A list of positive integers is called a partition of natural number \( n \) if the sum of its items is \( n \). Write a program to find

(a) a list of all partitions of a given natural \( n \). For example, if \( n=3 \) then an acceptable answer is \([3]; [1; 2]; [2; 1]; [1; 1; 1]\).

(b) a list of all sorted partitions of a given natural \( n \). For example, if \( n=3 \) then an acceptable answer is \([3]; [1; 2]; [1; 1; 1]\).

(c) the sorted list of all partitions of a given natural \( n \). For example, if \( n=3 \) then the answer is \([1; 1; 1]; [1; 2]; [2; 1]; [3]\).

(d) the sorted list of all sorted partitions of a given natural \( n \). For example, if \( n=3 \) then the answer is \([1; 1; 1]; [1; 2]; [3]\).

After trying the question, scroll down to the solution.
Given a sorted partition, to get the next sorted partition: cut off the

(b) a list of all sorted partitions of a given natural \( n \). For example, if \( n=3 \) then an acceptable answer is \([[[3]; [1; 2]; [2; 1]; [1; 1; 1]]] \).

Part (a) is subsumed by part (c).

(c) the sorted list of all partitions of a given natural \( n \). For example, if \( n=3 \) then the answer is \([[[1; 1; 1]; [1; 2]; [2; 1]; [3]]] \).

Part (b) is subsumed by part (d).

(d) the sorted list of all sorted partitions of a given natural \( n \). For example, if \( n=3 \) then the answer is \([[[3]; [1; 2]; [2; 1]; [1; 1; 1]]] \).

From Chapter 5 (Program using loops). Instead of gathering the partitions into a list, I print them (let's say \( x \) prints the value of \( x \) and \( ?x \) reads into variable \( x \)).

```plaintext
var n, m: int; n is the length of \( P \) and \( m \) is a temporary

val n = s"n"; ?n.

var P: [n*int];
for i := 0...n do P i := 1 od.

for i := 0...n do !P i, " " od. !newline.
exit when n < 2.

P(n-2) := P(n-2) + 1.
m := n - 1. n := m + P m - 1.
for i := m...n do P i := 1 od od
```

The exact execution time is obtained by putting \( t := t + 1 \) in front of the recursive call, and replace \( R \) by \( t' = t + 2^{n-1} - 1 \) replace \( A \Rightarrow R \) by \( t' = t + \sum i: 1..\#P. 2^{\sum P[i] \cdot \#P} - 1 \)

Part (c) is subsumed by part (d).
\(d := P(\#P-2) + P(\#P-1)\). \(f := P(\#P-2)+1\).

\(P := P[0;..\#P-2] ;; [(\text{div } df - 1)*f ; f + \text{mod } df]\)

and we have to modify \(R\) and \(A\).