
KLAS(SEN) : EQ	BLAD : 1 van 4 BLADEN
OND. DEEL : Object Georiënteerd Programmeren in C++	DOCENT : Harry Broeders
AFK. O.E. : OGOPRG	DATUM : -
KWART. : 3	TYPE : tentamen
	TIJD : -

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

Bij elke opgave staat tussen haakjes het maximale aantal te behalen punten vermeld.

Eindcijfer = (aantal behaalde punten + 10) / 10.

- 1.** Gegeven is het volgende C++ programma (geschreven door een C programmeur die net met C++ is begonnen):

```
#include <iostream>
using namespace std;

// Deze functie werkt alleen voor size >= 1
void minmaxElement(int array[], int size, int* max, int* min) {
    *max = array[0];
    *min = array[0];
    for (int i = 1; i < size; ++i) {
        if (array[i] > *max) *max = array[i];
        if (array[i] < *min) *min = array[i];
    }
}

int main() {
    int a[5] = {10, 2, 23, 1, 19};
    int groot, klein;
    minmaxElement(a, 5, &groot, &klein);
    cout << "Het maximum = " << groot << endl;
    cout << "Het minimum = " << klein << endl;
    cin.get();
    return 0;
}
```

De uitvoer van dit programma is als volgt:

```
Het maximum = 23
Het minimum = 1
```

- A. (10) De programmeur heeft hier voor de output parameters “call by reference” gebruikt via een *pointer*. In C++ kan dit eenvoudiger door gebruik te maken van *references*. Geef een verbeterde versie van de functie `minmaxElement` die gebruik maakt van **reference** parameters (voor de output parameters).
- B. (5) Hoe moet de verbeterde versie van `minmaxElement` in `main` worden aangeroepen? Geef de **aangepaste** regel(s) code.
- C. (10) De functie `minmaxElement` kan alleen gebruikt worden met array's van het type `int`. Maak een beter herbruikbare versie die met elk **willekeurig type** array gebruikt kan worden.

Zie volgende blad ⇨

KLAS(SEN) : EQ	BLAD : 2 van 4 BLADEN
OND. DEEL : Object Georiënteerd Programmeren in C++	DOCENT : Harry Broeders
AFK. O.E. : OGOPRG	DATUM : -
KWART. : 3	TYPE : tentamen
	TIJD : -

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

- D. (10) De functie `minmaxElement` kan alleen gebruikt worden met array's. In C++ gebruiken we echter vaak een `vector` in plaats van een array. Geef een versie van de functie `minmaxElement` die gebruikt kan worden om het minimum en maximum van een `vector` te bepalen. Het prototype van deze functie is:
- ```
template <typename T>
void minmaxElement(std::vector<T> v, T& max, T& min);
```

## 2. Gegeven is het volgende C++ programma:

```
#include <iostream>
#include <vector>
#include <string>
using namespace std;

class Winkelier {
public:
 Winkelier(const string& n);
 virtual ~Winkelier();
 void stelJeVoor() const;
protected:
 virtual string geefBeroep() const =0;
private:
 const string naam;
};

Winkelier::Winkelier(const string& n): naam(n) {
}
Winkelier::~~Winkelier() {
}
void Winkelier::stelJeVoor() const {
 cout << "Hallo, mijn naam is "
 << naam
 << " en ik ben "
 << geefBeroep() << endl;
}

class Bakker: public Winkelier {
public:
 Bakker(const string& n);
protected:
 virtual string geefBeroep() const;
};

Bakker::Bakker(const string& n): Winkelier(n) {
}
string Bakker::geefBeroep() const {
 return "Bakker";
}
```

**Zie volgende blad ⇨**

---

|                                                     |                         |
|-----------------------------------------------------|-------------------------|
| KLAS(SEN) : EQ                                      | BLAD : 3 van 4 BLADEN   |
| OND. DEEL : Object Georiënteerd Programmeren in C++ | DOCENT : Harry Broeders |
| AFK. O.E. : <b>OGOPRG</b>                           | DATUM : -               |
| KWART. : 3                                          | TIJD : -                |
| TYPE : tentamen                                     |                         |

---

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

---

```
class WarmeBakker: public Bakker {
public:
 WarmeBakker(const string& n);
protected:
 virtual string geefBeroep() const;
};

WarmeBakker::WarmeBakker(const string& n): Bakker(n) {
}
string WarmeBakker::geefBeroep() const {
 return "Warme " + Bakker::geefBeroep();
}

class WinkeliersVereniging {
public:
 void meldLidAan(const Winkelier& w);
 void stelJeAllemaalVoor() const;
private:
 vector<const Winkelier*> leden;
};

void WinkeliersVereniging::meldLidAan(const Winkelier& w) {
 leden.push_back(&w);
}

void WinkeliersVereniging::stelJeAllemaalVoor() const {
 cout << "De leden stellen zich voor:" << endl;
 for (auto lid: leden) {
 cout << "- ";
 lid->stelJeVoor();
 }
}

int main() {
 Bakker Jan("Jan");
 Jan.stelJeVoor();
 WarmeBakker Kees("Kees");
 Kees.stelJeVoor();

 WinkeliersVereniging wv;
 wv.meldLidAan(Jan);
 wv.meldLidAan(Kees);
 wv.stelJeAllemaalVoor();

 cin.get();
 return 0;
}
```

A. (15) Geef de **exacte uitvoer** van dit programma.

**Zie volgende blad** ⇨

---

|                                                     |                         |
|-----------------------------------------------------|-------------------------|
| KLAS(SEN) : EQ                                      | BLAD : 4 van 4 BLADEN   |
| OND. DEEL : Object Georiënteerd Programmeren in C++ | DOCENT : Harry Broeders |
| AFK. O.E. : <b>OGOPRG</b>                           | DATUM : -               |
| KWART. : 3                                          | TYPE : tentamen         |
|                                                     | TIJD : -                |

---

Tijdens dit tentamen mogen **alle** boeken, dictaten, aantekeningen enz. worden gebruikt.

---

- B. (10) Teken een **zo volledig mogelijk** UML klassediagram van het bovenstaande programma.
- C. (10) Geef een **zo volledig mogelijk** UML sequencediagram dat laat zien hoe de volgende regel uit main wordt uitgevoerd:  
`wv.stelJeAllemaalVoor();`
- D. (5) Wat gebeurt er bij het vertalen en het uitvoeren van het bovenstaande programma als de parameter van de constructor van `Winkelier` wordt gewijzigd in `string n` in plaats van `const string& n`? **Verklaar je antwoord!**
- E. (5) Wat gebeurt er bij het vertalen en het uitvoeren van het bovenstaande programma als de parameter van de memberfunctie `meldLidAan` van `WinkeliersVereniging` wordt gewijzigd in `Winkelier w` in plaats van `const Winkelier& w`? **Verklaar je antwoord!**

De functie main wordt nu als volgt aangepast:

```
int main() {
 Bloemist Sonja("Sonja");
 Sonja.stelJeVoor();
 WarmeBakker Kees("Kees");
 Kees.stelJeVoor();

 WinkeliersVereniging wv;
 wv.meldLidAan(Kees);
 wv.meldLidAan(Sonja);
 wv.stelJeAllemaalVoor();

 cin.get();
 return 0;
}
```

- F. (10) Geef **alle code** die aan het programma moet worden toegevoegd om deze main correct te laten werken.