دانشگاه تهران
دانشکده مهندسی برق و کامپیوتر

وراثت و چندریختی — ۷

چندریختی

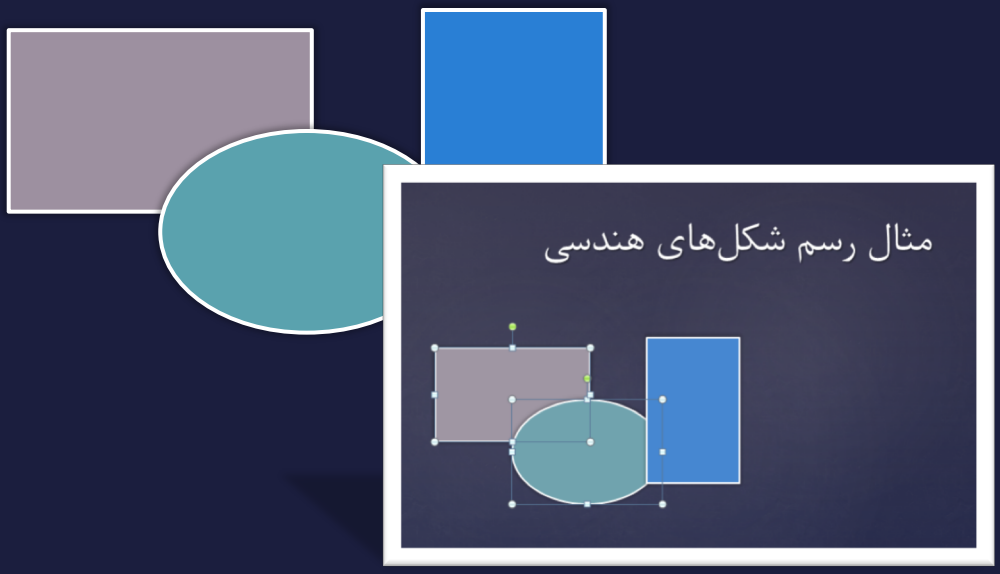
برنامه‌سازی پیشرفته — رامتین خسروی

بهار ۹۹

polymorphism

چندریختی

مثال رسم شکل‌های هندسی

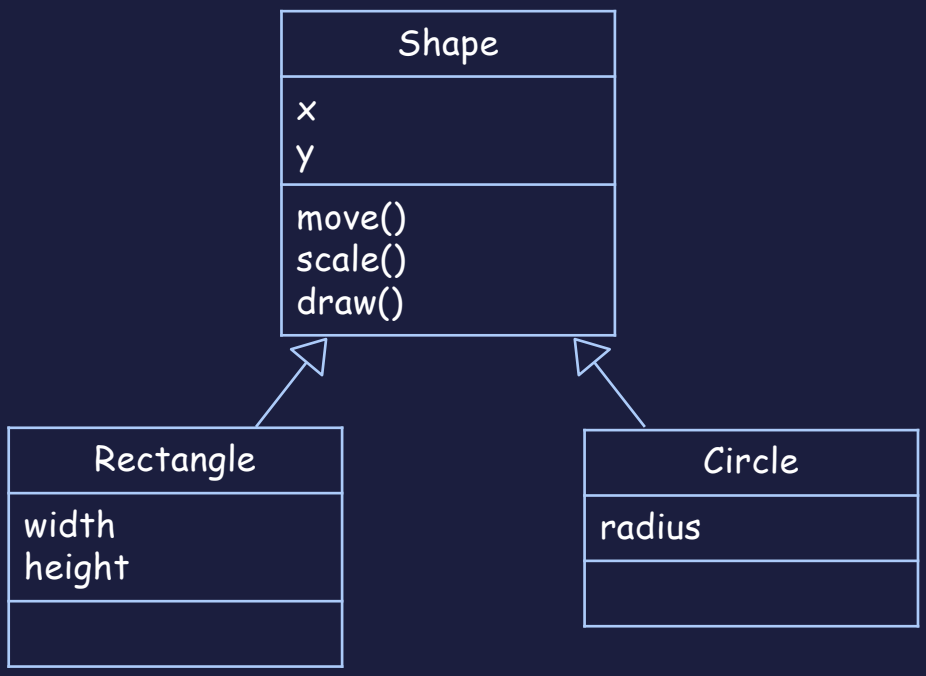


کلاس‌های متناظر با شکل‌ها

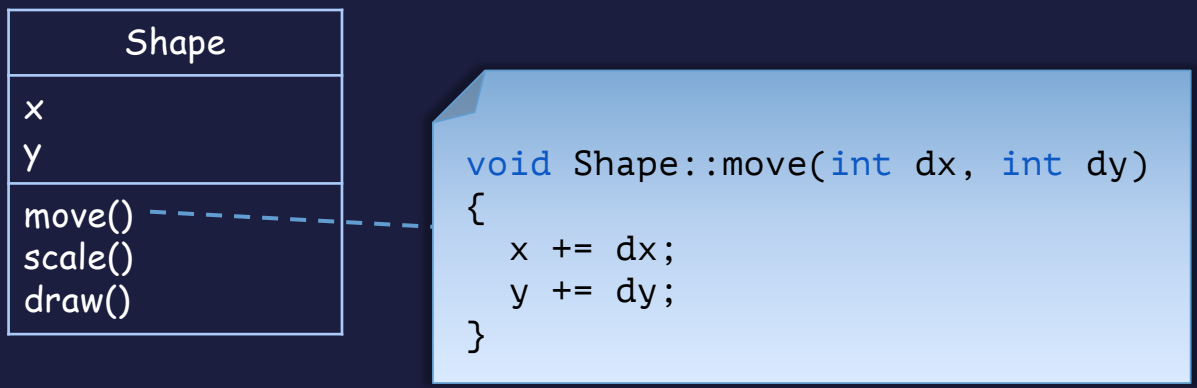
Rectangle
x y width height
move() scale() draw()

