Domácí úkol 7:

Mějme na vstupu seznam n hodnot a číslo m udávající kolik různých hodnot je na vstupu ($n >> m$). Chceme v očekávaném lineárním čase $O_{\mathbf{E}}(n)$ určit počty výskytů všech hodnot.

Pro řešení použijte nějaký c-univerzální hešovací systém, není podstatné který. Podstatné jsou parametry systému, tedy především velikost oboru hodnot.

Určete očekávanou časovou složitost algoritmu pomocí očekávaného počtu kolizí hodnot hešovací funkce. Určete prostorovou složitost.

Popis řešení:

Pořídíme si hešovací tabulku velikosti $\mathcal{O}(m)$ založenou na c-univerzálním systému, kde každá buňka bude držet seznam počítadel, jedno pro každou hodnotu s příslušným hešem. Pro každý prvek na vstupu spočítáme heš jeho hodnoty, najdeme příslušné počítadlo a zvýšíme jeho hodnotu (resp. počítadlo vytvoříme). Na konec projdeme tabulku a vypíšeme data ze všech počítadel.

Analýza prostorové složitosti:

Tabulka zabere pole velikosti $\mathcal{O}(m)$ a všechna počítadla v ní také $\mathcal{O}(m)$. Využíváme tedy $\mathcal{O}(m)$ pomocné paměti (+vstup).

Analýza časové složitosti:

Inicializaca tabulky zabere $\mathcal{O}(m)$, její přečtení na konci zabere $\mathcal{O}(m)$ za projití pole a $\mathcal{O}(m)$ za přečtení všech počítadel. Operace s počítadly (poté co je počítadlo nalezeno) jsou $\mathcal{O}(1)$ na prvek, tedy $\mathcal{O}(n)$ celkem. Zbývá určit celkovou složitost vyhledávání počítadel.

Pro každý prvek hledáme příslušné počítadlo v tabulce. Výpočet heše a lokalizace příslušné přihrádky tabulky je $\mathcal{O}(1)$, nalezení příslušného počítadla závisí na počtu jiných hodnot se stejným hešem (délce seznamu počítadel v přihrádce). Můžeme tedy spočítat kolik jiných (z celkových m různých) hodnot se bude kolidovat s fixní hodnotou x (z univerzality hešovacího systému a linearity střední hodnoty).

$$\mathop{\mathbb{E}}_{h\in\mathcal{H}}[\#x{-}colisions] = \sum_{y;y\neq x} \mathop{\Pr}_{h\in\mathcal{H}}[h(x)=h(y)] = (m{-}1)\frac{c}{m} = \mathcal{O}_{\mathbb{E}}(m/m) = \mathcal{O}_{\mathbb{E}}(1)$$

Očekávaná složitost přístupu ke každé hodnotě je tedy $\mathcal{O}(1)$, celkový počet přístupů bude mít časovou složitost $\mathcal{O}_{\mathbb{E}}(n)$. Celková složitost algoritmu je $\mathcal{O}_{\mathbb{E}}(n+m)$.

Nejčastější problémy a poznámky:

Kolize jsou mezi hodnotami (těch je m), nikoliv prvky na vstupu (n).

Celkový počet kolizí v analýza nepomůže. Nejčastější chybné použití bylo použití celkového počtu kolizí přímo jako časové složitosit. Pokud bychom přistupovali pouze jednou za každou hodnotu, potom by celková složitost byla závislá přímo na celkovém počtu kolizí - každý přístup závislý na počtu kolidujících hodnot, každá kolize se započítá dvakrát (jednou za každý prvek v kolidujícím páru). V našem případě se ale každá kolize započítá tolikrát kolikrát přistupujeme k prvkům z kolidujícího páru.

Složitost přístupu do tabulky závisí na velikosti přihrádek. Z celkového počtu kolizí nemáme garantováno, že neexistuje nekonstantně velká přihrádka (nebo dokonce několik), koliza mají zcela dovoleno se shlukovat. Protože nemáme kontrolu nad tím, kolik prvků ze vstupu bude přistupovat do velkých či malých příhrádek, korektní odhad složitosti jednoho přístupu (pokud odhadujeme z celkového počtu kolizí) nemůže být konstantní.

Tím, že spočítáme z univerzality přímo očekávané kolize s fixním prvkem (tedy velikosti přihrádek) se využije jiná vlastnost univerzálního systému (než garance celkového počtu kolizí) a problémy výše nenastanou. Dostaneme tak formálně to co intuitivně platí, tedy že přístup ke všem hodnotám je očekávaně konstantní.

Důležitý rozdíl! Očekávaně konstantní přístup pro všechny hodnoty znamená, že pokud pro každou hodnotu vezmeme průměr přes různé běhy algoritmu, dostaneme konstantu. Neznamená to, že v daném běhu algoritmu bude průměr přístupů k jednotlivým hodnotám konstantní. Oba průměry jsou zde počítány přes jiné množiny (jedna hodnota a různé běhy vs. jeden běh a různé hodnoty), což je přesně rozdíl kvůli kterému nefunguje úvaha přes celkový počet kolizí (jeden běh a různé hodnoty).

Alternativně, pokud m je velmi malé, mohli bychom použít tabulku velikosti m^2, která garantuje celkový počet kolizí $\mathcal{O}_{\mathbb{E}}(1)$. Za cenu většího prostoru a časové složitosti tak dostaneme dobrou garanci přístupu v očekávaně konstantním čase. Takové řešení má samozřejmě smysl hlavně pokud je $n >> m^2$, např. pokud $m = \mathcal{O}(1)$.