



به نام خدا  
دانشکده‌ی مهندسی برق و کامپیوتر دانشکده فنی دانشگاه تهران  
مبانی کامپیوتر و برنامه‌نویسی



استاد: دکتر مرادی، دکتر هاشمی

پروژه‌ی 3 (UT-Board)

نیمسال اول 02-03

این پروژه دارای بخش امتیازی می‌باشد. همچنین این پروژه دارای بخش تحویل الگوریتم نیز می‌باشد. هدف این پروژه، استفاده از آرایه پویا، اشاره گر ها، struct، فایل، لینک لیست و نحوه کار با آن ها می‌باشد.

استاد دانشکده برق و کامپیوتر بدنبال بستری هستند تا بتوانند کارهای روزمره خود را مدیریت کنند و طبق اولویت هر کدام، آن ها را انجام دهند. از آنجایی که دسترسی به اینترنت در هر جایی از دانشکده امکان پذیر نیست، آن ها از دانشجویان درس مبانی می‌خواهند تا برنامه ای بنویسند تا بدون نیاز به اینترنت، بتوانند از آن استفاده کنند.

## UT-Board

این برنامه، یک برنامه مدیریتی است که به کاربر اجازه می‌دهد با اضافه کردن تسک های مختلف، بتواند کارهای روزمره خود را رصد کنند و آن ها را مدیریت کند. این برنامه قابلیت های مختلفی دارد که در ادامه با آن ها آشنا می‌شویم.

کاربران برای استفاده از این برنامه باید ابتدا در آن ثبت نام کرده و یک نام کاربری و رمز عبور برای خود ایجاد کنند. بعد از آن با استفاده از این نام کاربری و رمز عبور، می‌توانند وارد برنامه شوند.

باید توجه شود که برنامه به کوچک و بزرگ بودن نام کاربری و رمز عبور حساس است.

برای ثبت نام و ورود به برنامه دستور باید به صورت زیر باشد:

```
signup <username> <password>
```

```
login <username> <password>
```

پس از وارد شدن، کاربران می‌توانند با دستورهایی که برای آن ها تعریف شده، از حساب خود استفاده کنند.

## منو برنامه

پس از ورود به برنامه، یک منو همانند شکل زیر به کاربر نشان داده می‌شود:

1. View Boards
2. Create New Board
3. Delete Board

پس از آن کاربر با وارد کردن عدد متناظر با هر یک از گزینه ها، به بخش مربوطه هدایت می‌شود.

## View Boards

کاربر میتواند Board های متفاوتی برای خود بسازد که متناسب با کارهای روزمره اوست. برای مثال یک Board برای کارهای شخصی، یک Board برای کارهای دانشگاه و ...

هنگامی که کاربر بخواهد Board های خود را مشاهده کند، باید لیست کامل آن ها همانند شکل زیر به او نمایش داده شود.

- Available Boards:
1. Work
  2. Personal
  3. Study

پس از انتخاب Board موردنظر، یک منو برای مدیریت لیست های آن باید برای کاربر همانند شکل زیر به نمایش درآید.

```
Board: Work
1. View Lists
2. Create New List
3. Delete List
4. Go Back
```

هر یک از گزینه های موجود در منوی مدیریت لیست، همانند شکل های زیر عمل می کنند.

```
Lists in 'Work' Board:
1. To Do
2. In Progress
3. Completed
```

*View Lists*

```
Add new list to 'Work':
What's the name of the new list?
```

*Create New List*

```
Delete list from 'Work':
Which list do you want to delete?
1. To Do
2. In Progress
3. Completed
```

*Delete List*

پس از اضافه کردن یا حذف یک لیست، در صورت نیاز می توانید از کاربر تاییدیه دوباره دریافت کنید. همچنین پس از انجام این دو عملیات، برنامه باید به منوی Board باز گردد.

