



در فاز های قبلی قابلیت های مختلفی در پیام رسان ایجاد شد، در این فاز میخواهیم چند قابلیت جدید به پیام رسان اضافه کنیم.

این فاز نیز از دو بخش تشکیل شده است.

**بخش ۱:** مانند هر پروژه بزرگی لازم است که ابتدا طراحی اولیه پروژه نوشته شده و اشکالات آن برطرف شود. تنها پس از نهای شدن طراحی می توان نسبت به پیاده سازی کد زبان C پروژه اقدام نمود. بنابراین ابتدا و در بخش اول این پروژه لازم است که مستندات طراحی بالا به پایین<sup>۱</sup> خود را رسم کرده و آن را همراه طراحی ساختار داده<sup>۲</sup> خود تحویل دهید. برای این کار لازم است که صورت پروژه را بطور کامل و با دقت بخوانید.

**بخش ۲:** در بخش دوم کد C پیاده سازی طراحی خود را نوشته و تحویل خواهید داد.

### مباحث درسی که در این فاز به آنها پرداخته می شود:

(۱) استفاده از تمامی مفاهیمی که در فاز های قبل آموخته اید. (مانند قواعد نام گذاری، کامنت گذاری و ...)

(۲) استفاده از طراحی بالا به پایین.

(۳) آشنایی با تخصیص حافظه پویا (Dynamic memory allocation) حافظه heap و توابع مرتبط با آن.

(۴) آشنایی با ساختار داده.

(۵) آشنایی با نحوه کار لیست پیوندی<sup>۳</sup>.

(۶) کار با رشته ها و عملیات های مختلف روی آنها.

<sup>۱</sup> Top-Down Design

<sup>۲</sup> Data Structure

<sup>۳</sup> Linked List

## شرح پروژه

در این فاز باید تغییرات خواسته شده نسبت به فاز های قبلی را اعمال کنید و همچنین دستورات جدیدی را نیز اضافه کنید.

لطفا توجه داشته باشید که قابلیت های پیام رسان که در این مرحله از پروژه در مورد آنها چیزی گفته نشده است عیناً مشابه دو پروژه قبل است.

در فاز های قبل، کاربر دستورهای خود را از طریق یک منوی عددی و با انتخاب عدد مربوط به هر گزینه انتخاب و به برنامه اعلام می کرد. در این فاز سعی داریم روش دیگر اعلام گزینه های انتخابی، که روش خط فرمان (Command Line) است را تجربه کنیم. در این روش فرمان مربوطه و پارامترهای آن در یک خط و پشت سر هم نوشته شده و به برنامه داده می شود. درست مشابه دستورهای که می توانیم در Command Prompt در سیستم عامل Win10 وارد کنیم.

لذا در این فاز به هیچ وجه مجاز به استفاده از منوهای عددی مشابه دو فاز قبلی برای اجرای دستورات پیام رسان نیستید.

## رفع محدودیت های فاز های قبل:

با توجه به امکاناتی که تا اینجا از زبان C آموخته اید، دیگر می توانید محدودیتهای برنامه را رفع نموده و برنامه باید بتواند با تعداد کاربران نامحدود، طول پیام های نامحدود و تعداد پیام های نامحدود کار کند.

## ورود و خروج:

کاربر می تواند با نوشتن دستورات زیر به ترتیب به حساب خود وارد شود و یا از آن خارج شود. توجه کنید که اگر نام کاربری یا رمز عبور اشتباه وارد شده باشد، و یا کاربری پیش از این به حساب کاربری خود وارد شده و هنوز خارج نشده و مجدداً بخواهد از دستور ورود استفاده کند، باید موضوع را با نمایش پیغام خطای مناسب به کاربر اطلاع دهید. همچنین اگر هنوز کاربری وارد نشده باشد، با وارد کردن دستور logout باید پیغام خطای مناسب نمایش داده شود.

