

Uždavinys “ŽODIS”.

Sprendimo idėja.

Suformuluotas uždavinys yra NP sunkumo, bet duotiems apribojimams galima sukonstruoti pakankamai greitą algoritmą, randantį *teisingą* atsakymą per leistiną laiką. Pirmiausia pastebėkime, kad tam tikra prasme atstumas nuo vieno langelio iki kito priklauso tik nuo langelių pozicijų, bet ne nuo kelio, kuriuo mes keliaujame nuo vieno prie kito:

Kelio ilgis nuo X iki Y yra 15.

Tegul S_i , $i=1, \dots, K$ (K – duotojo žodžio ilgis) yra i – tosios duotojo žodžio raidės pozicijų lentelėje aibė. Bet kuris kelias, kuriame pasitaiko bent vieną kartą kiekviena raidė iš žodžio, turi turėti bent vieną atstovą iš kiekvienos aibės S_i . Tai leidžia perrinkti visus įmanomus kelio variantus. Pasirinkę tam tikrus atstovus, galime juos iš dalies surikiuoti (tam tikslui naudojame lentelės eilučių, kurioms priklauso atstovai, numerius). Kiekvieną eilutę, kurioje yra mūsų daugiau, negu vienas atstovas, galima pereiti iš kairės į dešinę ne po du variantus. Perrinkę šiuos „subvariantus“, gauname – atstumų sumą.

Pavyzdys. Tarkime reikia „surinkti“ (uždavinio sąlygos prasme) žodį AŠIS lentelėje (nereikšmingos raidės lentelėje neparodytos). Šiuo atveju turime du variantus.

						A
			S			
	A			I		
S						

Reikia surinkti žodį
AŠIS

→	→	→	→	→	A↓
			Š→	↔↓	←
				I↓	
				↓	
S↓	←	←	←	←	
→	→	→	→	→	×

Pirmas variantas.

Pirmas variantas – kai imame “kampinę” A raidę. Kelio įvertis tada ir pats kelias – akivaizdus.

↓					
→	→	→	Š→	↓	
	A↓	←	←	I↓	
	↓				
S↓	←				
→	→	→	→	→	×

Antras variantas.

Antras variantas – sunkesnis – reikia tikrinti dar du „subvariantus“ – kur eiti iš Š raidės pozicijos – prie raidės A arba prie I ? Tik įvertinę bendrą kelią abiejų „subvariantų“ atvejais, galime pasirinkti tinkamą.