



درس برنامه سازی پیشرفته

تمرین دوم

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال اول ۹۹-۰۰

استاد:

وحید سلمانی

مبحث:

برنامه نویسی شی گرا

مهلت ارسال:

۹ آبان

ساعت ۲۳:۵۹:۵۹

ویراستار فنی:

محمد مهدی ابوترابی و پارسا محمدیان



به موارد زیر توجه کنید:

- * به ازای هر سوال در سامانه ی کوئرا، یک بخش جداگانه برای بارگذاری برنامه ی شما وجود دارد. برنامه ی خود با پسوند java. را در بخش مربوط به هر سوال بارگذاری کنید. در واقع کلاس هایتان را در یک فایل بنویسید و تنها آن فایل را آپلود کنید. و کلاس هایتان را در فایل های جدا ننویسید.
- * قسمتی که برای سوال ۴ وجود دارد امتیازی است و ۱۰ درصد نمره ی امتیازی برای هر تمرین است. شما باید یک فایل zip آپلود کنید که در آن کد سوالاتی که زدید موجود باشد ولی این بار باید کلاس های متفاوت در فایل های متفاوت باشند و پکیج بندی درست و clean code را رعایت کرده باشید که در این صورت در تحویل حضوری این zip بررسی میشود و در صورت درست بودن نمره امتیازی می گیرید.
- * ورودی و خروجی شما باید عیناً شبیه به نمونه های ورودی و خروجی باشد؛ لذا عبارت هایی همچون "Enter your number" را قبل از گرفتن ورودی نباید چاپ کنید.
- * پس از ارسال فایل مربوط به هر سوال، سامانه ی کوئرا به صورت لحظه ای برنامه ی شما را داوری کرده و نمره ی آن سوال را به شما اعلام می کند که در صورت کم بودن نمره تان، می توانید آن را تصحیح کرده و دوباره ارسال کنید.
- * هم فکری و هم کاری در پاسخ به تمرینات اشکالی ندارد و حتی توصیه نیز می شود؛ ولی پاسخ ارسالی شما باید حتماً توسط خود شما نوشته شده باشد. در صورت هم فکری در مورد یک سوال، نام فرد دیگر را به صورت کامنت در ابتدای کد هر سوال بنویسید.
- * شما می توانید تمامی سوالات و ابهامات خود را در سایت کوئرا در بخش مشخص شده برای این تمرین بپرسید.
- * مهلت ارسال تمرین تا ساعت ۲۳:۵۹:۵۹ روز ۹ آبان ۱۳۹۹ است.
- * به ازای هر روز تاخیر در ارسال پاسخ هر سوال، ۳۰ درصد از نمره ی کسب شده ی شما در آن سوال کم می شود. به عنوان مثال اگر پاسخ یک سوال را با دو روز تاخیر ارسال کنید، فقط ۴۰ درصد از نمره ی که برای آن سوال گرفته اید برای شما لحاظ خواهد شد.



* از آن جا که هدف این تمرین ، آشنایی با مفاهیم شی گرایی است، رعایت این موارد همانند رعایت کلین کد اجباری است.

* همچنین در سوال اول نیز شما ملزم به رعایت UML داده شده در طراحی خود هستید.

* توجه کنید که به ازای هر روز ارسال زودتر هر سوال به شرط کامل بودن و رعایت نکات تمیزی کد و نکات شی گرایی و طراحی براساس UML در سوال یک ، ۵% نمره اضافه به شما تعلق می گیرد. سقف تعداد روز هایی که برای این موضوع محاسبه می شود ، ۴ است. یعنی در صورت چهار روز ارسال زودتر ۲۰% نمره اضافه به شما تعلق می گیرد.



۱ سامانه‌ی دانشگاه

در این سوال شما باید بر اساس UML داده شده سامانه آموزش برای دانشگاه طراحی کنید! دقت شود که تمامی کلاسها باید مطابق با نمودار UML داده شده پیاده سازی شود و در غیر این صورت نمره ای به شما داده نمی شود! حال به بیان یک سری از ویژگی های سامانه ی آموزش می پردازیم:

اضافه کردن دانشجو با شناسه ی sID به سامانه:

```
addStudent <sID>
```

استاد با شناسه ی ltID به سامانه اضافه می شود. این استاد دروس cID1 و cID2 و ... را ارائه می دهد. همچنین ممکن است یک استاد هیچ درسی ارائه نکند یا فقط یک درس ارائه کند:

```
addLecturer <ltID> <cID1> <cID2> ...
```

دانشجوی با شماره ی sID درس cID را حذف می کند:

```
W <cID> <sID>
```

دانشجوی با شماره ی sID در درس cID ثبت نام می کند:

```
<sID> register <cID>
```

دانشجوی با شماره ی sID در دروسهای cID1 و cID2 و ... ثبت نام می کند:

```
<sID> register <cID1> <cID2> ...
```

استاد با شناسه ی ltID ظرفیت درس cID را به اندازه ی n افزایش می دهد:

```
<ltID> capacitate <cID> <n>
```



استاد با شناسه ی ltID که درس cID را ارائه می دهد، برای دانشجوی sID1 نمره ی mark1 ، برای دانشجوی sID2 نمره ی mark2 و ... را ثبت می کند. تضمین میشود نمره ی تمام دانشجویان درس cID وارد می شود:

```
<ltID> mark <cID> <sID1> <mark1> <sID2> <mark2> ...
```

استاد با شناسه ی ltID برای دانشجویان درس cID نمره ی mark را رد می کند:

```
<ltID> mark <cID> <mark> -all
```

شروع ترم جاری:

```
start semester
```

پایان ترم جاری:

```
end semester
```

پایان زمان ثبت نام:

```
end registration
```

درس با شناسه ی cID که unit واحد دارد به سامانه اضافه می شود:

```
addCourse <cID> <unit>
```



دستورات زیر دستورهای نمایشی هستند:

برای درس cID ویژگی خواسته شده نمایش داده میشود. (اگر students بود لیست تمامی دانشجویان ثبت نامی به ترتیب شماره ی دانشجویی، اگر lecturer بود، شناسه ی استاد ارائه دهنده ی درس، اگر capacity بود، ظرفیت درس و اگر average بود، معدل درس را چاپ کند). در صورت معتبر نبودن شماره ی درس یا حذف شدن آن (به حد نصاب نرسیدن)، پیغام you are not a student را چاپ کنید:

```
showCourse <cID> <students|lecturer|capacity|average>
```