**login <username> <pass>**

**logout**

## فرایند ثبت نام / ویرایش اطلاعات:

هنگام ثبت نام کاربر جدید با وارد کردن دستور زیر، نام کاربری، رمز عبور و شماره تلفن را از کاربر گرفته و بعد از این کاربر با نام کاربری خود نمایش داده خواهد شد، همچنین باید تاریخ ثبت نام کاربر جدید نیز در اطلاعات کاربر ثبت شود.

**signup <username> <password> <number>**

همچنین یک کاربر می تواند با دستورات زیر اطلاعات خود را ویرایش کند.

**edit\_username <new username>**

**edit\_pass <new password>**

نام کاربری و شماره تلفن جدید باید در بین کاربرانی که تا کنون ثبت نام کرده اند یکتا و غیر تکراری باشد.

همچنین رمز عبور جدید یک کاربر نباید با رمز عبور قبلی خودش یکسان باشد.

طول رمز عبور باید حداقل ۶ رقم باشد.

شماره تلفن نیز باید با صفر آغاز شده و دقیقا برابر ۱۱ رقم باشد. همچنین نباید شامل کاراکترهای غیر عددی باشد.

در صورت عدم رعایت هر کدام از موارد بالا، پیغام خطای مناسب را به کاربر نمایش دهید.

## نمایش پیام های خوانده نشده و اختصاص شناسه:

در این فاز باید هر پیامی شناسه مخصوص به خود را داشته باشد.

هنگامی که کاربری وارد پیام رسان می شود باید پیام های خوانده نشده آن کاربر بر اساس زمان دریافت (زمان ارسال از مبدا) آنها مرتب شده و فقط اطلاعات پیام ها (شامل فرستنده و تاریخ) نمایش داده شود و کاربر برای مشاهده کامل پیام باید از دستوراتی که در ادامه خواهد آمد استفاده کند.

## ارسال پیام:

برای ارسال پیام از دستور زیر استفاده خواهیم کرد.

**send <username> <content>**

توجه کنید که هر پیام باید در مشخصات خود تاریخ ارسال را داشته باشد.<sup>۱</sup>

---

<sup>1</sup> Hint: Use "time.h" library.

## نمایش اطلاعات کاربران:

هر کاربر میتواند با وارد کردن دستور زیر نام تمام کاربران را مشاهده کند.

**info**

همچنین با استفاده از دستور زیر میتوان اطلاعات کامل تری از یک کاربر شامل نام و شماره موبایل وی را مشاهده کرد.

**about <username>**

## نمایش پیام های دریافتی:

هر کاربر می تواند با وارد کردن دستور زیر پیام هایی را که از یک کاربر دریافت کرده است را مشاهده کند. توجه کنید که جزئیات پیام باید به صورت کامل (فرستنده، تاریخ و ...) نمایش داده شود.

**show\_received <username>**

## نمایش پیام های ارسالی:

هر کاربر می تواند با وارد کردن دستور زیر پیام هایی را که برای یک کاربر ارسال کرده است را مشاهده کند. توجه کنید که جزئیات پیام باید به صورت کامل (فرستنده، گیرنده، تاریخ و ...) نمایش داده شود.

**show\_sent <username>**

## بستن پیام رسان و ذخیره سازی اطلاعات:

با دستور زیر پیام رسان بسته می شود و باید ذخیره سازی انجام شود. دقت کنید که تنها هنگامی عملیات ذخیره سازی انجام می شود که از این دستور استفاده شود و در غیر این صورت نباید چیزی ذخیره شود.

**close**

## ذخیره سازی:

برای اجرای این دستور مانند فاز قبلی پروژه باید اطلاعات لازم ذخیره سازی شود. برای این کار از دو فایل استفاده خواهیم کرد:

- اطلاعات شناسایی: در این فایل فقط اطلاعات کاربران (نام کاربری، شماره موبایل، و ...) نگهداری می شود.

