

Tentamen Programmeermethoden

Woensdag 11 november 2015, 14:00–17:00 uur

Universiteit Leiden — Informatica



Let op: dit tentamen is bedoeld voor studenten Natuurkunde en/of Sterrenkunde.

Bij alle functies moeten de variabelen (constanten eventueel uitgezonderd) in de heading of lokaal voorkomen; vul zelf headings goed in. De te behalen punten (totaal 100) staan tussen haakjes bij de opgaven. Succes! Cijfers: www.liacs.leidenuniv.nl/~kosterswa/pm/cijf/res.html.

1. (35 punten) In een array `int A[n]` staan n (een `const` ≥ 1) gehele getallen > 0 .
 - a. (4) Schrijf een C++-functie `int hoe (A,X,n)` die bepaalt hoe vaak het gehele getal `X` in het array `A` voorkomt.
 - b. (4) Schrijf een C++-functie `int fac (k)` die met behulp van één `for`-loop de waarde van $k! = k \cdot (k - 1) \cdot \dots \cdot 2 \cdot 1$ uitrekent. Neem aan dat het gehele getal `k` groter dan 0 is. Gebruik *geen* recursie.
 - c. (6) Schrijf een Booleaanse C++-functie `controle (A,n)` die het volgende controleert voor het array `A`. In de elementen met even array-index > 0 moet een getal staan dat hooguit index keer voorkomt, en in de elementen met oneven array-index moet de faculteit van de index staan. (Bijvoorbeeld, met $n > 5$: als `A[4]` 7 is, mag 7 hooguit 4 keer voorkomen, en `A[5]` moet $5! = 120$ bevatten.) Als dit klopt, geeft de functie `true`, en anders `false`. De functie moet stoppen zodra het antwoord bekend is. Gebruik **a** en **b**.
 - d. (7) Schrijf een C++-functie `busort (A,n)` die het array `A` met behulp van *bubblesort oplopend* sorteert, waarbij het algoritme stopt zodra er in een ronde geen verwisselingen waren.
 - e. (4) Hoeveel vergelijkingen tussen array-elementen doet de methode van **d** minimaal en maximaal, uitgedrukt in n ? Geef voor beide situaties een rijtje getallen dat dit aantal realiseert.
 - f. (10) Schrijf een C++-functie `int langste (A,n,laatste)` die de lengte uitrekent van een langste direct opeenvolgende serie gelijke getallen in het array `A`. Daarbij moet `laatste` de array-index worden van het laatste element van een dergelijke serie; als er meer van dergelijke series zijn: degene met de hoogste index. Voor het array 1 7 1 1 4 7 7 3 zou het antwoord 2 zijn, en `laatste` moet 6 worden.

2. (30 punten) **a.** (6) Bij een functie kun je te maken hebben met *call by value* en *call by reference*, en ook met *locale* en *globale* variabelen. Verder onderscheiden we *formele* en *actuele* parameters. Leg deze zes begrippen duidelijk uit.

b. (6) Gegeven een C++-programma met daarin de volgende twee functies:

```
double gerard (double x, double y) {
    int i = 42; u++;
    for ( i = 1; i <= y; i++ ) { x = x + y; cout << x << endl; }//for
    y = 0; return x;
}//gerard
double harry (double v, double w) {
    if ( v < w ) w = gerard (v,w); else w = gerard (w,v);
    cout << w << endl; return v + w; u++;
}//harry
```

Verder zijn de globale variabelen `s`, `t` en `u` gegeven (`s` en `t` van type `double`, `u` van type `int`). Wat is dan de uitvoer van het volgende stukje programma (leg je antwoord duidelijk uit):

```
s = 2.1; t = 3.1; u = 41; cout << harry (s,t) << endl;
cout << s << ", " << t << ", " << u << endl;
```

c. (5) Idem, maar nu met vier `&`'s in de headings van `gerard` en `harry`.

