



UT Bank

در این تمرین کامپیوتری از شما خواسته شده است که یک سامانه بانکی ساده را شبیه سازی کنید.

در این تمرین کامپیوتری با مفاهیم Struct, Linked List, تخصیص حافظه پویا (Dynamic Memory Allocation)، خواندن و نوشتن در فایل بیشتر کار خواهیم کرد.

کاربران سامانه و دسترسی های ایشان

در این سامانه بانکی دو نوع کاربر وجود دارند: 1) کاربر مدیر (admin) 2) کاربر عادی
هر کاربر نام کاربری متفاوت و یکتایی باید داشته باشد.
هرکدام از این دو نوع کاربر اختیارات متفاوتی از یکدیگر دارند که در ادامه توضیح داده می شود.

کاربر مدیر

کاربر مدیر (admin) از ابتدا و بدون نیاز به ثبت نام در سامانه باید وجود داشته باشد. این کاربر یکتا بوده و به نوعی اداره کننده این سامانه است. نام و نام کاربری مدیر admin بوده، password مربوط به آن 123 و شماره تلفن مربوط به آن 000000000000 می باشد.

اختیارات این کاربر عبارتند از:

- 1- ثبت نام یک کاربر جدید
- 2- اضافه کردن موجودی به حساب کاربران
- 3- برداشت موجودی برای کاربران
- 4- مشاهده اطلاعات یک کاربر
- 5- حذف کردن حساب یک کاربر

کاربر عادی

هر کاربر جدیدی که به این سامانه توسط مدیر اضافه می شود از این دسته بوده و از این اختیارات برخوردار است:

- انتقال موجودی به حساب یک کاربر دیگر
- مشاهده اطلاعات حساب خود

ذخیره در فایل

کلیه اطلاعات، اعم از اطلاعات کاربران، نام کاربری ایشان، پسورد ها و تراکنش های همه کاربران در یک یک فایل ذخیره می شود.

برای آنکه اطلاعات کاربران با بستن برنامه از بین نرود باید تمام اطلاعات کاربران به جز کاربر مدیر (که ثابت بوده و هنگام شروع اجرای برنامه ایجاد می شود) هنگام خروج از برنامه در این فایل ریخته شده و هنگام اجرا شدن مجدد برنامه اطلاعات قبلی ذخیره شده در این فایل خوانده شده تا ادامه روند اجرای برنامه به نحوی باشد که انگار برنامه هیچگاه بسته نشده بود.

پیشنهاد می شود برای سادگی از یک فایل با نوع تکست برای ذخیره سازی استفاده نمایید. انتخاب ساختار این فایل با شما است، ولی اطلاعات زیر باید حتما در آن ذخیره شوند.

برای هر کاربر بجز کاربر مدیر (admin) باید اطلاعات زیر ذخیره شوند.

- 1- نام کاربری
- 2- رمز عبور
- 3- شماره تلفن
- 4- شناسه یکتا
- 5- مانده حساب
- 6- تعداد تراکنشها
- 7- فهرست تراکنشهای کاربر از زمان اضافه شدن به سامانه تا کنون.

هر تراکنش باید حاوی اطلاعات زیر باشد:

- 1- نوع تراکنش: انتقال به حساب دیگر، دریافت از حساب دیگر، واریز به حساب، برداشت از حساب
- 2- مقدار تراکنش

طبیعی است که بدلیل مشخص نبودن و متغییر بودن تعداد کاربران سامانه و همچنین مشخص نبودن و متغییر بودن تعداد تراکنشها باید از لیستهای پیوندی (Linked List) ها برای پیاده سازی آنها استفاده نمود. انتخاب مناسب ساختار و نوع متغیرهای این لیستهای پیوندی بعهد شما است و بخشی از نمره این پروژه است.

