```
1  procedure gen_komb(n, S, p: integer);
2  begin
3      { n: nr kombinacji od 0, S: liczba elementów zbioru, p: liczba elementów podzb. }
4      { v, lewy_r, prawy_r, i: zmienne cakowite }
5      n := n + 1;     { ...teraz kombinacje są numerowane od 1}
6      i := 0;     {nr pierwszego elementu zbioru}
7      while S > 0 do
8          v := (S p);     {wartość bieżącego elementu trójkąta}
9          prawy_r := (S-1 p);     { „prawy rodzic"}
10         if p < S then
11             lewy_r := v − prawy_r;
12         else
13             lewy_r := 1;     { „lewy rodzic"}
14         end if;
15         S := S − 1;     { „piętro" w górę}
16         if n > lewy_r then      {idziemy w prawo do góry}
17             n := n − lewy_r;
18             writeln('element ',i, 'nie należy do podzbioru');
19         else   {idziemy w lewo do góry}
20             writeln('element ',i, 'należy do podzbioru');
21             p := p − 1;     { zmniejszamy liczbę elementów podzbioru („k")}
22         end if;
23         i := i + 1;     { nr kolejnego elementu zbioru}
24     end while;
25 end.
```

1. Algorytm generowania k-elementowej kombinacji