

# پیاده سازی لایه تجرید برای نمایش و استفاده از سیستم‌های عددی

آرش طاهرکلاته ۸۸۲۹۸۰

هدف از این برنامه شبیه‌سازی آنچه بود که همانند آن را ما با زبان برنامه‌نویسی توصیفی اسکیم انجام دادیم، ساختن یک Interface یکسان برای کار با اعداد.

زبانی که برای این کار انتخاب شد پایتون بود، چرا که Dynamic Typing بودن آن و نحوه رفتار آن با توابع به صورت مرتبه اول، سهولت اجرای چنین برنامه‌ای را دوچندان میکرد.

و اما نحوه پیاده‌سازی : ساختار پایه سیستم در دو ماژول به نام‌های table و basePackage تعریف شده است. در ماژول table صرفاً یک دیکشنری تعریف شده است که این دیکشنری با توابع مورد نظر ما ( که در ادامه خواهیم دید ) پر خواهد شد. در کنار آن نیز عملیات روی این جدول، با نام‌های put ( که تابع جدید را در جدول قرار میدهد ) و get ( که بر حسب نوع عملیات و عملوندها تابع مناسب را برمیگرداند ) تعریف شده است .

در ماژول basePackage توابع ما به صورت عمومی تعریف شده‌اند. در واقع در اینجا واسطی که کاربر با آن سر و کار دارد را تعریف کرده‌ایم. این توابع، که نام عملگرها را بر خود دارند، با دریافت عملوندها و تشخیص نوع آنها، عملیات مناسب را از جدول واکنشی کرده و نتیجه مطلوب را بازمیگردانند.

در کنار این دو ماژول، پیاده‌سازی خود سیستم‌های عددی مورد نظر را داریم : اعداد صحیح، گویا، موهومی و چند جمله‌ای. برای هر کدام از این انواع، یک ماژول جداگانه به وجود آورده‌ایم. در این ماژول‌ها یک ساختمان داده مختصر برای نگاه داری داده‌های مربوط به این نوع داده تعریف شده است. در ادامه هم توابع جمع، تفریق، ضرب و تقسیم برای این نوع داده تعریف شده‌اند. در کنار همه اینها یک تابع نصب بسته هم در این ماژول قرار دارد که تمام این توابع را، با برچسب مناسب در جدول قرار میدهد.

این تمام چیزی بود که لازم داشتیم. حال برای استفاده، ماژول‌های متناظر با سیستم ریاضیاتی مورد نظر را با برنامه خود اضافه می‌کنیم. پس از نصب این بسته‌ها، می‌توان فارغ از نوع، توابع عمومی را روی آنها صدا زد. از آنجا که در کلاس هر کدام هم توابع \_\_str\_\_ و مقایسه‌ای بازنویسی شده است، نگرانی از بابت نمایش و مقایسه هم بی‌مورد شده و می‌توان از داشتن یک لایه انتزاع یکپارچه بر روی ساختارهای عددی لذت برد .