



دانشگاه صنعتی اصفهان

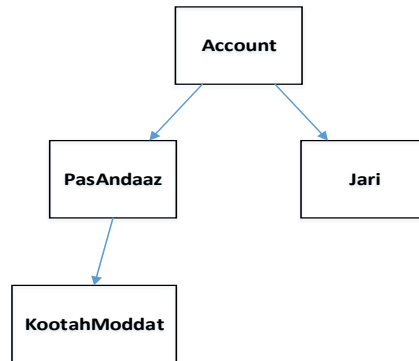
آزمون پایان ترم

درس برنامه‌سازی پیشرفته و آزمایشگاه

نیمسال دوم سال تحصیلی ۱۴۰۱-۱۴۰۰

زمان: ۱۸۰ دقیقه

۱. سلسله مراتب زیر را در نظر بگیرید. (۵۰ نمره)



کلاس حساب (Account) را در نظر بگیرید. فرض کنید این کلاس یک کلاس انتزاعی است که یک عدد double به عنوان mojoodi دارد که سطح دسترسی آن private است. این کلاس برای set و get کردن این مقدار باید تابع‌های مناسب داشته باشد.

کلاس PasAndaaz باید یک عضو داده‌ای از نوع double به نام percent با سطح دسترسی‌ای داشته باشد که کلاس KootahModdat بتواند آن را بخواند یا تغییر دهد؛ اما خود PasAndaaz در هیچ تابعی امکان مقداردهی این مقدار را فراهم نکند (قرار است این کار را کلاس‌های مشتق انجام دهند، پس کافیست در سازنده‌ی این کلاس مقدار آن را صفر کنید).

کلاس KootahModdat مقدار percent را به ۰/۱۱ مقداردهی می‌کند.

همه کلاس‌ها باید تابع محاسبه etebar را داشته باشند. کاری کنید که هر جا این تابع پیاده‌سازی شد نتواند چیزی را تغییر دهد. در Jari این تابع صرفاً مقدار mojoodi را برمیگرداند و در کلاس PasAndaaz این تابع مقدار موجودی به اضافه سود موجودی (* percent mojoodi) را باز میگرداند.

عملگر + را برای کلاس KootahModdat به نحوی پیاده‌سازی کنید که حاصل جمع etebar دو شیء از این نوع را بازگرداند.

تابع main را برای تست برنامه به صورت زیر بنویسید:

یک لیست (یا وکتور) متناسب با Account بسازید که بتوانید با پیمایش آن و استفاده از کاربردهای چندریختی، etebar یک حساب جاری (عضو اول) و دو حساب KootahModdat (دو عضو بعدی) که مقدار دهی شده‌اند را چاپ کند. پیش از چاپ کردن etebar برای حساب‌های KootahModdat مقدار percent برای آن‌ها را به ۰/۱۴ افزایش دهید.

۲. به کمک Qt برنامه‌ای بنویسید که: (۵۰ نمره)

- یک پنجره اصلی دارد. خروج از آن کل برنامه را پایان می‌دهد.
- کلاس مربوط به این پنجره اصلی از هر ۳ container اصلی priority_queue و queue و stack یک شیء از نوع int دارد.
- محتویات هر کدام از این‌ها را به هر نحوی که صلاح میدانید در یک ابزارک گرافیکی نمایش می‌دهد.

- متناظر با هر کدام از این‌ها یک spinbox و یک دکمه دارد. (می‌شود ۳ تا از هر کدام) جایش هم در نزدیکی ابزارک متناظر است. خواه بالا، پایین، چپ یا راست.
- دکمه متناظر stack مقدار متناظر را به بالای stack اضافه می‌کند. دکمه متناظر queue مقدار متناظر را به انتهای آن اضافه می‌کند و دکمه متناظر priority_queue مقدار متناظر را در آن insert می‌کند.
- یک container دیگر و یک ابزارک گرافیکی دیگر دارد که متناظر با داده‌هایی است که محتوای آن‌ها در قسمت بعد مشخص می‌شود.
- در منوی فایل علاوه بر گزینه خروج یک Action با عنوان transfer دارد که در slot مربوطه عملیات زیر در یک thread مجزا انجام می‌شود:

i. هر بار ۳ داده‌ی اول این سه container را pop می‌کند. با ۲ داده مربوط به priority_queue و queue یک زوج مرتب به شکل (priority_queue و queue) می‌سازد. حال این ۳ داده را در یک container دیگر درج می‌کند به نحوی که:

- داده‌ی مستخرج از stack کلید مرتب سازی در زمان insert باشد. (کوچک به بزرگ)
- اگر در stack دو داده‌ی برابر وجود داشت، زوج مرتب‌های متناظر آن‌ها در زمان درج به ترتیب عنصر اول زوج، مرتب قرار گیرند. (بزرگ به کوچک).
- مثلاً اگر در stack مقادیر ۱۰، ۹، ۹، ۱۱ قرار داشت، و در queue ۱۲، ۱۳، ۸، ۶، ۲۳ و priority_queue مقادیر ۱۳، ۱۲، ۱۰، ۶، ۱ قرار داشتند، داده‌ها به نحو زیر در ساختمان جدید قرار می‌گیرند.

- 9 (8,10)
- 9 (6,6)
- 10 (13,12)
- 10 (12,13)
- 11 (23,1)

- به صورت ساده‌تر یعنی داده‌ها ابتدا بر اساس داده خروجی از stack باید مرتب باشند و برای داده‌هایی که کلید برابر دارند بر اساس داده‌ی مستخرج از queue (عنصر اول زوج مرتب).
- اگر تعداد داده‌ها در ۳ container اصلی با هم برابر نبود slot مورد استفاده باید یک exception از نوع کلاس‌های استاندارد صادر کند که در همان تابع مدیریت می‌شود و در یک messagebox پیام مناسبی نشان بدهد.

پی‌نوشت» پیاده سازی containerها در هیچ کدام از سوالات لازم نیست. شما فقط آن را استفاده می‌کنید.

پی‌نوشت» در C++ زوج مرتب‌ها قابل مقایسه‌اند و عنصر اول اهمیت بیشتری دارد.

پی‌نوشت» هر چند با ترکیب دو container، درج فوق به صورت مرتب خیلی ساده انجام می‌شود؛ اما توصیه می‌کنم ابتدا به فکر ترتیب گفته شده نباشید. تنها ۱۰ نمره دارد پس سعی کنید اگر وقت اضافه آوردید به این قضیه فکر کنید.

پی‌نوشت» ظاهر برنامه مهم نیست، نحوه نمایش مهم نیست. البته تعمداً بد سلیقه نباشید.

پی‌نوشت» داشتن warning در زمان کامپایل ۱۰ درصد و کامپایل نشدن ۳۰ درصد کسر نمره سوال را در پی دارد.

پیروز باشید.