پس از نمایش لیست های موجود در یک Board (View Lists)، برای آن یک منو همانند شکل زیر به نمایش در می آید.

```
List: To Do

Tasks:
1. Task: Task 1, Priority: High, Deadline: year-month-day
2. Task: Task 2, Priority: Medium, Deadline: year-month-day
3. Task: Task 3, Priority: Low, Deadline: year-month-day

Options:
1. Add Task
2. Edit Task
3. Delete Task
4. Move Task to Another List
5. Go Back
```

هر یک از گزینه های این منو به صورت زیر عمل می کنند:

```
Add New Task:

Enter Task Name:
Enter Priority (Low/Medium/High):
Enter Deadline (YYYY-MM-DD):
```

*Add Task*

```
Edit Task:

Edit Task Name:
Edit Priority (Low/Medium/High):
Edit Deadline (YYYY-MM-DD):
```

*Edit Task*

```
Delete Task:
Which task do you want to delete?
1. Task 1
2. Task 2
3. Task 3
```

Delete Task

```
Move task to another list:
Which task do you want to move?
1. Task 1
2. Task 2
3. Task 3

Which list would you like to move the task to?
1. In Progress
2. Completed
```

Move Task to Another List

## Create New Board

کاربر می‌تواند Board های مختلفی ایجاد کند. با انتخاب این گزینه تنها لازم است تا نام Board وارد شود. پس از آن، کاربر باید به منوی اصلی بازگشته و در صورت لزوم، به Board ایجاد شده رفته و لیست های موردنیاز خود را در آن قرار دهد.

## Delete Board

کاربر می‌تواند یک Board را به طور کامل حذف کند. تنها نیاز است که نام Board های موجود در این مرحله به کاربر نمایش داده شود و انتخاب کند که کدام Board را می‌خواهد حذف نماید.

## File

تمامی اطلاعات برنامه باید به صورت فایل CSV پس از هر تغییر ذخیره شود. بطوریکه پس از اجرای دوباره برنامه، اطلاعات باید از فایل ها خوانده شده و اطلاعات کاربر پس از بستن برنامه نباید از بین برود. نحوه ذخیره سازی و تعداد فایل های ذخیره شده، بر عهده خودتان می‌باشد.

## امتیازی

### بخش اول (Sort)

کاربر باید بتواند هر یک از لیست های یک Board را برحسب Priority یا Deadline آن مرتب کند و لیست جدید مرتب شده به نمایش در آید. بعد از مرتب کردن یک لیست، در صورتی که دوباره وارد آن شود، باید مرتب شده باقی بماند و نیازی به برگشتن به حالت اصلی نیست. برای مرتب کردن براساس Priority، باید از بیشترین اهمیت به کمترین اهمیت مرتب شود. برای مرتب کردن براساس Deadline، باید براساس نزدیک ترین تاریخ تا دورترین تاریخ (نسبت به یکدیگر) مرتب شود. نحوه ورودی گرفتن برای مرتب کردن هر لیست، بر عهده خودتان می‌باشد.

List: To Do

Tasks:

1. Task: Task 1, Priority: Low, Deadline: 2023-01-01
2. Task: Task 2, Priority: Medium, Deadline: 2023-05-01
3. Task: Task 3, Priority: Low, Deadline: 2023-02-01
4. Task: Task 4, Priority: High, Deadline: 2023-02-26

لیست مرتب نشده

List: To Do

Tasks:

1. Task: Task 4, Priority: High, Deadline: 2023-02-26
2. Task: Task 2, Priority: Medium, Deadline: 2023-05-01
3. Task: Task 3, Priority: Low, Deadline: 2023-02-01
4. Task: Task 1, Priority: Low, Deadline: 2023-01-01

لیست مرتب شده بر اساس Priority

List: To Do

Tasks:

1. Task: Task 1, Priority: Low, Deadline: 2023-01-01
2. Task: Task 3, Priority: Low, Deadline: 2023-02-01
3. Task: Task 4, Priority: High, Deadline: 2023-02-26
4. Task: Task 2, Priority: Medium, Deadline: 2023-05-01

لیست مرتب شده بر اساس Deadline

در صورتی که از یک Priority چند تسک وجود داشت، ترتیب نمایش دادن آن ها اهمیتی ندارد.

## بخش دوم (Reminder)

کاربر پس از ورود به برنامه، در همان منوی ابتدایی، باید بتواند تسک هایی که موعد آن ها تا 3 روز دیگر است را به همراه آن Board مشاهده کند.

قبل از ورود به برنامه، باید تاریخ روز به آن داده شود تا برنامه برای یادآوری، یک مبدأ زمانی داشته باشد.