دستورهای اجرایی سامانه

این سامانه بانکی به صورت Command line دستورهای کاربر را دریافت می کند. فهرست دستورهای در جدول زیر آمده است و در ادامه به توضیح هر یک از این دستورها می پردازیم.

رد.	دستور	الگوی دستور	مثال
1	تعریف یک کاربر	add_user <username> <password> <phonenum>	add_user person 231 09120000000
3	ورود به سامانه	login <username> <password>	login person 231
3	خروج از سامانه		
4	اضافه کردن موجودی	deposit <username> <amount>	deposit person 1000
5	برداشت موجودی برای یک کاربر	withdraw <username> <amount>	withdraw person 1000
6	مشاهده اطلاعات کاربر توسط مدیر	view <username>	view person
7	حذف یک کاربر	delete <username>	Delete person
8	انتقال موجودی به حساب یک کاربر دیگر	transfer <username> <amount>	transfer person_2 1000
9	مشاهده اطلاعات حساب خود	view	view
10	خروج از برنامه	exit	exit

ثبت نام یک کاربر جدید

تنها کاربر مدیر می تواند یک کاربر را به این سامانه اضافه کند. لذا این دستور نباید توسط کاربران عادی اجرا شود و پیغام متناسب باید به ایشان داده شود.

کاربر مدیر برای اضافه کردن کاربر جدید باید نام کاربری و رمز عبور و شماره تلفن این کاربر را وارد کند. توجه کنید که هم نام کاربری و هم شماره تلفن کاربران باید یکتا بوده و هیچ کاربری با نام کاربری یا شماره تلفن مشابه نباید از قبل در سامانه وجود داشته باشد. اگر چنین کاربری از قبل در سامانه وجود داشته باشد، یک پیغام خطای مناسب باید چاپ شود. بعد از ثبت نام یک شناسه یکتا (id) هم باید به صورت اتوماتیک برای این کاربر جدید در نظر گرفته شود. فرمت این دستور به این شکل است:

`add_user <username> <password> <phonenum>`

Example: `add_user person 231 09120000000`

ورود به حساب کاربری

تمامی کاربرها، چه عادی و چه admin، باید بتوانند با استفاده از دستور login وارد سامانه شوند. در صورت عدم وجود کاربری با اطلاعات وارد شده باید خطای مناسبی چاپ کنید.

فرمت این دستور:

`login <username> <password>`

Example: `login person 231`

در این مثال کاربر با نام کاربری person با آن وارد حساب کاربری خود می شود. برای خروج هم از دستور logout استفاده می شود. توجه کنید که برنامه بعد از اجرای دستور logout باید به صفحه login منتقل شود و نه اینکه برنامه تمام شود زیرا ممکن است کاربر دیگری بخواهد وارد سامانه شود.

اضافه کردن موجودی به حساب کاربران

تنها کاربر مدیر قادر است که مبلغی را به موجودی حساب کاربران سامانه اضافه کند. برای این کار باید نام کاربری و مبلغی که می خواهد به موجودی کاربر مورد نظر بیافزاید را مشخص کند. در صورت عدم وجود کاربری با نام کاربری وارد شده باید یک پیغام خطای مناسب چاپ شود.

فرمت این دستور به این شکل است:

`deposit <username> <amount>`

Example: `deposit person 1000`

که به حساب کاربر با نام کاربری person 1000 واحد واریز می کند.

مشاهده اطلاعات یک کاربر

کاربر مدیر باید بتواند اطلاعات کاربران سامانه را مشاهده کند. این اطلاعات شامل:

- شناسه یا id کاربر
- نام کاربری
- شماره تلفن
- مقدار موجودی حساب

- تمام تراکنش های این کاربر اعم از انتقال موجودی به کاربری دیگر و دریافت موجودی از کاربری دیگر و واریز و برداشت موجودی از آن حساب باشد. در صورت عدم وجود کاربری با نام کاربری وارد شده باید یک پیغام خطای مناسب چاپ شود.
فرمت این دستور:

view <username>

Example: view person

که اطلاعات مذکور کاربر با نام کاربری person را نمایش می دهد. انتخاب نحوه نمایش اطلاعات کاربر برعهده خودتان است و مطابق سلیقه خود آن را نمایش دهید.