شناسه ی سه دانشجویی که بالاترین نمرات درس را کسب کرده اند چاپ می کند. (در صورت برابر بودن نمره ی چند دانشجو، سه نفر اول به ترتیب شماره ی دانشجویی چاپ شوند):

```
showRanks <cID>
```

معدل یک دانشجو را چاپ می کند. در صورت وجود نداشتن دانشجویی با شناسه ی sID پیام you are not a student را چاپ کنید:

```
showAverage <sID>
```

نمایش دانشجویان تا رتبه ی n از نظر معدل. (در صورت برابر بودن، به ترتیب شماره ی دانشجویی چاپ شوند). (اگر n بیشتر از دانشجویان کلاس بود، پیغام invalid number می دهد):

```
showTopRanks <n>
```

معدل همه دانشجویان را به ترتیب نمایش می دهد (در صورت برابر بودن، به ترتیب شماره ی دانشجویی چاپ شوند):

```
showRanks -all
```



نکات پایانی:

- هر درس به طور پیش فرض دارای ۱۵ نفر ظرفیت می باشد و باید دارای حداقل ۳ نفر برای تشکیل باشد.
- هر دانشجو باید حداقل ۱۲ واحد در یک ترم داشته باشد.
- هر درس تنها توسط یک استاد ارائه می شود، اگر addLecturer برای دو استاد با درس یکسان وارد شود پیغام خطای <cID> already taken را چاپ کند.
- هر دانشجو تنها ۳ واحد را می تواند W کند به این صورت که تا زمانی که واحدهایی که درخواست W آنها را داده از ۳ واحد بیشتر نیستند، آنها حذف می شوند و سپس دیگر حذف نمی شوند. مثلاً اگر دستور حذف یک درس ۴ واحدی را وارد کند، نه تنها درس حذف نمی شود بلکه هیچ واحد دیگری نیز نمی تواند حذف کند.
- شناسه ی اساتید و دانشجویان ۵ رقمی هستند.
- با شروع ترم، ثبت نام نیز شروع می شود.
- پس از پایان زمان ثبت نام، ابتدا تمام دروسی که به حد نصاب نرسیده اند حذف شده و دانشجویان دیگر آن درس ها را نخواهند داشت. سپس تعداد واحد تمام دانشجویان چک می شود. پس از این مرحله دیگر درسها چک نشده و ترم رسماً آغاز می گردد.
- در صورتی که یک دانشجو از کف مجاز واحد کمتر داشته باشد، ترم وی حذف می شود.
- دستورات اضافه شدن دانشجو و استاد قبل شروع ترم وارد می شوند.
- پس از اتمام ترم دستورات show وارد می شوند و برنامه با دستور endShow پایان می پذیرد.
- در صورت وارد شدن دستورات بی ربط چیزی چاپ نمی شود.
- تمامی محاسبات اعشاری با دقت یک رقم اعشار محاسبه شود.

Course	Student
<ul style="list-style-type: none"> - <u>courses : ArrayList<Course></u> - <u>courseID : int</u> - <u>unit : int</u> - <u>capacity : int</u> - <u>numberOfRegisteredStudents : int</u> - <u>score : double</u> 	<ul style="list-style-type: none"> - <u>students : ArrayList<Student></u> - <u>studentID : int</u> - <u>studentCourses : ArrayList<Course></u> - <u>average : double</u>
<ul style="list-style-type: none"> + <u>addNewCourse(course : Course) : void</u> + <u>getCourses() : ArrayList<Course></u> + <u>getNumberOfRegisteredStudents() : int</u> + <u>setNumberOfRegisteredStudents(number : int) : void</u> + <u>incrementNumberOfRegisteredStudents() : void</u> + <u>decrementNumberOfRegisteredStudents() : void</u> + <u>getCourseID() : int</u> + <u>addCapacity(number : int) : void</u> + <u>getScore() : double</u> + <u>getUnit() : int</u> + <u>setScore(score : double) : void</u> + <u>getCapacity() : int</u> + <u>showCourse(courseID : int, string : String) : void</u> 	<ul style="list-style-type: none"> + <u>getAverage() : double</u> + <u>addNewStudent(student : Student) : void</u> + <u>registerCourse(studentID : int, courseID : int) : void</u> + <u>addCourse(course : Course) : void</u> + <u>deleteCourse(studentID : int, courseID : int) : void</u> + <u>setMark(courseID : int, mark : double) : void</u> + <u>setAverages() : void</u> + <u>getStudents() : ArrayList<Student></u> + <u>getStudentID() : int</u> + <u>getStudentCourses() : ArrayList<Course></u> + <u>deleteStudent(studentID : int) : void</u> + <u>showAverage(studentID : int) : void</u> + <u>showRanks() : void</u> + <u>showRanks(courseID : int) : void</u>

Lecturer
<ul style="list-style-type: none"> - <u>lecturers : ArrayList<Lecturer></u> - <u>lecturerID : int</u> - <u>courses : ArrayList<Course></u>
<ul style="list-style-type: none"> + <u>getLecturers() : ArrayList<Lecturer></u> + <u>addNewLecturer(lecturer : Lecturer) : void</u> + <u>addCourse(courseID : int) : void</u> + <u>getCourses() : ArrayList<Course></u> + <u>getLecturerID() : int</u> + <u>addCapacity(lecturerID : int, courseID : int, number : int) : void</u> + <u>setMark(lecturerID : int, courseID : int, mark : double, studentID : int) : void</u>



ورودی نمونه

```
addCourse 40101 3
addCourse 40102 4
addCourse 40103 4
addCourse 40104 3
addCourse 25001 4
addCourse 25002 3
addCourse 25003 4
addCourse 65001 3
addCourse 65002 3
addCourse 65003 4
addStudent 95101
addStudent 95102
addStudent 95103
addStudent 95103
addStudent 95104
addStudent 94101
addStudent 94102
addStudent 94103
addLecturer 10001 40101 40103
addLecturer 10002 40102 40104
addLecturer 10003 25001 25002
addLecturer 10004 65001 65002 65003
start semester
95101 register 40101 40102 40103 25001 25002
95102 register 40101 40102 40103 25001 25002
95103 register 40101 40102 40103 25001 25002
95104 register 40101 40102 40103 25002 65002
94101 register 40102 40103 25001 25002 65001
94102 register 40102 40103 25001 25002 65001
94103 register 40102 40103 25001 65001
end registration
10001 mark 40101 95101 12 95102 18 95103 18.5 95104 9.9
10002 mark 40102 95101 19.8 95102 20 95103 16.5 95104 19 94101 17 94102
10 94102 15.9
10001 mark 40103 95101 17.4 95102 19 95103 15.7 95104 11.2 94101 19 94102
```



```
17.8 94103 20
10003 mark 25001 95101 9 95102 10 95103 10 94101 10 94102
19 94103 12
10003 mark 25002 95101 20 95102 14 95103 18 95104 9.5 94101 20 94102 18
94103 19.6
10004 mark 65001 94101 18 94102 10 94103 15
W 40101 95101
end semester
showAverage 95103
showCourse 25001 average
showRanks 65001
endShow
```

