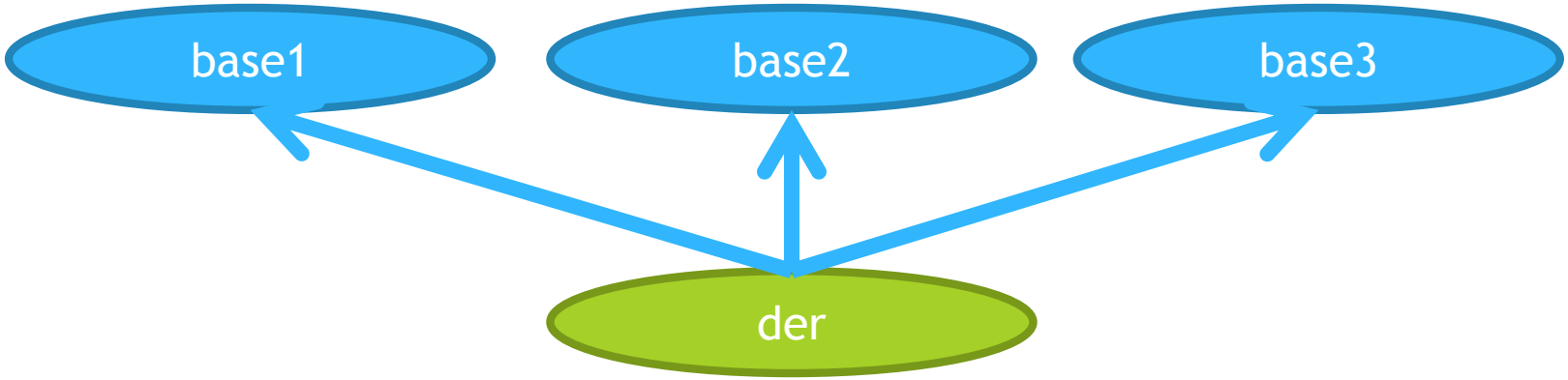
МНОЖЕСТВЕНО НАСЛЕДЯВАНЕ

гл.ас., д-р. Нора Ангелова

Можествено наследяване

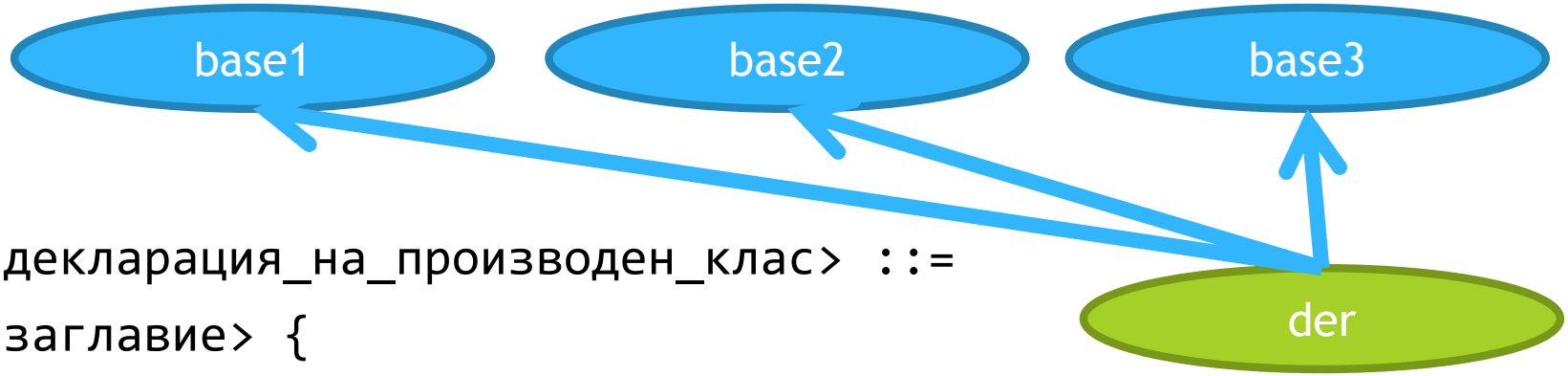
- Декларация на производен клас




```
<декларация_на_производен_клас> ::=
<заглавие> {
    <тяло>
};
```

