

AAL - dokumentacja wstępna

Temat 5: Zalewanie wykresu słupkowego

Przeanalizowanie problemu:

Algorytm przyjmuje na wejściu ciąg n nieujemnych liczb reprezentujących wysokości słupków. Moim zadaniem jest znalezienie maksymalnej ilości wody jaka się zmieści pomiędzy dwoma dowolnymi słupkami. Założeniem zadania jest to, że wody nie wolno nalewać na słupek, w taki sposób że rozlewa się ona na jego lewo i prawo. Zostało również założone, że przed pierwszym i za ostatnim słupkiem nie ma niczego, czyli rozlana tam woda przepada.

Propozycja rozwiązania, algorytm optymalny:

Korzystamy z algorytmu gąsiennicowego. Moja gąsiennica ma głowę i ogon, oba są iteratorami w liście reprezentującej wysokości kolejnych słupków. Poruszamy się raz od lewej strony ciągu oraz drugi raz od prawej strony.

W zadanym przejściu zaczynamy od początku. Przesuwamy głowę gąsiennicy do czasu, gdy głowa będzie wyższa lub równa od wysokości ogonu gąsiennicy. Wtedy obliczamy jaka ilość wody zmieści się w tej „dolinie” między głową a ogonem gąsiennicy (bierzemy pod uwagę odległość między tymi iteratorami oraz wysokość mniejszej wysokości spośród wysokości ogona i głowy). Jeżeli znaleziona teraz pojemność jest większa niż dotychczasowa największa, bierzemy aktualną jako MaxCapacity.

Po przejściu ciągu od lewej oraz od prawej jako wynik algorytmu zwracamy największą znaną pojemność z obu porzejsi.

To rozwiązanie ma złożoność liniową.

Algorytm brutalny:

Pierwszym rozwiązaniem jakie stworzyłem było algorytmem brutalnym. Polegało ono na przechodzeniu od słupka i -tego do ostatniego słupka. W danej iteracji i przechodzę po kolejnych słupkach do czasu gdy napotkam słupek równy bądź wyższy od słupka i -tego. Obliczana jest wtedy odległość między tymi słupkami i mnożona jest przez minimum wysokości i -tego słupka oraz tego słupka do którego dotarliśmy. Wykonując ten algorytm dla każdego $i=1\dots n$ znajduję maksymalną pojemność.

Rozwiązanie algorytmem brutalnym ma złożoność kwadratową.

Sposób testowania:

Generowanie przypadków losowych polega na wylosowaniu n dodatnich liczb z zadanego przez użytkownika przedziału.

Analiza jakości rozwiązania:

Jakość algorytmu sprawdzam przez generowanie losowych „planszy” dla których wyznaczane jest rozwiązanie najpierw za pomocą algorytmu brutalnego, a następnie za pomocą mojego algorytmu. Porównywane są wyniki, w ten sposób weryfikowana jest poprawność mojego rozwiązania. W dokumentacji końcowej chciałbym zwrócić uwagę na porównanie wydajności tych rozwiązań.