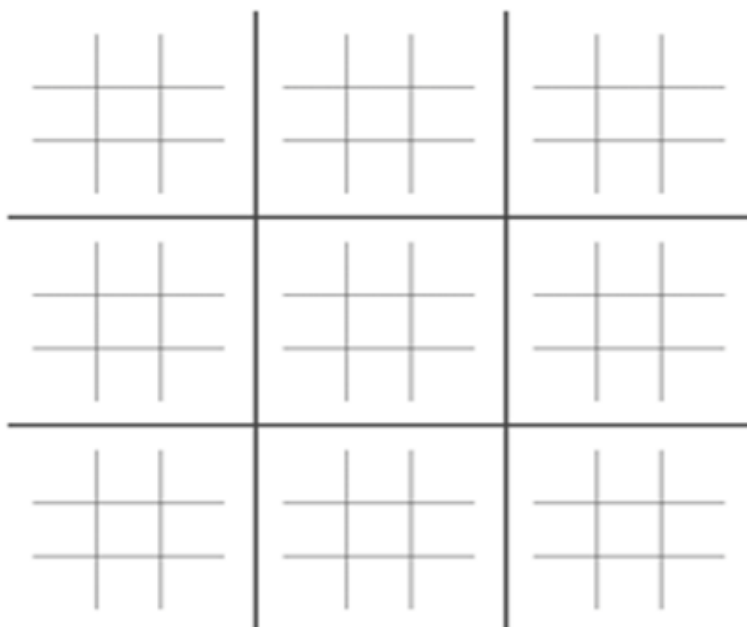
خروجی نمونه

```
15.5
11.7
94102 94103 94101
```



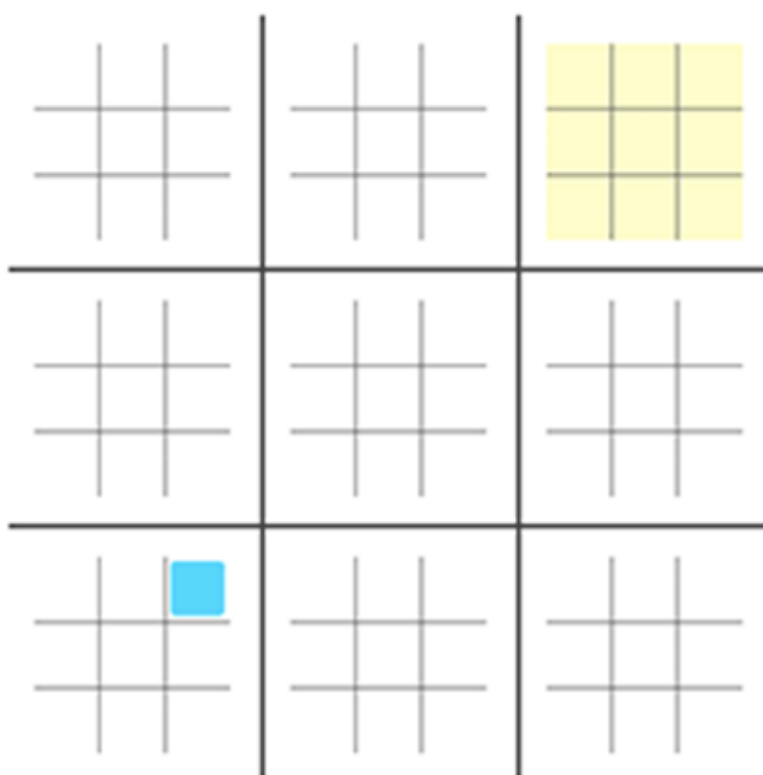
۲ بازی سزار

سزار پس از پیروزی بر دشمنان خود (با کمک برنامه ی رمزنگار شما) حوصله اش سر رفته و دنبال یک بازی است. او عاشق بازی tic tac toe یا همان ایکس-او است برای همین از شما خواهش کرده است تا یک بازی ایکس-او برایش بنویسید منتها با قوانین خودش. ایکس-او سزار مانند ایکس-او های معمولی یک جدول 3×3 است با این تفاوت که درون هر کدام از خانه های جدول یک جدول دیگر نیز قرار دارد!



