



معرفی :



امروزه دانشجویان و اساتید با استفاده از سامانه‌های مختلفی می‌توانند با یکدیگر در ارتباط باشند. این سامانه‌ها امکانات بسیاری از قبیل نمره‌دهی، تحویل تکالیف و ... را در اختیار دانشجویان و اساتید قرار می‌دهد تا امر آموزش تسریع یابد. [Moodle](#) یک سامانه در این جهت می‌باشد که با فراهم آوردن امکانات فراوان یکی از بهترین سامانه‌ها در این زمینه می‌باشد. سایت درسی [CECM](#) نیز بر پایه‌ی این سامانه می‌باشد.

در این پروژه شما باید در دو فاز یک نمونه‌ی ساده از این سیستم را پیاده‌سازی کنید که امکاناتی از قبیل ثبت‌نام، ورود، نمره‌دهی، دیدن نمره و .. را دارد. در فاز اول ساختار داده و طراحی بالا به پایین از شما تحویل گرفته می‌شود و در فاز دوم به پیاده‌سازی آن می‌پردازید.

اهداف پروژه :

- آشنایی با ساختار داده‌ی linked list
- آشنایی با کار با فایل (ذخیره‌ی اطلاعات در فایل و خواندن از آن)
- آشنایی بیشتر با پوینترها و حافظه‌ی پویا
- تمرین کدزنی برای پروژه‌های بزرگ‌تر

توضیح پروژه :

سیستم باید از ویژگیهای زیر برخوردار باشد :

ثبت نام :

در ابتدا دانشجوی جدیدی که وارد سیستم میشود باید بتواند در آن ثبت نام کند. این عمل با دستور زیر انجام می شود :

signup_student <username> <password>

همچنین اساتید باید بتوانند با این دستور در سیستم ثبت نام کنند. این عمل با ساختار زیر انجام میشود :

signup_teacher <username> <password>.

- بعد از ثبت نام موفقیت آمیز، کاربر به صورت خودکار و بدون نیاز به وارد کردن دستورات ورود، به سیستم وارد می شود.
- دانشجویان و اساتید می توانند نام کاربری یکسان داشته باشند ولی دو دانشجو و یا دو استاد نمی توانند نام کاربری یکسان داشته باشند.

ورود :

هر کدام از دانشجویان و اساتید می توانند با وارد کردن دستورهای زیر وارد سیستم بشوند :

login_student <username> <password>

login_teacher <username> <password>

خروج :

هر کدام از دانشجویان و اساتید می توانند با وارد کردن دستور زیر از سیستم خارج بشوند :

logout

ورود به درس :

دانشجویان می توانند به یک استاد برای ورود به درس درخواست دهند :

request_course <teacher_username>

همچنین اساتید می توانند لیست دانشجویانی که به ایشان درخواست داده اند را مشاهده کنند :

requests_list

و در نهایت می توانند ورود یک دانشجو به درسشان را با دستور زیر قبول کنند :

accept <student_username>

نمره دهی :

استاد می تواند به دانشجویان نمره دهد :

grade <student_username> <grade_score>

- در صورتی که دانشجو نمره ای داشت، نمره ی جدید جایگزین نمره ی قبلی می شود.
- نمره ها می توانند اعشاری هم باشند.

دانشجو می تواند نمره ی یک درس خاص را مشاهده کند :

show_grade <teacher_username>

- در صورتی که استاد هنوز نمره ای وارد نکرده بود یا دانشجو در آن درس اضافه نشده بود هر کدام با پیغام مناسب نمایش داده شود.

لیست دروس :

دانشجو می تواند لیست تمام اساتیدی که درخواست او را قبول کرده اند را با دستور زیر ببیند:

teachers_list

خروج از سیستم و ذخیره ی اطلاعات :

با استفاده از این دستور کار سیستم با ذخیره ی تمامی اطلاعات در فایل های جداگانه پایان می یابد و از برنامه خارج می شود.

exit

- در هنگام اجرای دوباره ی برنامه باید اطلاعات از فایل ها خوانده شود و ادامه ی کار با استفاده از آن اطلاعات از سر گرفته شود.
- شیوه ی ذخیره سازی در فایل ها برعهده ی خودتان است و هر روش قابل قبولی پذیرفتنی است اما بهتر است که یک فایل برای ذخیره ی اطلاعات مربوط به دانشجویان و یک فایل برای ذخیره ی اطلاعات مربوط به اساتید داشته باشید.