- اطلاعات پیام‌ها: کلیه پیام‌هایی که بین کاربران رد و بدل شده است به همراه اطلاعات هر پیام (فرستنده، گیرنده، زمان ارسال و ....) باید در این فایل ذخیره شود.

این موارد پیشنهادی بوده و شما همچنان می‌توانید به شکل دلخواهی اطلاعات را در فایل‌ها ذخیره کنید.

### بازیابی اطلاعات:

همچنین در هر بار اجرای برنامه باید اطلاعاتی که در فایل‌ها (در اجراهای قبلی) ذخیره سازی شده است از فایل(ها) خوانده شود و در متغیرهای مناسب ریخته شود.

## بخش های امتیازی

### ۱- نمایش کاراکتر (Prompt) دریافت دستور جدید (امتیازی):

هنگامی که برنامه منتظر وارد کردن دستور توسط کاربر است در ابتدای آن خط باید علامت ">>" نشان داده شود.

توجه کنید که این علامت فقط باید در ابتدای خط هایی که در آن منتظر وارد کردن دستور توسط کاربر هستیم نمایش داده شود و در ابتدای خط هایی که پاسخ یک دستور یا پیغامی را به کاربر نمایش می دهید، این علامت نباید وجود داشته باشد (به نمونه مراجعه کنید).

### ۲- نمایش رنگی پیام های ارسالی خوانده نشده (امتیازی):

در صورتی که پیامی هنگام نمایش برای فرستنده (دستور show\_sent) توسط گیرنده خوانده نشده بود باید با رنگ متفاوت (سبز یا قرمز) نمایش داده شود.

### ۳- تشکیل و عضویت در گروه (امتیازی):

هر کاربر می تواند با وارد کردن دستور زیر یک گروه جدید بسازد. هنگامی که گروه جدید ساخته می شود باید به آن یک شناسه یکتا اختصاص داده شود.

#### **create\_group <group\_name>**

کاربر سازنده آن گروه باید به عنوان مدیر (admin) در نظر گرفته شده و اولین عضو گروه خودش باشد.

نام گروه نیز باید در بین نام تمامی گروه های موجود، یکتا و غیر تکراری باشد.

مدیر گروه توانایی اضافه کردن یک کاربر جدید با دستور زیر را دارد.

#### **add\_to <group name> <username>**

۱. در دستور های ارسال، نمایش پیام های دریافتی و نمایش پیامهای ارسالی می توان از نام گروه نیز به جای نام کاربر استفاده کرد.

۲. همچنین در زمان ذخیره سازی اطلاعات کاربران باید نام، نام خانوادگی، گروه هایی که هر کاربر در آنها عضو است نیز ذخیره شوند.

۳. دقت کنید که در زمان ذخیره سازی پیامها، کلید پیام هایی در گروه ها وجود دارد هم باید ذخیره و در زمان باز شدن برنامه باز یابی شوند.

## نحوه تحویل این پروژه در دو بخش:

### بخش اول:

#### مهلت تحویل: ۱ بهمن ۹۹

همانطور که در ابتدای پروژه گفته شد، بر اساس مواردی که باید پیاده سازی شوند یک طراحی بالا به پایین و کدهای مربوط به تعریف ساختارهای داده (ساختار لیست پیوندی و structure ها) تحویل داده شود.

### بخش دوم:

#### مهلت تحویل: ۷ بهمن ۹۹

پس از انجام طراحی و تعریف ساختارهای داده، پیام رسان با تمام ویژگی هایی که در بالا ذکر شده است را پیاده سازی کنید.