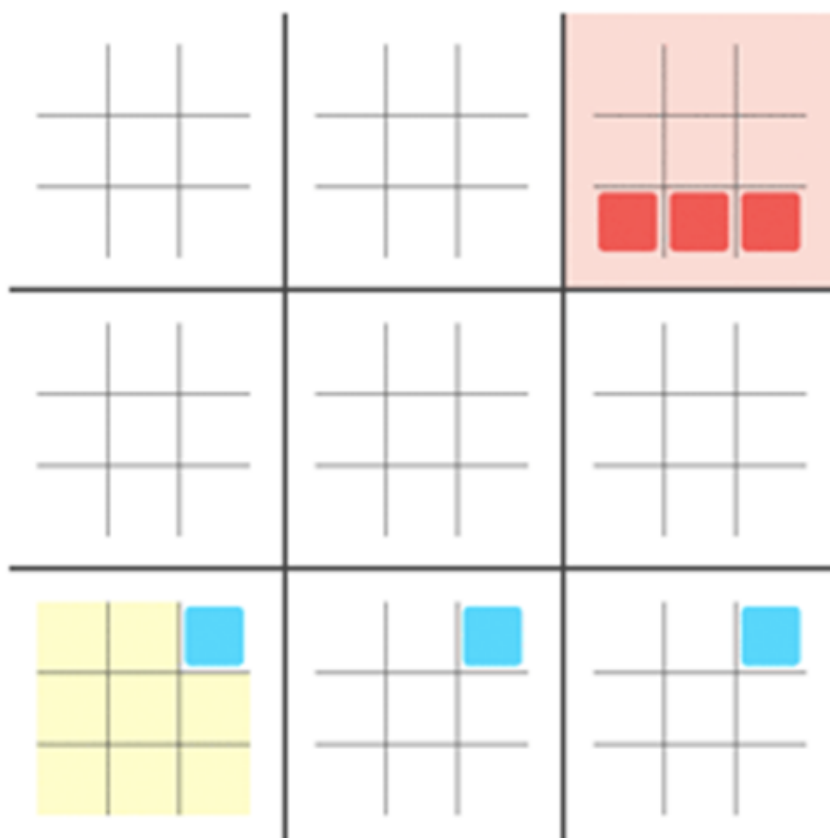


قوانین این بازی به این صورت است که بازیکن اول (در حقیقت حرکت اول) در هر کدام از این خانه ها حرکت خود را انجام می دهد اما بازیکن دوم موظف است تا در همان خانه ای که بازیکن اول بازی خود را شروع کرده اما در جدول «بزرگ» به بازی خود ادامه دهد. به شکل زیر: (یعنی بازیکن دوم حتما باید حرکت خود را در خانه ی های- لایت شده ادامه دهد)





بازی در خانه های کوچک ادامه پیدا میکند تا زمانی که در خانه های کوچک یکی از بازیکن ها برنده میشود و آن خانه متعلق به آن بازیکن می شود.



قوانین اضافه::

اگر آن خانه ی کوچک که شما باید در آن بازی کنید تکلیفش مشخص شده باشد آنگاه انتخاب با شماست تا در کدام خانه بازی کنید
اگر یکی از خانه های کوچک جدول مساوی شده باشد برای هیچ کدام از بازیکن ها حساب نمیشود و بازی ادامه پیدا میکند.
برای آشنایی بیشتر میتوانید این لینک را مشاهده کنید : (لینک)



البته بازی ای که باید بنویسید، صرفا خود بازی ایکس-او نیست. بلکه باید مکانیزم حساب کاربری و ثبت نام و لاگین و... را مطابق مواردی که در ادامه برای شما گفته می شود، پیاده سازی کنید. در این برنامه، در اصل سه منو (Menu) داریم. یکی منوی ابتدای بازی یا منوی ثبت نام است. یکی منوی اصلی و یکی هم منوی خود بازی. در ادامه به توضیح دستوراتی که در هر کدام از این منوها وارد می شود می پردازیم. توجه کنید که در هنگام شروع برنامه، به طور پیش فرض، کاربر در منوی ثبت نام قرار دارد. در تمامی منوها دستور help وجود دارد که توضیح آن را در هر منو قرار داده ایم. ضمنا توجه کنید تمامی پیغام های انجام موفقیت آمیز دستورات یا خطاها، در یک خط مجزا چاپ می شوند و بعد به خط بعدی می رود.

منوی ثبت نام

در ابتدای شروع برنامه کاربر در این منو قرار دارد.

```
register [username] [password]
```

همان طور که مشخص است، یک کاربر با نام کاربری و پسورد مشخص شده را ایجاد می کند. نام کاربری و پسورد باید فقط شامل حروف الفبای انگلیسی، اعداد و کاراکتر آندرلاین باشند. خطاهای مربوط به این دستور به این ترتیب چک می شوند و هر خطا که رخ داده بود، پیغام همان خطا چاپ شده و سایر خطاها بررسی نمی شوند. اگر هیچ خطایی رخ نداد و عملیات موفقیت آمیز بود، پیغام

```
register successful
```

چاپ خواهد شد.

خطاها:

اگر نام کاربری شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیغام

```
username format is invalid
```



اگر پسورد شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیام

password format is invalid

اگر کاربری با username گفته شده از قبل وجود داشت پیام

a user exists with this username

چاپ خواهد شد.

login [username] [password]

برای لاگین کردن به حساب کاربری مشخص با نام کاربری و پسورد داده شده استفاده می شود. نام کاربری و پسورد باید فقط شامل حروف الفبای انگلیسی، اعداد و کاراکتر آندرلاین _ باشند. خطاهای مربوط به این دستور به این ترتیب چک می شوند و هر خطا که رخ داده بود، پیام همان خطا چاپ شده و سایر خطاها بررسی نمی شوند. اگر هیچ خطایی رخ نداد و عملیات موفقیت آمیز بود، پیام

login successful

چاپ خواهد شد و پس از آن کاربر به طور خودکار وارد منوی اصلی خواهد شد.
خطاها:

اگر نام کاربری شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیام:

username format is invalid

اگر پسورد شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیام:

password format is invalid

اگر کاربری با username گفته شده وجود نداشت:

no user exists with this username

اگر پسورد غلط بود:



incorrect password

چاپ خواهد شد.

remove [username] [password]

برای حذف کردن یک حساب کاربری مشخص با نام کاربری و پسورد داده شده استفاده می شود. نام کاربری و پسورد باید فقط شامل حروف الفبای انگلیسی، اعداد و کاراکتر آندرلاین _ باشند. خطاهای مربوط به این دستور به این ترتیب چک می شوند و هر خطا که رخ داده بود، پیغام همان خطا چاپ شده و سایر خطاها بررسی نمی شوند. اگر همه موفق آمیز بودند پیغام

removed [username] successfully

چاپ خواهد شد که به جای username باید نام کاربری فرد حذف شده نشان داده شود.

خطاها:

اگر نام کاربری شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیغام:

username format is invalid

اگر پسورد شامل کاراکترهایی به جز کاراکترهای ذکر شده بود، پیغام:

password format is invalid

اگر کاربری با username گفته شده وجود نداشت:

no user exists with this username

اگر پسورد غلط بود:

incorrect password

چاپ خواهد شد.



```
list_users
```

این دستور لیست تمامی کاربرانی که وجود دارند را به ترتیب الفبایی (Lexicographical) نمایش می دهد.

```
help
```

این فرمان انواع دستوراتی که در این بخش قابل نمایش هستند را نمایش می دهد. از این دستور صرفاً برای اطمینان حاصل کردن از این که در منوی درست قرار دارید. خروجی این دستور در این منو به صورت زیر است:

```
register [username] [password]
login [username] [password]
remove [username] [password]
list_users
help
exit
```

```
exit
```

بیانگر اتمام اجرای برنامه است و بعد از چاپ پیام

```
program ended
```

اجرای برنامه پایان می پذیرد.