لیست نمرات (امتیازی) :

دانشجوی می تواند لیست تمامی نمرات وارد شده برای او به همراه میانگین آنها را با دستور زیر مشاهده کند :

show_grades

- در این لیست ابتدا اسم هر استاد به همراه نمره ی آن و در آخر میانگین تمام نمرات دانشجو نمایش دهید.

برای مثال در بخش زیر استادی با نام کاربری mohammadi و دانشجویی با نام کاربری farzad در سامانه ثبت نام کرده اند. سپس farzad برای ورود به درس mohammadi به او درخواست داده است. mohammadi درخواست farzad را قبول می کند و به او نمره ی ۱۷.۷۵ را می دهد. در نهایت farzad با ورود به سامانه نمره ی خود را مشاهده می کند.

```
signup_teacher mohammadi 1234
user mohammadi added to teachers
logout
you logged out successfully
signup_student farzad 1234
user farzad added to students
request_course mohammadi
you added to this course
logout
you logged out successfully
login_teacher mohammadi 1234
welcome back mohammadi
requeusts_list
      Reqeusts list
-----
Student Username :      reza
-----
accept farzad
student accepted
grade farzad 17.75
farzad graded
logout
you logged out successfully
login_student farzad
welcome back farzad
show_grade mohammadi
your grade is 17.75
logout
you logged out successfully
```

شیوهی پیاده‌سازی :

- در این پروژه حافظه‌ها باید به صورت پویا گرفته شوند (با استفاده از دستور `malloc` و `realloc`) و نباید در تعداد کاربرها، طول نام کاربرها، تعداد دروس یک دانشجو و طول دستورها هیچ‌گونه محدودیت عددی وجود داشته باشد. (قرار دادن آرایه با طول بزرگ قابل قبول نیست)
- شما برای ذخیره‌سازی اطلاعات کاربران در برنامه‌تان تنها مجاز به استفاده از ساختار داده‌ی `linked list` هستید و استفاده از آرایه‌ها قابل قبول نیست.
- می‌توانید به این صورت عمل کنید که تمام دانشجویان یک لینک لیست و تمام اساتید نیز یک لینک لیست مربوط به خود را داشته باشند. و در نهایت `struct` مربوط به گرهی هر استاد یک لینک لیست از دانشجویانش داشته باشد.
- با اجرای چندباره‌ی برنامه باید اطلاعات قبلی مثل کاربرها و نمره‌ها وجود داشته باشد. یعنی با بستن سیستم و دوباره بازکردن برنامه برای مثال کاربر نیاز به ثبت نام نداشته باشد..
- در این پروژه باید قسمت‌هایی از پروژه که وظیفه‌ی خاصی را بر عهده دارند به صورت تابع نوشته شوند. و جدا کردن فایل‌ها و نوشتن کتابخانه نیز نمره دارد.

نکات مربوطه به پروژه :

- شما در این پروژه موظف هستید که انواع خطاهای ورودی را چک کنید. تشخیص مصداق‌های آن به عهده‌ی شماست. برای مثال دو نمونه از این خطاها عبارتند از :
 ۱. اشتباه تایپی در وارد کردن دستور مانند : `signap_student farzad 1234`
 ۲. اشتباهات معنایی مانند این که تا زمانی که کاربری وارد نشده‌است بتوان نمره‌ای مشاهده کرد و یا این که تا زمانی که کاربری خارج نشده است کاربر دیگری بتواند وارد سامانه شود..