Circle
x y radius
move() scale() draw()

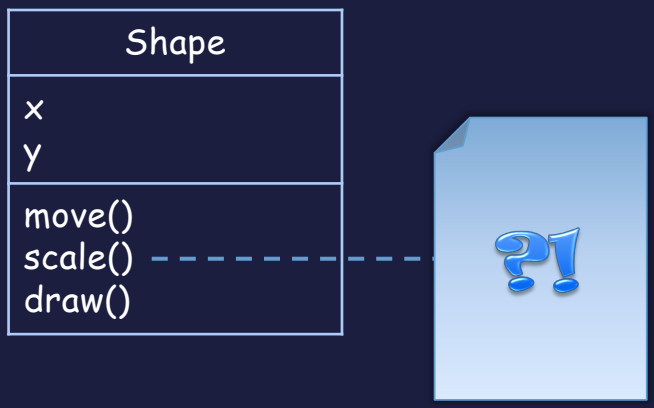
فاکتورگیری اعضای مشترک



پیاده‌سازی move

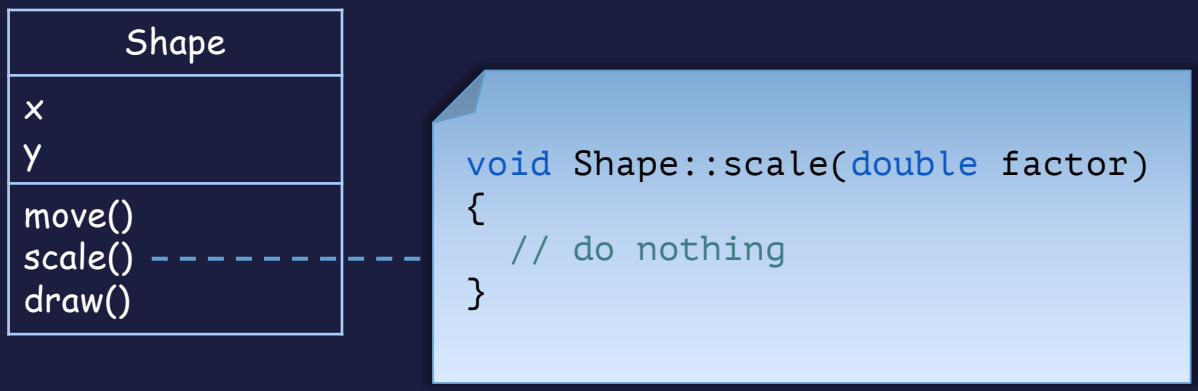


پیاده‌سازی scale



پیاده‌سازی این متد در کلاس Shape قابل انجام نیست و باید توسط زیرکلاس‌ها صورت‌پذیرد.

پیاده‌سازی scale



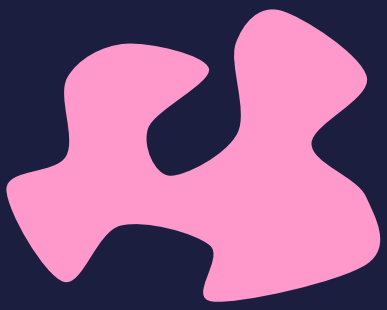
نظیر همین موضوع در مورد draw وجود دارد.