منوی اصلی:

همان طور که گفته شد، در صورت موفقیت آمیز بودن login وارد این منو خواهید شد. دستورات:

```
new_game [username] [limit]
```

با این دستور یک بازی جدید شروع می شود. در این بخش باید username یک کاربر دیگر وارد شود و بدین ترتیب بازی با آن کاربر شروع خواهد شد. کاربری که اکنون لاگین کرده و دستور را زده است در حین بازی به عنوان X و بازیکنی که در این دستور نام او وارد



شده به عنوان O خواهد بود. Limit هم یک عدد است و بیانگر محدودیت تعداد حرکات (نوبت بازی) است. اگر این عدد ۰ باشد به معنی نبودن هیچ محدودیتی در بازی است. در بخش مربوط به limit جزئیات بیشتر این دستور را توضیح خواهیم داد. خطاهای این دستور:

اگر username از کاراکترهایی که در منوی لاگین توضیح دادیم تشکیل نشده بود، پیام:

```
username format is invalid
```

اگر limit عددی کوچکتر از ۰ بود:

```
number should be positive to have a limit or 0 for no limit
```

اگر کاربری که دستور را زده است، نام کاربری خودش را وارد کرد:

```
you must choose another player to start a game
```

اگر کاربری با این نام کاربری وجود نداشت:

```
no user exists with this username
```

چاپ خواهد شد.

در صورت اجرای موفقیت آمیز دستور، پیام:

```
new game started successfully between [first] and [second] with limit [limit]
```

چاپ می شود که در آن first نام کاربری بازیکن X و second نام کاربری بازیکن O و limit هم عدد محدودیت بازی است (در حالتی که ۰ وارد شده بود به معنی عدم محدودیت هم عدد ۰ نمایش داده خواهد شد و استثنایی وجود ندارد).

```
scoreboard
```

پیش از توضیح خروجی این دستور باید در مورد امتیاز دهی در بازی صحبت بکنیم. هر برد معمولی (از طریق زدن شاه) ۳ امتیاز و باخت به این شکل ۰ امتیاز دارد. در صورتی که یکی از بازیکنان انصراف بدهد (با دستوری که در بخش مربوط به بازی توضیح می دهیم)،



برنده بازی ۲ امتیاز دریافت کرده و شخصی که انصراف داده ۱ امتیاز منفی کسب می کند. در صورت تساوی بازی (که در اثر اتمام limit اتفاق می افتد) هر بازیکن ۱ امتیاز می گیرد. بعد از اجرای این دستور باید کاربران با این فرمت نوشته بشوند:

```
[username] [score] [wins] [draws] [losses]
```

منظور از wins و draws و losses تعداد بردها، تساوی ها و باخت هاست. ترتیب مرتب سازی کاربران هم به ترتیب از بالاترین اولویت به کمترین به این صورت است: بیشترین امتیاز در صورت برابری امتیاز بیشترین تعداد برد در صورت برابری تعداد برد بیشترین تعداد تساوی در صورت برابری تساوی کمترین باخت و در صورت برابری تمامی موارد، براساس حروف الفبا (lexicographical) به طور صعودی (یعنی a زودتر از z می آید و...))

```
list_users
```

کاملاً مشابه همین دستور که در منوی قبلی توضیح داده شد، عمل می کند.

```
help
```

مشابه همان چیزی است که در بخش قبل توضیح دادیم. خروجی آن عیناً به این شکل است:

```
new_game [username] [limit]
scoreboard
list_users
help
logout
```

```
logout
```

با این دستور کاربر از حساب کاربری خود خارج می شود. پس از اجرای این دستور پیام

```
logout successful
```

چاپ شده و کاربر وارد منوی ثبت نام که پیش تر توضیح دادیم، می شود.



منوی بازی:

(فرض کنید نفر اول دایره با علامت O و نفر دوم ضربدر با علامت X است)

select board [x]

این دستور یکی از ۹ زمین کوچک تر بازی را انتخاب می کند ، زمین ها از سمت چپ از ۱ تا ۹ شماره گذاری شده اند.
خطاهای این دستور:
اگر مختصات به طور کلی از محدوده خارج بود (یعنی در بازه ۱ تا ۹ نبود) پیغام:

wrong coordination

در صورتی که بازی جدول انتخاب شده تمام شده باشد و یکی از دو بازیکن ان را برده باشد پیغام:

[player_name] has already won on this board

در صورتی که جدول انتخاب شده مساوی شده باشد پیغام:

this board is tie

چاپ می شود
دقت کنید که در اکثر دقت کنید که در اکثر حالات بردی که کاربر باید در ان بازی کند فقط یک برد خاص است اما کاربر می تواند برد های خالی دیگر را هم انتخاب کند و دستور select اروری نمی دهد.

deselect

این دستور برد انتخاب شده را از حالت انتخاب شده خارج می کند.
خطای دستور:
در صورتی که بردی انتخاب نشده باشد پیغام:

no board is selected

چاپ می شود. (دقت کنید که بعد از اتمام نوبت هر فرد به صورت پیش فرض زمین بازی برد انتخاب شده ندارد)



put mark [x]

این دستور در خانه x برد انتخاب شده علامت بازیکن (ایکس یا او) را می گذارد
در صورتی که بردی انتخاب نشده باشد پیغام:

no board selected

چاپ می شود.
در صورتی که با گذاشتن علامت بازیکن جدول را ببرد پیغام:

board [board_number] won by [player_name]

چاپ می شود.
در صورتی که با گذاشتن علامت برد بدون بردن کسی پر شود پیغام:

board [board_number] is a tie

چاپ می شود.
خطاهای این دستور:
در صورتی که در خانه ی موردنظر علامتی وجود داشت و یا براساس قوانین توضیح داده
شده نمی توانستیم در آن برد علامت بگذاریم پیغام:

cannot put mark here

چاپ می شود.

next turn

نوبت را به حریف منتقل می کند و در صورت اجرای موفقیت آمیز پیغام:

turn completed

چاپ می شود. توجه کنید که طبق قوانین شطرنج هر نفر باید در نوبت خود حتماً یک
حرکت انجام بدهد. بنابراین اگر هیچ حرکتی انجام نشده بود، پیغام خطای:

you must move then proceed to next turn



نوشته شود.

