/KnapsackRecur.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
ControllingC... Unicode.java CanonicalEqu... MultiLine.java DoTall.java KnapsackRecu... partitionPr...
BST src DynamicProgramming KnapsackRecur main(String[]) : void
1 package DynamicProgramming;
2
3 public class KnapsackRecur {
4
5 public static void main(String[] args) {
6 int val[] = new int[] { 10,40,30,50 };
7 int wt[] = new int[] { 5,4,6,3 };
8 int W = 10;
9 int n = val.length;
10 System.out.println("Max Profit using recursion = "+knapSackRec(W, wt, val,
11
12 }
13 private static int knapSackRec(int w, int[] wt, int[] val, int n) {
14 if(w==0 || n==0) {
15 return 0;
16 }
17 if(wt[n-1]>w) {
18 return knapSackRec(w, wt, val, n-1);
19 }else {
20 return Math.max(val[n-1] + knapSackRec(w-wt[n-1], wt, val, n-1),
21 knapSackRec(w, wt, val, n-1));
22 }
23
24 }
25
26
27
28
29
30
31
32
33
34
35
36
Console
<terminated> KnapsackRecur [Java Application] C:\Program Files\Java\jdk-15.0.1\bin\javaw.exe (22-Nov-2021, 9:30:18 am – 9:30:20 am)
Max Profit using recursion = 90
Writable
Smart Insert
10 : 55 : 295
ENG
IN
09:30
22-11-2021