Можествено наследяване

- Декларация на производен клас



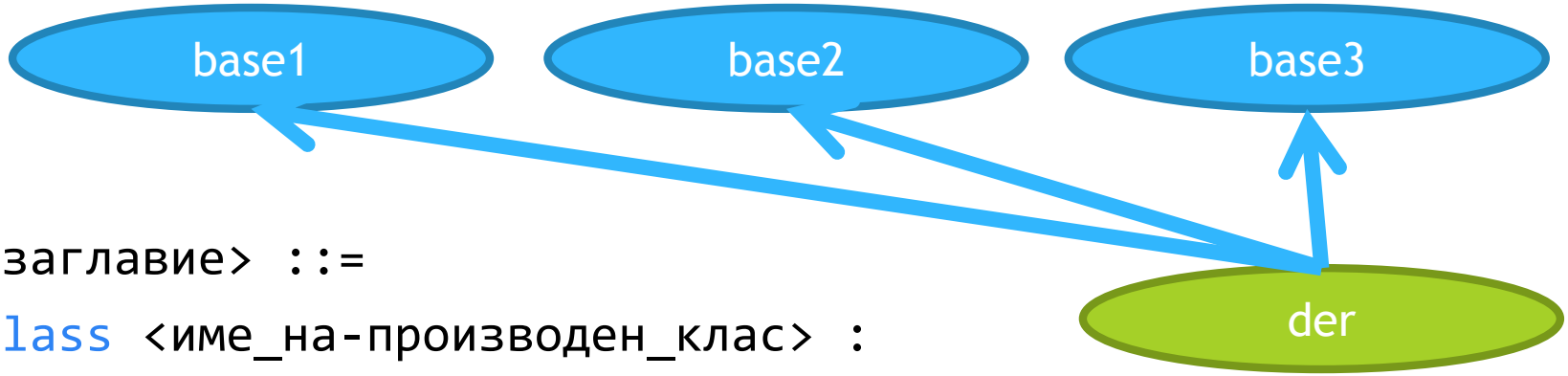

```
<декларация_на_производен_клас> ::=
<заглавие> {
   <тяло>
};

<заглавие> ::=
class <име_на_производен_клас> :
[ <атрибут_за_област> ] <име_на_базов_клас>,
[ <атрибут_за_област> ] <име_на_базов_клас>
{ , [ <атрибут_за_област> ] <име_на_базов_клас> }опц
```

Множествено наследяване

- Декларация на производен клас?
- Какви стойности може да приема атрибутът за област и коя е стойността по подразбиране?

Можествено наследяване

- Декларация на производен клас




```
<заглавие> ::=
class <име_на-производен_клас> :
[ <атрибут_за_област> ] <име_на_базов_клас>,
[ <атрибут_за_област> ] <име_на_базов_клас>
{ , [ <атрибут_за_област> ] <име_на_базов_клас> }опц
```

```
<атрибут_за_област> ::= public | protected | private
```

Множествено наследяване

- Декларация на производен клас?
- Какви стойности може да приема атрибутът за област и коя е стойността по подразбиране?
- Обяснете какви са атрибутите за област в дефиницията:

```
class der : base1, base2, base3 {
   ...
};
```

Как си обяснявате това?

Множествено наследяване

- Производният клас наследява компонентите на всички базови класове като видът на наследяване се определя от атрибута за област на базовия клас.
- Правилата за наследяване при пряк и външен достъп са същите като при единичното наследяване.