برداشت موجودی برای کاربران

تنها کاربر مدیر میتواند از حساب کاربران موجودی برداشت کند و کاربرانی که میخواهند بخشی از موجودی حساب خود را برداشت کنند، باید از طریق کاربر مدیر اقدام نمایند. برای این کار کاربر مدیر باید نام کاربری و مبلغی که میخواهد برداشت کند را وارد کند. توجه کنید که مبلغ مذکور باید از موجودی حساب آن کاربر بیشتر نباشد و در غیر این صورت باید پیغام خطای مناسب چاپ شود. همچنین در صورت عدم وجود کاربری با نام کاربری وارد شده باید پیغام خطای مناسبی چاپ شود. فرمت این دستور به این شکل است:

withdraw <username> <amount>

Example: withdraw person 1000

که از حساب کاربری کاربر با نام کاربری person 1000 واحد برداشت می کند.

توجه: لازم نیست که مکانیزم درخواست به مدیر برای برداشت موجودی توسط کاربران پیاده سازی شود و تنها پیاده سازی امکان برداشت توسط کاربر مدیر کفایت می کند.

حذف یک کاربر

کاربر مدیر میتواند یک کاربر را از سامانه حذف کند. توجه کنید که با این کار تمام اطلاعات این کاربر باید از حافظه و طبیعتاً فایل پاک شوند.
فرمت این دستور:

delete <username>

Example: delete person

که حساب کاربری کاربر با نام کاربری person را پاک می کند.

انتقال موجودی به حساب یک کاربر دیگر

هر کاربر این سامانه می تواند به حساب یک کاربر دیگر موجودی منتقل کند. برای این کار باید نام کاربری کاربر مقصد و مبلغی که میخواهد منتقل کند را وارد نماید. توجه کنید که مبلغ مذکور باید از موجودی حساب کاربر مبدا بیشتر نباشد و در غیر این صورت باید پیغام خطای مناسب چاپ شود. همچنین در صورت عدم وجود کاربر مقصد وارد شده باید پیغام خطای مناسبی چاپ شود.
فرمت این دستور:

transfer <username> <amount>

Example: transfer person_2 1000

که به حساب کاربری کاربر با نام کاربری 1000 person_2 واحد انتقال می دهد.

مشاهده اطلاعات حساب خود

هر کاربر باید بتواند اطلاعات حساب خود را مشاهده کند. اطلاعات نمایش داده شده در این بخش همان اطلاعاتی هستند که در بخش مشابه اختیارات مدیر بیان شد.
فرمت این دستور:

view

که اطلاعات کاربری که login شده است را نمایش می دهد.

خروج از برنامه

برای خروج از برنامه باید از دستور زیر استفاده شود. در هنگام خروج باید کلیه اطلاعات در فایل ذخیره شود.
فرمت این دستور:

exit

راهنمایی

- 1- این پروژه بزرگتر از دو پروژه قبلی است، لذا طراحی بالا به پایین (Top down design) برای شناسایی اجزای کوچکتر این پروژه و شروع از نوشتن آنها بسیار مفید خواهد بود.
- 2- پیشنهاد می شود برای انجام هر یک از دستورهای اجرایی سامانه تابع مستقلی بنویسید.
- 3- دقت کنید که دستورهای مجاز زمانی که کاربری (مدیر و یا عادی) لاگین کرده و یا نکرده باشد فرق می کنند. همچنین فهرست دستورهای مجاز زمانی که مدیر لاگین باشد و یا کاربر عادی با هم فرق می کنند. استفاده از نگاه ماشین حالت می تواند مفید باشد.
- 4- پیشنهاد می شود. تابعی برای تشخیص و بررسی اجزای یک دستور بنویسید.
- 5- برای اضافه و کم کردن نود لیستهای پیوندی از malloc و free استفاده کنید.
- 6- برنامه با خواندن از فایل شروع و با نوشتن به فایل باید خاتمه یابد.
- 7- بلافاصله بعد از این که فایل را خواندید و یا اطلاعات را در آن نوشتید، فایل را ببندید.
- 8- دقت کنید که بعد از توابعی نظیر malloc، fopen، fscanf، scanf، حتما مقدار برگشتی این توابع را بررسی نموده و متناسب با آن اقدام نمایید.

