Zaimplementuj strukturę danych „lista dwukierunkowa” w sposób następujący:

* Pojedynczy element struktury powinien przechowywać wartość typu int.
* Zdefiniuj metodę addTo(dlista l, int arg, unsigned pos), która pozwoli na dodanie nowego elementu na wskazaną pozycję listy;
* Zdefiniuj metode deleteFrom(dlista l, unsigned pos), która pozwoli na usunięcie elementu ze wskazanej pozycji listy;

Rozwiąż następujące zadanie.

Masz do dyspozycji dwa pliki:

VnumGrupy\_\*.tsk - zawiera dane wejściowe Twojego algorytmu;

solv\_VnumGrupy\_\*.tsk – zawiera docelowy wynik działania algorytmu.

Otwórz plik VnumGrupy\_\*.tsk w trybie binarnym, wczytuj jego zawartość bajt bo bajcie.

* Wczytaj bajt, to będzie kod polecenia.
* Jeżeli wczytany bajt jest kodem znaku ‘<’:

Wczytaj kolejny bajt, to będzie numer pozycji, **pos**,

Wczytaj kolejny bajt, to będzie wartość, **arg**,

Dodaj wartość **arg** na pozycję **pos**.

* Jeżeli wczytany bajt jest kodem znaku ‘>’:

Wczytaj kolejny bajt, to będzie numer pozycji, **pos**,

Usuń element listy o numerze **pos**.

Po wczytaniu całego pliku przebuduj listę dwukierunkową w drzewo binarne. Pola next oraz prev będą spełniały role, odpowiednio: pól left oraz right.

Wyświetl zbudowane drzewo binarne.

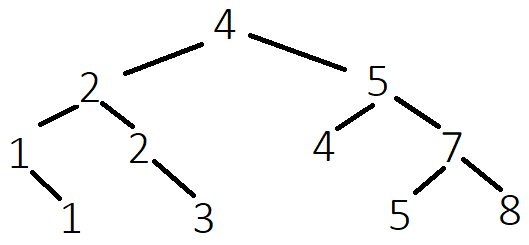
Wyznacz wysokość zbudowanego drzewa. Wysokość definiujemy jako maksymalną liczbę krawędzi grafu na drodze od korzenia do liścia.

Po skończeniu programu zwolnij pamięć, usuwając drzewo.

Przykład:

wejście: 4 5 2 1 7 8 5 4 2 1 3

zbudowane drzewo:



Wysokość: 3