d. (4) Wat gaat er fout bij (met de `&`'s er bij, net als in c):

```
s = 2.1; t = 4.1; u = 7; cout << harry (s,s) << endl;
cout << s << ", " << t << ", " << u << endl;
```

e. (4) Als in de functie `gerard` ergens `x = harry (3*x,u)`; staat, compileert het programma dan nog? Onderscheid gevallen met en zonder `&`.

f. (5) Geef een eenvoudige uitdrukking in `x` en `y` voor de waarde die `gerard (x,y)` retourneert. Bekijk hier de situatie zonder `&`'s, net als bij b.

3. (35 punten) Gegeven is een `m` bij `n` (beide `const > 0`) array `H` met gehele getallen ≥ 0 . Hierbij geeft `H[i][j]` de hoogte ter plekke (i, j) aan (met $0 \leq i < m$ en $0 \leq j < n$). Hiernaast staat een voorbeeld met `m = 4` en `n = 5`. De constanten `m` en `n` hoeven bij deze opgave niet doorgegeven te worden als parameter.

3	0	4	2	0
7	2	4	9	6
1	2	0	5	1
6	9	3	1	0

a. (8) Schrijf een C++-functie `aflagen (H,i,j,p,q)` die alle array-elementen > 0 in het array `H` in de rechthoek met linksboven (i, j) en rechtsonder (p, q) , met één aflaat. Neem aan dat $0 \leq i < p < m$ en $0 \leq j < q < n$. (Met `aflagen (H,0,0,m-1,n-1)` wordt het hele array behandeld.) Vergeet niet de heading compleet in te vullen!

b. (8) Schrijf een C++-functie `int gemiddeld (H,i,j)` die het gemiddelde van de vier (aan de randen drie of twee) hoogtes van direct horizontaal of verticaal aangrenzende burens van (i, j) berekent. Neem aan dat $0 \leq i < m$ en $0 \leq j < n$. Het gemiddelde moet op de gebruikelijk manier worden afgerond; zo moet `gemiddeld (H,2,4)` in het voorbeeld $(6 + 5 + 0)/3 = 11/3 \rightarrow 4$ opleveren.

c. (9) Schrijf een C++-functie `int maximum (H,i0,j0,aantal)` die het grootste getal uit `H` bepaalt, en de coördinaten daarvan in `i0` en `j0` oplevert. De parameter `aantal` moet aangeven hoe vaak dit maximum voorkomt. In het voorbeeld zou het antwoord 9 zijn (met `aantal` gelijk aan 2), waarbij `i0` bijvoorbeeld gelijk aan 3 wordt en `j0` gelijk aan 1 (of 2 respectievelijk 3: als het maximum meerdere keren voorkomt, mag je zelf kiezen).

d. (10) Schrijf een C++-functie `int egaliseer (H,drempel)` die het array `H` als volgt "egaliseert". Trek telkens van het (of beter: een) grootste getal van het array 1 af, totdat het maximum minstens `drempel` (een gegeven geheel getal > 0 en $\leq m \cdot n$) keer voorkomt. Het aantal benodigde stappen moet worden teruggegeven. In het voorbeeld, met `drempel` gelijk aan 4, is dat 7 keer: dan zijn de 9, 9 en 7 alle drie 6 geworden; het maximum 6 komt dan trouwens 5 keer voor. Gebruik c.

Op donderdagochtend 12 november is er alleen werkcollege voor hen die C++ doen, of de derde opgave nog niet hebben ingeleverd. Plaats: Snellius, zaal 302. Volgende week, woensdag 18 november, is het eerste van vier colleges over Python; dit wordt gegeven door dr. Kris Rietveld in de De Sitterzaal, van 11:15 tot 13:00 uur. Op de donderdagen is er vanaf dan ook weer werkcollege in het Snellius. Let op: aanwezigheid is verplicht, en wordt gecontroleerd!