Множествено наследяване

- Декларация на производен клас?
- Какви стойности може да приема атрибутът за област и коя е стойността по подразбиране?
- Обяснете какви са атрибутите за област в дефиницията:

```
class der : base1, base2, base3 {
   ...
};
```

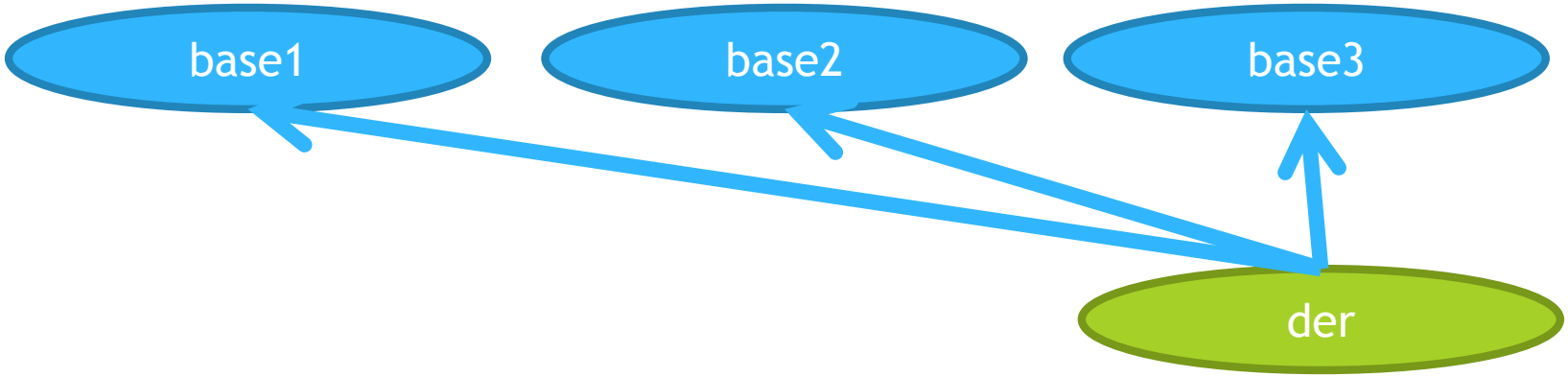
Как си обяснявате това?

- Какви са правилата за наследяване за пряк и външен достъп при единичното наследяване.

МОЖЕСТВЕНО НАСЛЕДЯВАНЕ

- Декларация на производен клас

```
public:    int b11      public:    int b21      public:    int b31
private:   int b12      private:   int b22      private:   int b32
protected: int b13      protected: int b23      protected: int b33
```




```
class der : base1, protected base2, public base3 {
        public: …
        private: int d2;
        protected: …
};

der d;
```

Множествено наследяване

- За член-функциите на голямата четворка на производен клас с множествено наследяване са в сила аналогични правила, като при производен клас с единично наследяване. В общия случай тези член-функции на основните класове не се наследяват от производния им клас.

Множествено наследяване

- Декларация на производен клас?
- Какви стойности може да приема атрибутът за област и коя е стойността по подразбиране?
- Обяснете какви са атрибутите за област в дефиницията:

```
class Der : Base1, Base2, Base3
{ ...
};
```

Как си обяснявате това?

- Какви са правилата за наследяване, за пряк и външен достъп при единичното наследяване?
- Какви са правилата за член-функциите на голямата четворка на производен клас с единично наследяване?

Множествено наследяване

- Конструктор

При извикването на конструктор на производен клас последователно се изпълняват:

1) Конструкторите на базовите му класове в реда на тяхното задаване в декларацията на производния клас, а не в инициализиращия списък на конструктора.

   Ако за някой основен клас не е посочен конструктор в инициализиращия списък, изпълнява се конструкторът по подразбиране на класа, ако такъв е дефиниран (или може да се създаде), или се съобщава за грешка.

2) Конструкторите по подразбиране на класовете, чиито обекти са член-данни на производния клас, в случай че в инициализиращият списък не е указано как да се инициализират. Редът на извикване съответства на реда на деклариране на тези член-данни в тялото на производния клас;

3) Тялото на конструктора на производния клас.

Множествено наследяване

Конструктор

- В някой от основните класове не е дефиниран конструктор в т.ч. за присвояване (такъв може да се генерира)

1. Ако в производния клас има дефиниран конструктор в инициализиращия списък не трябва да се прави обръщение към конструктор на основния му клас и наследената му част остава неинициализирана.