show turn

این دستور برای مشخص شدن این است که نوبت کدام بازیکن است. خروجی آن به فرم زیر است:

its player [player_name] turn with mark [player_mark]

undo

یکی از ویژگی های منحصر به فرد شطرنجی که باید پیاده سازی کنید، قابلیت undo است. هر بازیکن در کل طول یک بازی می تواند دو بار از این قابلیت و در هر نوبت حداکثر یک بار استفاده کند. این قابلیت بدین صورت است که اگر بازیکنی حرکتی انجام داده باشد، در صورتی که بخواهد و طبق شرایط بالا تعداد دفعات استفاده اش تمام نشده باشد، حرکت خود را برگرداند. توجه کنید که چون تغییر نوبت با next_turn انجام می شود، همچنان بعد از حرکت نوبت با بازیکن فعلی است تا زمانی که دستور next_turn زده شود. خطاهای این دستور: اگر پیش تر به اندازه تعداد کل undo های مجاز (یعنی ۲) بار از این قابلیت استفاده شده باشد.

you cannot undo anymore

اگر بازیکن در این نوبت حرکتی انجام نداده باشد که بخواهد undo کند:

you must move before undo

اگر در همین نوبت از undo استفاده کرده باشد:

you have used your undo for this turn

چاپ خواهند شد.

در صورت اجرای درست دستور نیز باید عبارت:

undo completed



چاپ بشود.
توجه کنید که بعد از اجرای undo، مهره‌ای که از قبل select شده همچنان در همین حالت باقی می ماند.

undo number

تعداد دفعات باقی مانده undo یک بازیکن را نشان می دهد. در ابتدای بازی این عدد دو است و با انجام حرکت undo کاهش می یابد تا در نهایت بعد از دو بار انجام این حرکت، به صفر می رسد. فرم پیام خروجی به صورت:

you have [n] undo moves

خواهد بود که n تعداد undo های باقی مانده است.

show moves

این دستور کل حرکاتی که بازیکن فعلی انجام داده است را به فرمت زیر نشان می دهد.

[player_mark] [board_number] [inner_board_number]

منظور از متغیر دوم شماره برد است و متغیر سوم نشان دهنده شماره خانه ی برد است.

show moves -all

این دستور مشابه دستور قبلی است با این تفاوت که حرکات تمامی بازیکنان را به ترتیب از ابتدای شروع بازی نمایش می دهد.

show board

این دستور صفحه ی بازی را به فرمت زیر نشان می دهد (O نشان دهنده دایره و X نشان دهنده ضرب در و E نشان دهنده خانه خالی است)



```
E|E|E E|E|E E|E|E
E|O|E E|O|E E|E|E
E|E|E E|E|E E|E|E
E|E|E X|X|X E|E|E
E|E|E X|X|X E|E|E
E|E|E X|X|X E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
```

دقت کنید اگر یکی از برد ها توسط یکی از بازیکن ها برنده شده باشد تمام خانه های ان برد علامت ان بازیکن گذاشته شود در صورت مساوی لازم نیست کاری انجام دهید و صرفاً برد را نشان دهید

help

مشابه دستور help در سایر بخش هاست. در این جا باید عیناً این خروجی نمایش داده شود:

```
select board [x]
deselect
put mark [x]
next turn
show turn
undo
undo number
show moves [-all]
show board
forfeit
help
```




forfeit

این دستور مربوط به انصراف از بازی می شود و بازیکنی که این دستور را وارد کند، بازنده بازی خواهد بود و نفر دیگر برنده بازی خواهد بود. همان طور که بالاتر گفتیم، با این اتفاق، حریف که برنده بازی شده، ۲ امتیاز می گیرد و کسی که انصراف داده منفی ۱ امتیاز دریافت می کند (یعنی ۱ امتیاز از او کم می شود).
بعد از اجرای این دستور باید این دو پیام در دو خط پشت سرهم نمایش داده شوند:

you have forfeited

player [player_name] with mark [player_mark] won

در اینجا دستورات منوی بازی تمام می شود به موارد زیر توجه داشته باشید در صورت برد یکی از بازیکنان که باید بعد از دستور turn next بررسی بشود بازی به پایان می رسد و پیغام زیر نمایش داده می شود.

player [player_name] with mark [player_mark] wonn

اگر تمام خانه های بازی پر شود و کسی برنده نشود بازی به پایان می رسد و دستور زیر چاپ می شود.

draw

و کاربر به منو اصلی باز گردانده می شود.
همان طور که پیش تر هم گفتیم، در اثر تساوی به هر کدام از بازیکنان ۱ امتیاز داده می شود.

اگر در هر کدام از منو ها دستوری زده شد که با قالب دستور های همان منو نمی خواند پیغام زیر چاپ شود

invalid command

در یک خط چاپ شود و به خط بعدی برویم.



دقت داشته باشید اگر خطای مربوطه یکی از خطاهای دستورات آن منو باشد دیگر این پیغام چاپ نشود و پیغام مربوطه به خطا چاپ شود
در مورد خطاهای دستورات، توجه کنید که ترتیب چک شدن آن‌ها به ترتیبی است که در این مستند نوشته شده‌اند و در صورت رخ دادن اولین خطا و چاپ پیام خطا، سایر خطاها بررسی نمی‌شوند.

ورودی نمونه

```
register ceaser 1234
register decimus 1234
register brutus 1234
register ali*& 1234
register ceaser 1234
login amir 1234
remove decimus 1234
login decimus 1234
register decimus 1234
list_users
help
login ceaser 1234
new_game amir 0
new_game amir -100
new_game brutus 0
select board 1
put mark 5
next turn
put mark 5
select board 5
put mark 3
show board
next turn
select board 3
put mark 5
next turn
show turn
next turn
```



```
select board 5
put mark 7
next turn
select board 7
put mark 5
next turn
select board 5
put mark 5
next turn
show moves
show moves -all
show board
forfeit
scoreboard
logout
exit
```