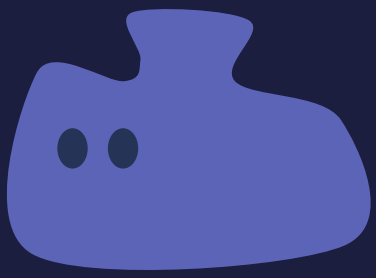
استفاده از Shape

یک شکل!

```
Shape s(10, 20);
s.move(-1, 2);
s.scale(3);     // does nothing
```

اصولاً یک شکل چه شکلی است؟!





کلاس‌های مجرد (abstract)

Shape
x y
move() *scale()* *draw()*

```
class Shape {
public:
  Shape(int, int);
  void move(int dx, int dy);
  virtual void scale(double s) = 0;
  virtual void draw() = 0;
};
```

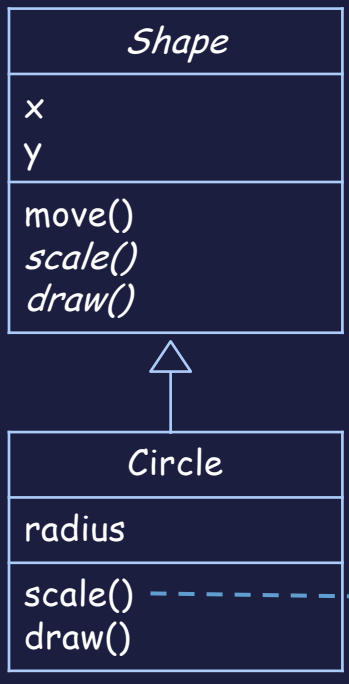
کلاس مجرد:
کلاسی که حداقل یک متد مجازی خالص داشته باشد.

متد مجازی خالص (pure virtual):
متدی که بدنه ندارد و پیاده‌سازی آن به زیرکلاس‌ها واگذار می‌شود.

ساختن شیء از کلاس مجرد ممکن نیست!

```
Shape s(10, 20);  ← compile error!
s.move(-1, 2);
s.scale(3);     // does nothing
```

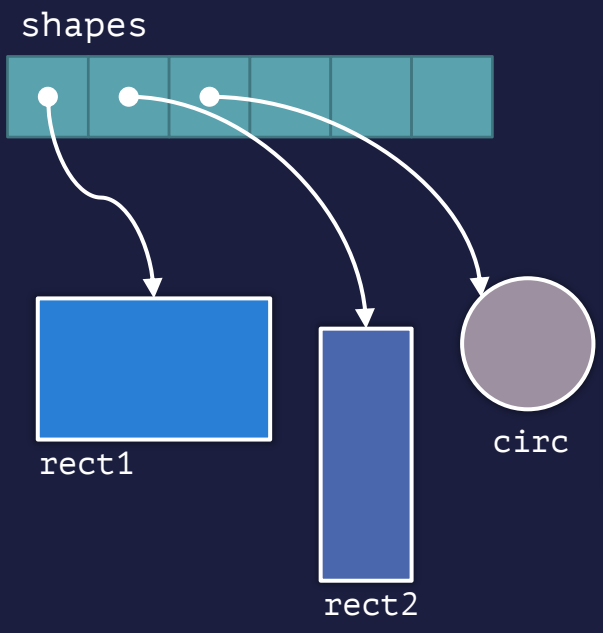

مسئولیت زیرکلاس‌ها



زیرکلاس‌ها باید متدهای مجرد را پیاده‌سازی کنند،
وگرنه خود نیز مجرد محسوب می‌شوند.

```
void Circle::scale(double factor)
{
  radius *= factor;
}
```

