روش شیء گرا:

در بخش قبل " پیمانه ای " برنامه نوشتن توضیح داده شد، برای آنکه گام فراتر نهیم و باز هم نوشتن برنامه ها ساده تر شود علاوه بر آن مزایا دیگری در برنامه نویسی بدست می آوریم می توانیم شیءگرا[1] بیندیشیم و برنامه بنویسیم. احتمالاً این بخش خیلی واضح نخواهد بود برای روشن تر شدن بحث باید تا 6-6 صبر کنید در این طرز تفکر علاوه بر آنکه کار به بخشهای کوچکتر تقسیم می شود، به هر بخش کار بصورت یک شی (موجودیت) نگریسته می شود، به این معنی که هر بخش کار همچون موجودیت هایی که با آنها در زندگی روزمره سرو کار داریم خواصی دارند و افعالی مثلاً یک چرخ گوشت وزن و جرم و ... (خواص) دارد و گوشت را خرده خرده می کند (افعال). این طرز تفکر باعث می شود که به سادگی برخوردهای روزمره با اجزای کوچک برنامه (تفکیک های انجام شده) کار کنیم یعنی نتیجه مورد نظر خود را از تعامل چند موجودیت بدست آوریم.

از توصیف فلسفی بالا که بگذریم بصورت ساده می توان گفت در روش شیئی گرا علاوه بر تفکیک راه حل کلی به بخشهای کوچک خود این بخشها را طبقه بندی کرده و به هر طبقه نامی می دهیم این کار باعث جلوگیری از مخلوط شدن اسامی، زیربرنامه ها و کلا ساده تر شدن برنامه نویسی می گردد.

احتمالاً بسیاری از شما فیلم های سه گانه ماتریکس[2] را دیده اید. ایده این فیلم هم دقیقاً منطبق بر شیئی گرایی است شیئی که زمین شبیه سازی می کند، شیئی که درختها را مدیریت می کند و ...

ایده این فیلم هم دقیقاً منطبق بر شیئی گرایی است شیئی که زمین را شبیه سازی می کند، شیئی که درختها را مدیریت می کند و ... با تصور برنامه بودن هر کدام از این اشیا عملاً یک برنامه شیئی گرا را تصور

کرده اید. با این متد در تجرید انواع داده (یعنی تولید یک ساختار داده ذهنی جدید) هم پیشرفت شایان ذکری ایجاد شده است.

این نوع برنامه نویسی هم برخی کارها را ممکن کرده است که سابقاً بسیار مشکل قابل انجام بود. امروزه مسایلی نظیر تعامل موجودات زنده تک سلولی در محیطی خاص را هم می توان با روش شیئی گرایی شبیه سازی کرد. بدین صورت که یک شیئی محیط و یک شیئی تک سلولی با تمام مشخصات و افعالشان طرح می شود سپس چند نمونه از شیئی تک سلولی و یک نمونه از شیئی محیط زیست ایجاد شده و ادامه برنامه اجرا می شود و اتفاقی که در اثر سالها تعامل تک سلولیها قرار است اتفاق بیفتد پس از چند ساعت اجرای برنامه پیش بینی می شود. کلاً شبیه‌سازیهای [3]پیچیده با این روش راحت تر قابل انجام است.

نحوه پیاده سازی این روش و اصولاً نحوه کار زبان های سطح بالا را توضیح دادن از حوصله این متن بیرون است و لذا فقط نکات کلیدی روشها و مفاهیم که احتمالاً کاربرد آنها وجود بیان می شود.

از دیگری مزایای شیئی گرایی ارث بری است بدین معنا که در نوشتن برنامه و طراحی اشیاء متفاوتی که بخش قابل توجهی در اشتراک دارند می توان بخش مشترک را در یک کلاس والد نوشت و اشیاء متفاوت نام برده را از آن به ارث برد، به این ترتیب بخش مشترک فقط یک بار پیاده سازی می شود و حجم کار کاهش می یابد، مسلما اشیا به ارث برده شده تمامی خصوصیات شیء والد و افعال آن را دارند.

مقوله ارث بری و سایر مشخصات شیئی گرای فوائد زیاد دیگری دارند که به علت کاربرد کم این مفاهیم در برنامه نویسی در المپیاد از ذکر آنها صرف نظر می کنیم.

ObjectOriented [1]

The Matrix [2]

Simulation [3]