هوالحکیم

گزارش تمرین سری چهارم برنامه نویسی پیشرفته

محمدرضا حلاجیان 9423043

سوال 1:

Move Semantics

با استفاده از این روش به جای کپی اطلاعات بین دو شیء ، اطلاعات را جا به جا (move)می کنیم. باید توجه داشت که علی رغم سرعت بیشتر این روش و جلوگیری از کپی بی مورد، داده ای در شیء اولیه باقی نخواهد ماند!

Polymorphism

اگر تایعی با اسم مشابه در چند کلاس داشته باشیم که خود از یک کلاس دیگر مشتق می شود، از این متد استفاده کرده ایم. یعنی تابع با توجه به تعریف مختص خود در کلاس اجرا می شود.

Pure Abstract

وقتی کلاسی را به صورت Pure Abstract تعریف می کنیم، توابع را در آن فقط به صورت یک پروتوتایپ ساده می نویسیم و تعاریف را در هر کلاس مشتق شده از این کلاس متانسب با مفهوم مورد نظر در آن کلاس ها انجام می دهیم.

Override

وقتی که یک کلاس parent و یک کلاس child داشته باشیم، به این معنا که کلاس child از parent ارث برده باشد، اگر یک تابع را در parent تعریف کنیم و همان تابع را در child نیز تعریف کنیم عملیات override را انجام داده ایم. به این معنا که اگر آبجکتی از جنس child بسازیم ، و تابع مربوط را صدا کنیم تابع واقع در child اجرا می شود. اما اگر آبجکتی با کانستراکتور child اما با نوع parent (parent obj = child()) بسازیم، به تابعی که در parent واقع است اشاره می کند تابع واقع در child را overriding و تابع واقع در parent را overridden می نامند.

Inline

اگر تابعی را inline تعریف کنیم، یک کپی در هر نقطه از فراخوانی تابع وجود خواهد داشت، اگر کد تابع تغییر کند باید یک بار دیگر کد اجرا شود تا این کپی ها جایگزین شوند.

Explicit

این اسم در تعریف کانستراکتور کلاس استفاده دارد بدین صورت که باعث می شود در تعریف کردن آبجکت محدودیت های به وجود بیاید برای مثال اگر یک کلاس box با ورودی کانستراکتور int داشته باشیم که explicit نیز هست، برای ساخت ابجکت باید بنویسیم : box A = box(1) ، حال آن که اگر explicit نبود می توانستیم بنویسیم : box A = 1 . در حقیقت از explicit از Implicit conversion جلوگیری می کند.

سوال 2:

در روش اول با افزودن هر المان به وکتور، resize شده و size و capacity همواره تغییر میکند. اما در روش دوم ابتدا وکتوری با ظرفیت 1000 می سازیم و در هنگام اضافه کردن هر المان، تنها به سایز وکتور اضافه شده و capacity ثابت می ماند چون 1000 تا برای آن رزرو کرده ایم از قبل.

سوال 3:

کپی شده!

سوال 4:

در حالت کلی هرگاه بخواهیم تابعی در کلاس مشتق شده از یک کلاس دیگر تعریف کنیم، تابع را در کلاس مرجع virtual تعریف می کنیم و در کلاس های مشتق شده از آن override می کنیم. در سوال قبل چون توابع area و volume در کلاس های مرجع(والد) خود هم تعریف و هم استفاده شده اند نیازی به تعریف کردن آنها به صورت virtual نیست.

سوال 5:

ابتدا کلاس های stack و ctext تعریف شده اند. کلاس stackمطابق ساختار معمول برای پشته ها تعریف شده و کلاس ctext هم مشابه کلاس string است. توابع pop و push هم مطابق آنچه در کد main خواسته شده تعریف گردید.

سوال 6:

در بخش اول ابتدا مقادیر 2 موجود، حذف شد اما سایز تغییری نکرد! بقیه ی موارد هم در کد با کامنت مشخص شده اند.

سوال 7:

بخش های مختلف سوال با کامت هایی از یکدیگر جدا شده اند. برای نمایش وکتور و آرایه دو تابع Disp و DispArray تعریف شده اند که دستورات آنها در تمام بخش ها کامنت شده اند. برای مشاهده خروجی هر بخش لطفاً آنها را از شکل کامنت خارج کنید.

لینک Git