نگهداری شکل‌ها

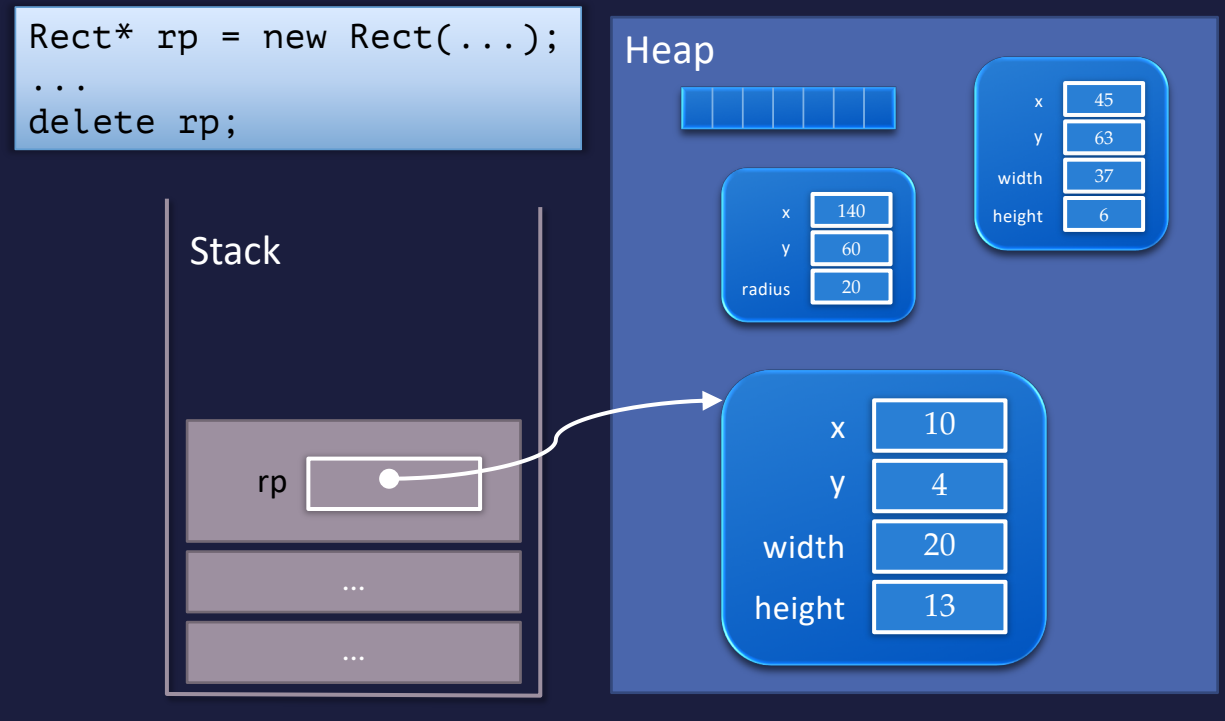



```
vector<Shape*> shapes;

Rect rect1(10, 7, 3, 2);
Rect rect2(3, 15, 1, 4);
Circle circ(3, 1);

shapes.push_back(&rect1);
shapes.push_back(&rect2);
shapes.push_back(&circ);
```

مشکل: لزوماً نمی‌دانیم کاربر برنامه چندتا مستطیل و چندتا دایره قرار است رسم کند.

تخصیص حافظه‌ی پویا

```
Rect* rp = new Rect(...);
...
delete rp;
```



آزاد کردن حافظه‌ی تخصیص‌یافته

- چرا باید حافظه‌ای را که به طور پویا می‌گیریم حتماً آزاد کنیم؟
  - استفاده بهینه از حافظه در طول اجرا
  - اتکا نکردن به سیستم عامل برای آزاد کردن حافظه پس از اتمام اجرا

انضباط در مدیریت حافظه

- حافظه‌ای را که به طور پویا گرفته‌اید را حتماً آزاد کنید.
- حافظه‌ای را که با new گرفته‌اید با delete و حافظه‌ای را که با malloc گرفته‌اید با free آزاد کنید
- حافظه‌ای را دوبار آزاد نکنید!

تمرین عملی

در شرکت الف دو نوع قرارداد کار وجود دارد: ساعتی و تمام‌وقت. یک کارمند ساعتی دستمزد ثابتی برای هر ساعت کار خود دارد و دریافتی آن از حاصل‌ضرب این دستمزد در ساعت کارکرد به دست می‌آید. یک کارمند تمام‌وقت یک حقوق ثابت برای ۱۴۰ ساعت کارکرد در ماه دارد و اگر بیش از این مقدار کار کند به اندازه ۴۰ درصد اضافه‌کار به او تعلق می‌گیرد. مثلاً اگر حقوق ثابت یک کارمند ۲.۸۰۰.۰۰۰ تومان باشد، یعنی دریافتی او ساعتی ۲۰.۰۰۰ تومان است. اگر او در یک ماه ۱۸۰ ساعت کار کرده باشد مبلغ ۲.۸۰۰.۰۰۰ ساعت حقوق پایه و ۱.۴×۴۰×۲۰.۰۰۰=۱.۱۲۰.۰۰۰ تومان اضافه‌کار دریافت می‌کند و دریافتی کل او ۳.۹۲۰.۰۰۰ خواهد شد.

یک کلاس مجرد Employee را به همراه دو زیرکلاس آن یعنی FullTime و Hourly تعریف کنید. برای این کلاس‌ها متدی با امضای double earnings(int hours) تعریف کنید که دریافتی یک ماه کارمند را محاسبه کند.

17

تمرین عملی

کلاسی به نام OrgUnit تعریف کنید که معادل یک واحد سازمانی است. یک واحد سازمانی شامل تعدادی کارمند و تعدادی واحد سازمانی زیرمجموعه است. برای این کلاس متدی با امضای double total_earnings(int avg_hours) تعریف کنید که مجموع دریافتی تمام کارمندان (شامل تمام کارمندان واحدهای زیرمجموعه) را برگرداند، اگر تمام کارمندان avg_hours ساعت کار کرده باشند.

18