در صورتی که تاریخ روز برابر با 2023-01-01 باشد، منوی برنامه باید همانند زیر باشد:

Reminder:

1. Task: Task 1, Priority: Low, Deadline: 2023-01-01, Board: Work
2. Task: Task 5, Priority: High, Deadline: 2023-01-02, Board: Personal
3. Task: Task 3, Priority: High, Deadline: 2023-01-04, Board: Study

1. View Boards
2. Create New Board
3. Delete Board

## چند نکته...

- نحوه ورودی گرفتن از کاربر برای پیمایش، بر عهده خودتان است.
- نمایش پیام های اضافی در صورتی که منطقی و معقول باشد ایرادی ندارد.
- لزومی ندارد که اسامی و متن های داخل برنامه و چیدمان صفحه، مطابق صورت پروژه باشد. در این پروژه می توانید از خلاقیت خود استفاده کنید.
- پیاده سازی این برنامه باید به وسیله آرایه پویا صورت گیرد و در صورت مورد نیاز نبودن بخشی از حافظه (مانند حذف کردن یک لیست)، آن قسمت از حافظه باید free شود و memory leak رخ ندهد.
- کاربر می تواند به هر تعداد که نیاز دارد، Task، Board و ... به برنامه اضافه کند و محدودیتی در این زمینه وجود نداشته باشد.

## چند نکته دیگر ...

- استفاده از هر گونه توابع C++ ممنوع می باشد و نمره ای به آن تعلق نمی گیرد.
- پروژه باید نسبت به ورودی های نامعتبر مقاوم بوده و در صورت وارد کردن ورودی نامعتبر، عبارت مناسبی را نمایش دهد. توجه شود که در صورت وارد کردن ورودی نامعتبر، برنامه نباید به اتمام برسد و پس از نمایش عبارت مناسب، باید دوباره ورودی بگیرد و برنامه پیش برود.
- بهتر است هر کدام از مراحل مختلف برنامه را در تابع های جداگانه بنویسید و از آن ها استفاده کنید.

## شیوهی نمره دهی

عنوان	نمره
نامگذاری مناسب و اصولی متغیرها	5
استفاده از تمام ورودی های تابع در آن و ناگذاری مناسب توابع	5
عدم وجود قطعه کد تکراری	5
نمایش درست دستورات در command line	5
دریافت صحیح ورودی در command line	5
رسیدگی به خطاها	10
نمایش پیام های مناسب	5
استفاده از آرایه پویا	10
Free کردن و نداشتن memory leak	10
استفاده صحیح از struct	10
کامنت گذاری مناسب در هر جایی که نیاز به مستندسازی دارد	5
استفاده از header file	5
فلوچارت الگوریتم روند پروژه	5
تست و اجرای برنامه	15
امتیازی بخش اول	10
امتیازی بخش دوم	10
مجموع	120

## تحويل

1. تنها فایل‌های با فرمت ".c" و ".h" را در یک فایل زیپ با فرمت ".zip" و با نام CA3-SID.zip قرار دهید که SID همان شماره دانشجویی شماست. برای مثال اگر شماره دانشجویی شما 810102123 باشد، باید نام فایل خود را CA3-810102123.zip قرار دهید و آن را در قسمت در نظر گرفته شده در صفحه درس در سامانه ایلرن آپلود نمایید.
2. برنامه‌های شما باید با زبان برنامه نویسی C نوشته شود و استفاده از دیگر زبان‌های برنامه نویسی مجاز نیست.
3. لازم است تا برای این پروژه، فلوچارت الگوریتم روند پروژه نیز طراحی شده و آپلود گردد. مهلت آپلود فلوچارت تا روز **جمعه 15 دی ماه** می‌باشد.
4. همچنین مهلت آپلود بخش کد پروژه، تا روز **دوشنبه 25 دی ماه** می‌باشد. لازم به ذکر است شما می‌توانید از تاخیر شناور (grace) باقی مانده خود تنها برای بخش کد استفاده نمایید.
5. این پروژه دارای **تحويل حضوری می‌باشد**. لازم است تا در زمان تحويل، بر پروژه و کد خود تسلط کامل داشته باشید. 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 وسیله نرم افزارهای مربوطه چک می‌شود) برای هر دو نفر نمره‌ی صفر در نظر گرفته خواهد شد.
6. در صورت وجود هرگونه سوال می‌توانید پرسش‌های خود را در فروم درس (در بخش مربوط به این پروژه) مطرح نمایید و یا از طریق ایمیل یا تلگرام با [سپیل](#) در ارتباط باشید.

موفق و سربلند باشید.