نحوه‌ی تحویل پروژه :

فاز اول :

در فاز اول شما باید طراحی بالاپایین خود را به همراه ساختار داده‌تان را که شامل شیوه‌ی پیاده‌سازی استراک‌های ذخیره اطلاعات و ارتباط آن‌ها با یکدیگر می‌باشد را در یک فایل `pdf` در قسمت در نظر گرفته شده با عنوان `CA4-Phase1-[UploadHere]` در صفحه‌ی درس در سایت `CECM` آپلود نمایید. مهلت آپلود این فاز تا ساعت ۲۳:۵۵ یکشنبه ۲۵ آذر ماه است و هیچ‌گونه تاخیری برای ارسال آن قابل قبول نیست.

فاز دوم :

در فاز دوم شما باید فایل‌های `h` ، `c` خود را در قسمت در نظر گرفته شده با عنوان `CA4-Phase2-[UploadHere]` در صفحه‌ی درس در سایت `CECM` آپلود نمایید. مهلت آپلود تا ساعت ۲۳:۵۵ یکشنبه ۹ دی ماه است . تأخیر در سه روز نخست به ازای هر روز ۱۰ درصد و در سه روز دوم به ازای هر روز ۱۵ درصد جریمه خواهد داشت. پس از این شش روز، به هیچ وجه نمره‌ای

در نظر گرفته نخواهد شد. همچنین توجه کنید که با توجه به تاخیر در نظر گرفته شده به هیچ وجه امکان تمدید پروژه‌ها وجود ندارد.

در هر دو فاز فایل‌های خود را در یک فایل زیپ با فرمت “zip” و با نام CA4-Phase1/2-SID.zip قرار دهید که SID همان شماره‌ی دانشجویی شماست. برای مثال اگر شماره‌ی دانشجویی شما ۸۱۰۱۹۷۰۰۰ باشد باید نام فایل خود را برای فاز اول CA4-Phase1-810197000.zip و برای فاز دوم قرار CA4-Phase2-810197000.zip دهید .

پروژه‌ها برای یادگیری برنامه‌نویسی 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 وسیله‌ی نرم افزارهای مربوطه چک میشود) برای هر دو نفر نمره‌ی 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 icsp97.ca@gmail.com ایمیل بزنید.

نحوه‌ی نمره‌دهی :

| عنوان | نمره |
|-----------------------------------------------------|-----------------------|
| طراحی بالا به پایین | ۲۰ (مربوط به فاز اول) |
| طراحی ساختار داده | ۲۰ (مربوط به فاز اول) |
| تقسیم درست مسئله به مسائل کوچکتر | ۱۰ (مربوط به فاز اول) |
| استفاده از توابع برای زیرمسئله‌ها | ۱۰ |
| نام‌گذاری مناسب و اصولی متغیرها | ۵ |
| استفاده از تمام ورودی‌های تابع در آن | ۵ |
| نام‌گذاری مناسب توابع | ۵ |
| عدم وجود قطعه کد تکراری | ۵ |
| استفاده‌ی درست از هدر فایل‌ها برای تقسیم بندی توابع | ۱۰ |
| رسیدگی به خطاها | ۱۰ |
| تست و اجرای بخش ثبت‌نام | ۱۵ |
| تست و اجرای بخش ورود و خروج | ۲۰ |
| تست و اجرای بخش ورود به درس | ۲۰ |
| تست و اجرای بخش نمره‌دهی | ۱۰ |
| تست و اجرای بخش لیست دروس | ۵ |
| تست و اجرای بخش خروج از سیستم و ذخیره اطلاعات | ۱۵ |
| استفاده از لینک‌لیست | ۲۵ |
| تقسیم صحیح اطلاعات بین لینک‌لیست‌های مختلف | ۵ |

| | |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ۱۵ | درست خواندن از فایل و درست نوشتن در آن (شامل از بین رفتن اطلاعات در این عملیات و استفاده از فایل های جداگانه برای ذخیره ی اطلاعات نامربوط به یکدیگر می باشد) |
| ۲۰ | استفاده از حافظه ی پویا |
| ۱۰ (امتیازی) | پیاده سازی بخش لیست نمرات |

توجه داشته باشید که نمره ی این پروژه از ۲۵۰ محاسبه می شود که ۵۰ نمره مربوط به فاز اول می باشد. همینطور ۱۰ نمره ی امتیاز برای این پروژه در نظر گرفته شده است.