Множествено наследяване

Конструктор

- В някой от основните класове е дефиниран конструктор с параметри, от който не следва подразбиращият се конструктор

1. Ако в производния клас е дефиниран конструктор, в инициализиращия му списък задължително трябва да има обръщение към конструктора с параметри на този основен клас.
2. Ако в производния клас не е дефиниран конструктор, компилаторът ще съобщи за грешка.

Множествено наследяване

Конструктор

- В някой от основните класове са дефинирани няколко конструктора в т.ч. подразбиращ се

1. Ако в производния клас е дефиниран конструктор, в инициализиращия му списък може да не се посочи конструктор за този основен клас. Ще се използва подразбиращият се конструктор на основния клас.
2. Ако в производния клас не е дефиниран конструктор, компилаторът автоматично създава за него подразбиращ се конструктор.
   В този случай всички основни класове на производния клас трябва да имат конструктори по подразбиране.

Множествено наследяване

- Деструктор

Всеки деструктор трябва да разруши само онези собствени компоненти, които са реализирани в динамичната памет.

Извикването на деструкторите на базовите класове и производния им клас се осъществява автоматично в следната последователност:

1) извиква се деструкторът на производния клас,
2) в обратен ред, се извикват деструкторите на класовете на обектите, които са член-данни на производния клас (ако има такива),
3) изпълнявят се деструкторите на основните му класове, отново в обратен ред на реда на извикване на техните конструктори.

Множествено наследяване

- Конструктор за копиране

Извиква конструктор за копиране или друг подходящ, с който да се инициализират наследените член-данни

```
<инициализиращ_списък> ::=
<празно> |
: <име_на_основен_клас>(p)
{ , <име_на_основен_клас>(p) }
{ , <член-данна>(<параметри>) }
```

Множествено наследяване

Конструктор за копиране

- В производния клас не е дефиниран конструктор за копиране

Тогава компилаторът автоматично генерира за него конструктор за копиране, който преди да се изпълни активира и изпълнява конструкторите за присвояване (копиране) на всички основни класове в реда, указан в декларацията на производния клас.

В този случай конструкторите за присвояване (копиране) на основните класове се наследяват от производния клас.

Множествено наследяване

Конструктор за копиране

- В производния клас е дефиниран конструктор за копиране

Препоръчва се в инициализиращия му списък да има обръщения към конструкторите за присвояване на основните класове (ако такива са дефинирани).

Ако за някои основен клас не е указано такова обръщение, а е указан обикновен негов конструктор, инициализирането на наследените член-данни на този клас става чрез указания конструктор.

Ако не е указано обръщение към конструктор за някой от основните класове, използва се конструкторът по подразбиране на основния клас, ако такъв съществува или се съобщава за отсъствието на подходящ конструктор за този основен клас, ако в него не е дефиниран конструктор по подразбиране.

МНОЖЕСТВЕНО НАСЛЕДЯВАНЕ

- **Операторна функция за присвояване**

Присвояването на наследените член-данни става в тялото на операторната функция.

```
<производен_клас>& <производен_клас>::operator=
(const <производен_клас>& p) {
 if (this != &p) {
  // Дефиниране на присвояването за наследените член-данни
  <основен_клас1>::operator=(p);
  <основен_клас2>::operator=(p);

  // Дефиниране на присвояването за собствените член-данни
  Del(); // разрушаване на онези собствени член-данни на подразбиращия
         // се обект, които са разположени в ДП

  Copy(p); // копиране на собствените член-данни на p в съответните
           // член-данни на подразбиращия се обект
 }
 return *this;
}
```

Множествено наследяване

Операторна функция за присвояване

- В производния клас не е дефинирана операторна функция за присвояване

Тогава компилаторът създава такава. Тя изпълнява операторните функции за присвояване (дефинирани или генерирани от компилатора) на всички основни класове на производния клас.

Множествено наследяване

Операторна функция за присвояване

- Ако в производния клас е дефинирана операторна функция за присвояване, тя трябва да се погрижи за присвояването на всички наследени член-данни.

Ако това не е направено явно за някой основен клас, стандартът на езика не уточнява как ще стане присвояването на наследените от този клас член-данни.

КРАЙ