

Werkcollege Programmeermethoden 2 en 3 november 2023

1. (opgave 1 van het tentamen van 6 januari 2014)

In een array `int A[n]` staan n (een `const > 0`) gehele getallen.

a. Schrijf een C++-functie `hoevaak (A,X,n)` die teruggeeft hoe vaak het gehele getal X in het array A voorkomt.

b. Schrijf een Booleaanse C++-functie `uniek (A,n)` die precies dan `true` teruggeeft als geen enkel getal twee maal (of vaker) voorkomt in A , en anders `false`. Hierbij moet de functie van **a** *zinnvol* gebruikt worden (hoe vaak komt $A[i]$ voor?).

c. Schrijf een C++-functie `meest (A,n)` die het meest voorkomende getal uit A teruggeeft. Als er verschillende kandidaten zijn (bijvoorbeeld voor het array 17 12 30 12 42 30) moet het kleinste getal dat het meest voorkomt worden geretourneerd. In het voorbeeld is dit 12 (dat even vaak voorkomt als 30). Maak opnieuw gebruik van de functie van **a**.

d. Schrijf een C++-functie `sorteer (A,n)` die de getallen in A zodanig ordent dat voor alle getallen (behalve het laatste) geldt dat ze hooguit even vaak voorkomen als hun rechter buurman. Tip: pas de C++-code voor *bubblesort* eenvoudig aan; gebruik **a**.

e. Hoe vaak wordt de functie `hoevaak` aangeroepen in **d**?

2. (opgave 3 van het tentamen van 3 januari 2018)

Gegeven is het 2-dimensionale array `int stemmen[m][n]`; met $2 \ 5 \ 0 \ 3 \ 2$
zekere `const int m ≥ 2` en `const int n ≥ 2`. Er geldt dat $1 \ 5 \ 1 \ 4 \ 1$
`stemmen[i][j] ≥ 0` het aantal stemmen van jurylid i op liedje j $2 \ 1 \ 3 \ 3 \ 3$
is. Een voorbeeld met $m = 4$ en $n = 5$ staat hiernaast. $8 \ 1 \ 0 \ 0 \ 3$

a. Schrijf een Booleaanse C++-functie `eerlijk (stemmen)` die controleert of alle juryleden exact evenveel stemmen hebben uitgebracht. Oftewel: hebben alle rijen dezelfde totaalsom? In het voorbeeld: `true`.

b. Twee juryleden stemmen *ongeveer hetzelfde* als voor ieder liedje geldt dat hun aantal stemmen hooguit één verschilt, en tevens dat bij minstens één liedje er evenveel stemmen zijn. Schrijf een Booleaanse C++-functie `idem (stemmen,i1,i2)` die dit controleert voor de juryleden $i1$ en $i2$. Neem aan: $0 ≤ i1 < m$ en $0 < i2 < m$. In het voorbeeld `true` bij 0 en 1.

c. Begin bij een zeker jurylid $i1$, met $0 ≤ i1 < m$. Zoek het nieuwe jurylid $i2$ dat ongeveer hetzelfde stemt als $i1$ (gebruik **b**), waarbij $i2$ zo klein mogelijk is. Zoek vanuit $i2$ het nieuwe (verschillend van $i1$ en $i2$) jurylid $i3$ ($i3$ zo klein mogelijk) dat ongeveer hetzelfde stemt als $i2$, enzovoorts: $i4$ verschilt van $i1$, $i2$ en $i3$, en stemt ongeveer hetzelfde als $i3$, ... Je stopt indien je een dergelijk nieuw jurylid niet meer kunt vinden. Schrijf een C++-functie `int vrienden (stemmen,i1)` die dit doet, en teruggeeft hoeveel juryleden je zo gevonden hebt, inclusief $i1$. Tip: gebruik een Booleaans hulpparray.

Voor meer oude tentamens en antwoorden, zie

www.liacs.leidenuniv.nl/~kosterswa/pm/tentamens.php