خروجی نمونه

```
register successful
register successful
register successful
username format is invalid
a user exists with this username
no user exists with this username
removed decimus successfully
no user exists with this username
register successful
brutus
ceaser
decimus
register [username] [password]
login [username] [password]
remove [username] [password]
list_users
help
exit
login successful
```



```
no user exists with this username
number should be positive to have a limit or 0 for no limit
new game started successfully between ceaser and brutus with limit 0
turn completed
no board selected
E|E|E E|E|E E|E|E
E|O|E E|E|E E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|X E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
E|E|E E|E|E E|E|E
turn completed
turn completed
its player ceaser turn with mark X
you must play then proceed to next turn
turn completed
turn completed
board 5 won by ceaser
turn completed
O 1 5
O 3 5
O 7 5
O 1 5
X 5 3
O 3 5
X 5 7
O 7 5
X 5 5
E|E|E E|E|E E|E|E
E|O|E E|E|E E|O|E
E|E|E E|E|E E|E|E
E|E|E X|X|X E|E|E
E|E|E X|X|X E|E|E
E|E|E X|X|X E|E|E
E|E|E E|E|E E|E|E
E|O|E E|E|E E|E|E
E|E|E E|E|E E|E|E
```



```
you have forfeited  
player ceaser with mark X won  
ceaser 2 1 0 0  
decimus 0 0 0 0  
brutus -1 0 0 1  
logout successful  
program ended
```



۳ فیفا ۲۱

در این سوال ، باید یک لیگ فوتبال (مانند بازی فیفا) را پیاده سازی کنید . در این لیگ ، تعدادی تیم وجود دارد که هر تیم از چند بازیکن تشکیل شده است .

لیگ

همان طور که اشاره شد ، در این لیگ ، تعدادی تیم وجود دارد . و همچنین بازیکن هایی که تیم ندارند در لیگ قرار می گیرند . در واقع ، تمام کارهای اصلی (از جمله خرید و فروش بازیکن ، انجام بازی های دوستانه ، و سپری کردن یک فصل از لیگ) در سازمان لیگ انجام می شود .

تیم

هر تیم ، یک نام ، مقداری پول به عنوان بودجه ، یک ترکیب اصلی و تعدادی بازیکن ذخیره دارد .

ترکیب اصلی (squad)

در ترکیب اصلی ۱۱ بازیکن که قرار است در بازی ها حاضر شوند ، قرار می گیرند . برای راحتی کار ، تمام ترکیب ها از آرایش 4-3-3 برخوردارند ؛ ۱ دروازه بان - ۴ مدافع ۳ - هافبک و ۳ مهاجم .

بازیکن

هر بازیکن ، نام ، نام خانوادگی ، سن ، و یک قرارداد دارد . ۴ پست مختلف برای بازیکن ها وجود دارد ؛ دروازه بان ، مدافع ، هافبک ، و مهاجم . هر بازیکن به نسبت پست خودش ، معیارهایی برای سنجش توانایی هایش دارد .

دروازه بان

معیارهای هر دروازه بان عبارت اند از :

- قدرت شوت گیری (Shoot Saving)
- واکنش نشان دادن (Reactions)
- پنالتی گیری (Penalty Saving)



مدافع

معیارهای هر مدافع عبارت اند از:

- قدرت بدنی (Strength)
- خشونت (Aggression)
- سر زنی (Heading)

هافبک

معیارهای هر هافبک عبارت اند از:

- پاس کاری (Passing)
- شوت زنی (Shooting)
- سانتر کردن (Crossing)

مهاجم

معیارهای هر مهاجم عبارت اند از:

- سر زنی (Heading)
- تمام کنندگی (Finishing)
- پنالتی زنی (Penalties)

هر بازیکن در زمان ایجاد شدن (پیوستن به دنیای فوتبال)، یک بازیکن آزاد محسوب می شود. زمانی که بازیکن به تیمی بپیوندد قراردادی امضا می کند که در آن قرارداد، تعداد سال هایی که بازیکن با آن تیم قرارداد دارد ذکر شده است. همچنین یک نوع قرارداد قرضی نیز داریم که یک بازیکن در صورتی که در یک تیم مشغول به فعالیت باشد، می تواند برای مدتی در تیمی دیگر بازی کند و پس از اتمام قرارداد قرضی، به تیم اولیه خود بازگردد. مدت قرارداد قرضی نباید از مدت قرارداد اصلی بیشتر باشد.

در هر فصل از لیگ، هر تیم با تمام تیم های دیگر دو بار بازی می کند. یک بار در خانه ی خودش و بار دیگر در خانه ی حریف. در صورتی که تیمی در یک بازی مهمان باشد، از معیار های هر بازیکن آن ۵ تا کم می شود.

هر تیم، مقداری بودجه دارد که از آن برای خرید بازیکن استفاده می کند. هزینه ی خرید بازیکن ها از بودجه ی تیم مقصد پرداخت شده و به بودجه ی تیم مبدا واریز می شود و بازیکن



خریداری شده به لیست بازیکنان ذخیره‌ی تیم جدید منتقل می‌شود.
بودجه‌ی تیم‌ها و هزینه‌های مشخص شده همگی بر حسب میلیون دلار هستند.

نتیجه هر بازی به این صورت محاسبه می‌شود:

- در صورتی که میانگین قدرت پاس‌کاری هافبک‌ها از ۸۵ بیشتر باشد، موقعیت‌های بهتری نصیب مهاجم‌ها می‌شود و به قدرت تمام‌کنندگی هر مهاجم ۵ تا اضافه می‌شود. همچنین اگر هافبکی در تیم باشد که قدرت پاس‌کاریاش از ۹۰ بیشتر باشد، به قدرت تمام‌کنندگی هر مهاجم ۵ تا اضافه می‌شود.
- در صورتی که میانگین قدرت بدنی مدافع‌های حریف از ۸۵ بیشتر باشد، آن‌ها در مصاف‌هایی که در مقابل مهاجمان دارند پیروز می‌شوند و از قدرت تمام‌کنندگی هر مهاجم ۳ تا کم می‌شود. همچنین اگر مدافعی در تیم باشد که قدرت بدنی‌اش از بیشتر از ۹۰ باشد از قدرت تمام‌کنندگی هر مهاجم ۳ تا کم می‌شود.
- به ازای هر مهاجم، اگر قدرت تمام‌کنندگی او از قدرت واکنش نشان دادن دروازه‌بان حریف بیشتر باشد، آن مهاجم یک گل به ثمر می‌رساند. و اگر میانگین قدرت تمام‌کنندگی مهاجم‌ها از قدرت واکنش نشان دادن دروازه‌بان حریف حداقل ۵ تا بیشتر باشد، تیم حمله‌کننده ۲ گل به ثمر می‌رساند.
- به ازای هر هافبک، اگر قدرت شوت زنی او از قدرت شوت‌گیری دروازه‌بان حریف بیشتر باشد، او یک گل به ثمر می‌رساند.
- به ازای هر مدافع یا مهاجم، اگر قدرت سرزنی او ضرب در میانگین قدرت سانتر هافبک‌ها، از مجذور قدرت واکنش نشان دادن دروازه‌بان حریف بیشتر باشد، آن بازیکن یک گل به ثمر می‌رساند.
- اگر میانگین خشونت مدافعان حریف از ۸۵ بیشتر باشد، تیم حمله‌کننده صاحب یک ضربه پنالتی می‌شود و مهاجمی که بیشترین قدرت پنالتی‌زنی را دارد پشت توپ می‌ایستد. در صورتی که قدرت پنالتی‌زنی او از قدرت پنالتی‌گیری دروازه‌بان حریف بیشتر باشد، پنالتی تبدیل به گل می‌شود، و در غیر این صورت دروازه‌بان پنالتی را می‌گیرد.
- پس از محاسبه‌ی تعداد گل‌های هر تیم، برنده و بازنده‌ی آن بازی (یا تساوی تیم‌ها) مشخص می‌شود. هر برد ۳ امتیاز و تساوی ۱ امتیاز دارد. باخت هم امتیازی ندارد.