## نکات تکمیلی:

- ۱) در این پروژه حافظه ها باید به صورت پویا گرفته شوند (با استفاده از دستورات `malloc` و `realloc`) و نباید در تعداد کاربرها، طول نام کاربرها، تعداد پیام ها، و یا طول دستورها هیچ گونه محدودیت عددی وجود داشته باشد (قرار دادن آرایه با طول بزرگ پذیرفته نیست).
- ۲) در صورتی که از دستور `goto` در برنامه خود استفاده کنید، نمره کل پروژه را از دست خواهید داد.
- ۳) شما برای نگهداری اطلاعات کاربران در طول اجرای برنامه ی خود تنها مجاز به استفاده از لیست های پیوندی هستید و استفاده از آرایه ها قابل قبول نیست.
- ۴) تمامی دستورات و آرگومان های آن ها با استفاده از `space` از یکدیگر جدا می شوند. بنابراین نام گروه ها، کاربران و نیز پیام ها نباید شامل `space` باشد.
- ۵) با اجرای چند باره ی برنامه باید اطلاعات قبلی (تمامی اطلاعات مربوط به کاربران، گروه ها و پیام ها) همچنان وجود داشته باشد، یعنی با بستن برنامه و بازکردن دوباره آن، هیچ اطلاعاتی از دست نرود.
- ۶) قسمت هایی از پروژه که وظیفه خاصی را برعهده دارند می بایست به صورت تابع نوشته شوند (تعداد خطوط هر تابع شما نباید بیشتر از ۲۰ تا ۳۰ خط باشد).
- ۷) جدا کردن فایل های سرآیند<sup>۱</sup> و نوشتن کتابخانه نیز بخشی از نمره شما را تشکیل می دهد.
- ۸) برنامه ی شما باید در محیط `Microsoft Visual Studio 2015` کامپایل و اجرا شده و در صورت وجود خطای کامپایل و یا `Warning`، حداقل ده درصد از نمره پروژه را از دست خواهید داد.
- ۹) در این پروژه باید انواع خطاهای ورودی را چک کنید. تشخیص مصداق های آن برعهده ی شماست. برای مثال نمونه ای از این خطا ها عبارتند از:
  - a) وارد کردن نام تکراری هنگام ثبت نام و یا ایجاد گروه
  - b) اشتباه تایپی در وارد کردن دستورات مانند: `logn mahdi 1234`
  - c) وارد کردن دستوراتی که لازم است قبل از آن یک کاربر وارد شده باشد در حالیکه کسی وارد نشده است (مانند ارسال پیام).
- ۱۰) پیش از آنکه نگران انجام بخش های امتیازی باشید، در مورد بخش اصلی پروژه طراحی درست آن، عملکرد صحیح آن و کامل بودن آن وقت لازم و مکفی بگذارید. داشتن یک کد تمیز و درست بهتر است داشتن کدی ناقص است که بخش های امتیازی را هم داشته باشد.

---

<sup>1</sup> Header Files



نمونه ای از اجرای برنامه در رابط خط فرمان<sup>۱</sup>:

```
*Welcome to UT Messenger
please enter your command
*****
>> signup user1 pass12 09123456789
      successfully registered
>> logout
      successfully logged out
>> signup user2 pass22 09113456789
      successfully registered
>> send user1 salam
      message successfullly sent!
>> create_group friends
      new group created!
>> add_to friends user1
      new member added to group!
>> send friends HI
      message successfully sent!
>> show_sent friends
      in:2021/1/6 17:40
      in group: friends
      HI
>> show_sent user1
      message to:user1
      in:2021/1/6 17:39
      salam
>> logout
      successfully logged out
>> login user1 pass12
      successfully logged in
      Unread messages:
      message from: user2 at 17:39 in: 2021/1/6
      message in: friends at 17:40 in: 2021/1/6
>> show_received user2
      message from:user2
      in:2021/1/6 17:39
      salam
>> show_received friends
      message from:user2
      in:2021/1/6 17:40
      in group: friends
      HI
>> close
```

