

Univerzita Karlova v Praze
Přírodovědecká fakulta
Katedra aplikované geoinformatiky a kartografie

Úvod do programování

zadanie č. 44 – Selection Sort

technická správa

Daniela DANČEJOVÁ
Geografie a kartografie, 2. ročník
Čierne Pole, 2022

Zadanie

Vytvorte program, ktorý zoradí vstupné dáta vzostupne použitím algoritmu *Selection Sort*.

Rozbor úlohy

Selection Sort, resp. triedenie výberom je triediaci algoritmus s časovou zložitou $O(N^2)$, ktorý je pre svoju jednoduchosť a výkon často používaný pre usporiadanie malého množstva dát [1]. Pracuje na základe porovnávania a vymieňania dvojíc prvkov v zozname, pričom uskutočňuje minimálny počet výmen. Prvky môžu byť zoradené aj zostupne.

Algoritmus nie je stabilný – môže zmeniť poradie prvkov s rovnakým kľúčom, pretože vymieňa prvky, ktoré spolu nemusia susediť. Podobne nie je vhodný ani na spracovanie veľkého množstva dát, na ktoré sú určené algoritmy s menšími časovými zložitostami.

Popis algoritmu

Na vstupe je n prvkov na zoradenie. Algoritmus zvolí prvý prvok v poradí a porovná ho s najmenším prvkom, ktorý nájde pri prechádzaní zvyšku zoznamu; ak je tento prvok menší, vymení svoju pozíciu s prvým prvkom. Tento prvok je považovaný za „zoradený“, algoritmus ho ďalej ignoruje a zvolí ďalší prvok v poradí, ktorý rovnakým spôsobom vymení (prípadne nevymení) s druhým najmenším nájdeným prvkom. Takto prechádza zvyšok zoznamu, kým nezoradí všetky prvky, pričom pracuje na stále menšom rozsahu pôvodného zoznamu [1].

Počet porovnaní

Pri prvom prechádzaní zoznamu o počte n prvkov vykoná $n - 1$ porovnaní a vymení najmenší prvok s prvkom na prvej pozícii, druhý najmenší prvok sa dostane na druhú pozíciu porovnaním zvyšných $n - 1$ prvkov atď. Z toho vyplýva, že celkový počet opakovaní je:

$$(n - 1) + (n - 2) + \dots + 1 = \frac{n - 1}{2}n = \frac{n^2 - n}{2}$$

Príklad

Majme 5 náhodne zoradených číselných prvkov s rôznymi hodnotami:

12	9	34	21	5
----	---	----	----	---

Algoritmus zvolí prvý prvok a hľadá v zvyšku zoznamu čo najmenšiu hodnotu, ktorou je 5 (prechádza celý zvyšok zoznamu). Hodnoty zvoleného prvku a nájdeného prvku sa porovnávajú:

12	9	34	21	5
----	---	----	----	---

Keďže je 5 menšia ako 12, prvky vymení a zvolí nasledujúci prvok (9), pre ktorého pozíciu hľadá čo najmenšiu hodnotu. S prvým prvkom ďalej nepracuje.

5	9	34	21	12
---	---	----	----	----

Menšia hodnota ako 9 nebola nájdená; druhý prvok sa tak považuje za zoradený, ignoruje sa a zvolí sa ďalší prvok, ktorý sa zoradí rovnakým spôsobom.

5	9	34	21	12
---	---	----	----	----

Algoritmus pracuje na stále menšom rozsahu, na záver porovná zvyšné nezoradené pozície:

5	9	12	21	34
---	---	----	----	----

Zoradený zoznam:

5	9	12	21	34
---	---	----	----	----

Vstupné dáta

Vytvorený program požaduje vstupný súbor vo formáte CSV. Vstupné dáta by mali tvoriť výhradne racionálne čísla usporiadané do riadkov; v jednom riadku by sa malo nachádzať práve jedno číslo. Vstup by nemal obsahovať prázdny riadok (prípustný je však prázdny riadok na konci súboru), v súbore sa nesmú nachádzať nenumerné znaky. V týchto prípadoch sa program ukončí.

Program zoradí vstupné dáta vzostupne; po prijatí programom budú konvertované do formátu *float* (aby sa zabezpečila podpora čísel s ukončeným desatinným rozvojom). Poskytnuté sú vzorové dáta *random_numbers.csv*, ktoré obsahujú náhodných 50 čísel s tromi desatinnými miestami. Program bol vytvorený s pomocou týchto dát a preto preddefinovane požaduje

vstupný súbor s týmto názvom; tento názov je možné zmeniť prepísaním na vlastný vo vnútri programu pri premennej *infile*.

Výstupné dáta

Výstupom programu je zoradený zoznam čísel usporiadaný do riadkov, ktoré sú uložené do nového súboru vo formáte CSV pod preddefinovaným názvom *sorted_numbers.csv*. Tento názov je možné zmeniť vo vnútri programu pri premennej *outfile*.

Dokumentácia

Pre funkčnosť programu je potrebné ho spustiť v rovnakom adresári, v akom sú uložené vstupné dáta; v opačnom prípade program vyhodí chybu. Po spustení algoritmus prechádza riadky a kontroluje ich správnosť, následne vstupné dáta prekonvertuje do formátu *float* a zavolá metódu *selection_sort* s algoritmom popísaným vyššie. Zoradený zoznam vyexportuje do nového súboru vo formáte CSV v rovnakom adresári pomocou metódy *export_array*, ktorá vypíše každý prvok v zoradenom zozname do nového riadku.

Námety na vylepšenie

Program by bolo možné vylepšiť o schopnosť zoradiť prvky zostupne. Takisto by mohol podporovať ďalšie formáty vstupných súborov. V neposlednom rade by sa mohol zjednodušiť prístup používateľovi, aby mohol efektívnejšie zadávať vstupné súbory, špecifikovať výstupné súbory a určovať vlastný rozsah na zoradenie.

Zoznam použitých zdrojov

[1] javaTpoint: Selection Sort Algorithm. <https://www.javatpoint.com/selection-sort> (9. 1. 2022)

[2] Python Software Foundation: Python 3.10.2 documentation. <https://docs.python.org/3/> (9. 1. 2022)