نکات مهم

- برای ذخیره کاربران این سامانه و تبادلات ایشان باید از لیستهای پیوندی (Linked List) استفاده شود. استفاده از آرایه ها قابل قبول نیست.
- دقت کنید که برنامه شما به هیچ وجه memory leak نداشته باشد. طبیعتاً هر کاربری که از سامانه حذف می شود تمام اطلاعات مربوط به او باید از حافظه پاک شوند و حافظه آزاد شود.
- کاربر مدیر و کاربران عادی برای آن که بتوانند کاری در سامانه انجام دهند ابتدا باید با وارد کردن نام کاربری و رمز عبور خود وارد سامانه شوند. در صورت عدم وجود کاربری با نام کاربری و رمز عبور وارد شده باید پیام خطای مناسبی چاپ شود.
- در تمام بخش ها چه برای موفقیت آمیز بودن و چه شکست خوردن باید پیام مناسبی را چاپ کنید. انتخاب محتویات این پیام برعهده خودتان است اما باید مناسب و قابل فهم باشد.

نحوه ارزیابی:

عنوان	نمره
نام گذاری مناسب و اصولی متغیرها	۱۰
استفاده از تمام ورودی های تابع در آن	۵
نام گذاری مناسب توابع	۵
عدم وجود قطعه کد تکراری	۱۰
دریافت صحیح ورودی	۱۰
رسیدگی به خطاها و چاپ پیام های خطای مناسب	۱۰
کامنت گذاری مناسب در هرجایی که نیاز به مستندسازی دارد	۵
نداشتن memory leak	۱۰
استفاده صحیح از ساختار داده Linked List	۱۰
تسلط بر روی کد	۱۰
تست و اجرای برنامه	۱۵

نحوه‌ی تحویل پروژه :

شما باید فایل‌های c و h خود را در یک فایل زیپ با فرمت "zip" و با نام CA3_SID.zip قرار دهید که SID همان شماره‌ی دانشجویی شماست. برای مثال اگر شماره‌ی دانشجویی شما 810198000 باشد باید نام فایل خود را CA3_810198000.zip قرار دهید. مهلت آپلود تا ساعت ۲۳:۵۵ روز سه شنبه ۱۷ دی است. تأخیر در سه روز نخست به ازای هر روز ۱۰ درصد و در دو روز دوم به ازای هر روز ۱۵ درصد جریمه خواهد داشت. پس از این پنج روز، به هیچ وجه نمره‌ای در نظر گرفته نخواهد شد. همچنین توجه کنید که با توجه به تأخیر در نظر گرفته شده به هیچ وجه امکان تمدید پروژه‌ها وجود ندارد.

پروژه‌ها برای یادگیری برنامه نویسی و مباحث مطرح شده در کلاس طراحی میشوند و انجام آنها به صورت انفرادی خواهد بود. همچنین، در صورت شباهت میان دو پروژه (که به وسیله ی نرم افزارهای مربوطه چک میشود) برای هر دو نفر نمره ی صفر در نظر گرفته خواهد شد.

در صورت وجود هرگونه سوال میتوانید پرسشهای خود را در فروم درس (در بخش مربوط به این پروژه) مطرح نمایید یا به

icsp98.ca@gmail.com ایمیل بزنید.