

Algoritmy i Struktury Danych II, Zestaw 6

SORTOWANIE TOPOLOGICZNE

Proszę napisać program wykonujący sortowanie topologiczne [Link 1, Link 2] acyklicznego grafu skierowanego (ang. *directed acyclic graph*, *DAG*) [Link 3] w oparciu o algorytm DFS. Działania algorytmu dla grafu (DAGu) G tłumaczy poniższy pseudokod:

TOPOLOGICAL-SORT(G)

```
1 Wykonaj DFS( $G$ ) w celu obliczenia czasów przetworzenia  $v.f$ 
dla każdego wierzchołka  $v$ 
2 Wstaw każdy wierzchołek na początek listy,
kiedy tylko zostanie przetworzony
3 RETURN lista wierzchołków
```

Na poniższym rysunku zaprezentowano przykład zastosowania sortowania topologicznego opartego o algorytm DFS:

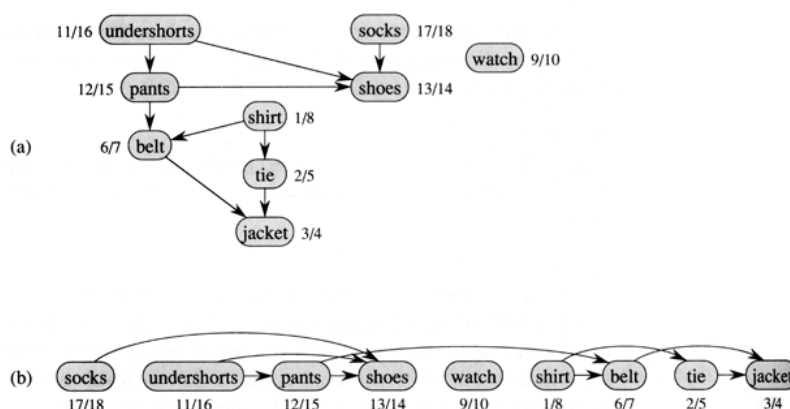


Figure 23.7 (a) Professor Bumstead topologically sorts his clothing when getting dressed. Each directed edge (u, v) means that garment u must be put on before garment v . The discovery and finishing times from a depth-first search are shown next to each vertex. (b) The same graph shown topologically sorted. Its vertices are arranged from left to right in order of decreasing finishing time. Note that all directed edges go from left to right.

Sortowanie topologiczne proszę przeprowadzić na wygenerowanym losowo DAGu. Można to zrobić generując jednostkową przypadkową trójkątną dolną macierz sąsiedztwa [Link 4]. Ponadto, proszę przyjąć brak pętli (tj. $\text{Adj}[i][i] = 0$). Proszę rozważyć prawdopodobieństwo wystąpienia linku $P(1) = p$ oraz jego braku $P(0) = 1 - p$, dla różnych wartości p z przedziału $[0, 1]$. Proszę rozważyć grafy o różnej ilości węzłów V . Wykorzystując algorytm DFS, proszę sprawdzić czy otrzymany graf jest rzeczywiście acykliczny. W oparciu o macierz sąsiedztwa, proszę narysować rozważany graf. Można w tym celu wykorzystać np. program online do którego kieruje [Link 5].

Materiały pomocnicze: Link 1, Link 2, Link 3, Link 4, Link 5.