---

<sup>1</sup> Command Line Interface

## شیوه نمره دهی:

| عنوان  | نمره                         |
|--|------------------------------|
| طراحی بالا به پایین و تقسیم صحیح و مناسب مسئله به مسائل کوچکتر     | ۲۵ (بخش اول)                 |
| طراحی ساختارهای داده   | ۲۵ (بخش اول)                 |
| استفاده مناسب از توابع برای زیر مسئله ها (کوچک بودن توابع)         | ۱۰                           |
| نام گذاری اصولی و مناسب متغیر ها                                   | ۲                            |
| استفاده از تمام ورودی های تابع در آن                               | ۲                            |
| نام گذاری مناسب توابع  | ۲                            |
| عدم وجود قطعه کد تکراری  | ۵                            |
| استفاده صحیح از فایل های سرآیند <sup>۱</sup> برای تقسیم بندی توابع | ۵                            |
| رسیدگی به خطاها  | ۵                            |
| اجرای بخش ثبت نام و ویرایش اطلاعات کاربر                           | ۱۰                           |
| اجرای بخش ورود و خروج کاربر  | ۵                            |
| اجرای بخش دریافت و ارسال پیام                                      | ۱۵                           |
| اختصاص شناسه به هر پیام  | ۴                            |
| نمایش پیام های خوانده نشده بصورت مرتب شده                          | ۶                            |
| استفاده صحیح از لیست پیوندی  | ۲۰                           |
| تقسیم صحیح اطلاعات بین لیست های پیوندی مختلف                       | ۴                            |
| خواندن و نوشتن صحیح در فایل  | ۱۵                           |
| استفاده صحیح از حافظه پویا   | ۲۰                           |
| نمایش رنگی پیام های ارسالی خوانده نشده (امتیازی)                   | ۱۰                           |
| تشکیل و عضویت در گروه (امتیازی)                                    | ۲۰                           |
| نمایش کاراکتر دریافت دستور جدید (امتیازی)                          | ۵                            |
| تسلط و پاسخ دهی متناسب با میزان عملکرد صحیح برنامه                 | ۵۰ + ۱۰۰ (بخش اول + بخش دوم) |
| مجموع  | ۳۷۰                          |

<sup>1</sup> Header Files

## نحوه تحویل پروژه

- (۱) برنامه خود شامل تمام فایل های h و c را در یک فایل zip و با نام CA3-P[1 or 2]-SID.zip در صفحه CECM درس بارگذاری کنید؛ بطوریکه SID شماره دانشجویی شماست. برای مثال اگر شماره دانشجویی شما ۸۱۰۱۹۹۱۲۳ باشد، برای بخش اول و دوم نام پرونده شما باید به ترتیب CA3-P1-810199123.zip و CA3-P2-810199123.zip باشد. دقت کنید که عدم رعایت ساختار گفته شده موجب کسر ۵ درصد از نمره کل هر بخش خواهد شد.
- (۲) هدف از این پروژه یادگیری شماست؛ لطفاً پروژه را خودتان انجام دهید. در صورت کشف تقلب و یا شباهت میان دو کد، برای فرستنده و گیرنده نمره صفر لحاظ خواهد شد.
- (۳) در صورت وجود هرگونه سوال و یا ابهام می توانید پرسشهای خود را در فروم درس مطرح نمایید و یا به [icsp99.ca@gmail.com](mailto:icsp99.ca@gmail.com) ایمیل بزنید.
- (۴) به تمامی مواردی که در فروم درس و یا جلسه توجیهی پروژه مطرح می شوند استناد خواهد شد و شما موظف هستید این موارد را بررسی کرده و آنها را رعایت نمایید.
- (۵) تحویل با تاخیر برای هر کدام از بخش های اول و دوم پروژه حداکثر تا ۲ روز پس از مهلت اعلام شده مجاز بوده و جریمه ی تاخیر برای هر روز ۱۰ درصد می باشد. پس از این زمان به هیچ وجه امکان ارسال پروژه وجود نخواهد داشت.

موفق باشید

سید مهدی حسینی

محمدرضا عظیمی