- در انتهای فصل، به تیم های اول، دوم و سوم به ترتیب ۵۰، ۱۰۰ و ۲۰ میلیون دلار پاداش داده می شود. بازیکنانی که قراردادشان تمام شده به لیست بازیکنان آزاد منتقل می شوند و بازیکنانی که با قرارداد قرضی در تیم دیگری بازی می کردند به تیم اصلی بازمی گردند. همچنین سن تمام بازیکنان یک سال افزایش می یابد. بازیکنانی که ۴۰ سال یا بیشتر سن دارند بازنشسته شده و از فوتبال خداحافظی می کنند.

دستورات ورودی

```
create team teamName
```

یک تیم با نام مشخص شده می سازد. بودجه ی اولیه ی تیم ۱۰۰ میلیون دلار است. در صورتی که تیمی با این نام وجود داشته باشد `a team exists with this name` چاپ می شود. در صورتی که تیم با موفقیت ساخته شود `team created` چاپ می شود.

```
delete team teamName
```

تیم مورد نظر را منحل می کند. در صورتی که تیمی با این نام وجود نداشته باشد `invalid team`، چاپ می شود. پس از حذف تیم مورد نظر، بازیکنان آن به بازیکنان آزاد منتقل می شوند و قراردادهای آنها نیز فسخ می شود.

```
create player firstName lastName age position
```

دقت کنید که `position` باید با یکی از عبارات `ST MF DF GK` باشد.

GK = دروازه بان

MF = هافبک

DF = دفاع

ST = مهاجم

اگر بازیکنی با این نام وجود داشته باشد `a player exists with this name` چاپ می شود. اگر ورودی مشکلی نداشت، به نسبت پست بازیکن، معیارها را دریافت می کند. به



این صورت که به ترتیب آورده شده در بالا معیارها چاپ می شود و مقادیر مورد نظر از ورودی دریافت می شود. مثلاً برای یک مدافع، عبارات زیر در خروجی چاپ شده و پس از چاپ شدن هر خط، مقدار مربوطه برای هر معیار دریافت می شود:

Strength:
Aggression:
Heading:

زمانی که بازیکن که ساخته شد player created چاپ شده و بازیکن به لیست بازیکن های آزاد منتقل می شود.

```
print free players
```

لیست بازیکن های آزاد را ابتدا به بر اساس پست (به ترتیب دروازه بان، مدافع، هافبک، مهاجم) و سپس نام خانوادگی و نام مرتب کرده و چاپ می کند.
مثال:

```
1. Mohammad Mohammadi GK  
2. Ali Alavi DF  
3. Reza Rezaei DF  
...
```

```
move free player firstName lastName to teamName with #years years contract
```

در صورتی که تیم یافت نشود invalid team چاپ می شود.
در صورتی که بازیکن آزاد مورد نظر یافت نشود invalid free player چاپ می شود.
در صورت قابل قبول بودن ورودی ها، بازیکن آزاد مورد نظر با قرارداد years ساله به تیم مشخص شده منتقل می شود و پیام free player moved چاپ می شود.

```
print stats of team teamName
```

مشخصات یک تیم از جمله نام تیم، بودجه ی آن، و بازیکنان اصلی و ذخیره آن تیم را چاپ می کند. در صورتی که تیم یافت نشود invalid team چاپ می شود.



ترتیب بازیکن ها باید بر اساس پست (دروازه بان، مدافع، هافبک، مهاجم)، و سپس نام خانوادگی و نام مرتب شود.

Team: Chelsea

Budget: 100

Squad players:

1. Kepa Arrizabalaga GK Age: 24 Contract: 6 years

2. Marcos Alonso DF Age: 28 Contract: 4 years

3. Cesar Azpilicueta DF Age: 29 Contract: 3 years

Reserve players:

1. Willy Caballero GK Age: 37 Contract: 1 years

2. Gary Cahill DF Age: 33 Contract: 1 years

print stats of player firstName lastName

مشخصات یک بازیکن را چاپ می‌کند در صورتی که بازیکن نامعتبر باشد
invalid player چاپ می‌شود.
مثال:

Eden Hazard ST

Age: 28

Heading: 61

Finishing: 84

Penalties: 86

renew contract of firstName lastName for #years years

قرارداد بازیکن مورد نظر را به مدت مشخص شده تمدید می‌کند. در صورتی که بازیکن مورد نظر، قراردادی نداشته باشد، پیغام invalid renewal command چاپ می‌شود. در صورتی که بازیکن یک قرارداد قرضی داشته باشد، قراردادش با تیم اصلی تمدید می‌شود. در صورت موفقیت آمیز بودن تمدید، پیام زیر چاپ می‌شود:

contract renewed. new contract is valid for #years years

در صورتی که بازیکن یافت نشود، پیغام invalid player چاپ می‌شود.



```
terminate contract of firstName lastName
```

قرارداد بازیکن مورد نظر را فسخ می کند. در صورتی که بازیکن مورد نظر، قراردادی نداشته باشد، پیغام `invalid termination command` چاپ می شود. در صورتی که بازیکن یک قرارداد قرضی داشته باشد، قرارداد قرضی اش فسخ می شود. در صورتی که بازیکن یافت نشود، پیغام `invalid player` چاپ می شود.

```
sell player firstName lastName from sourceTeam to destinationTeam for  
#price$ with #years years contract
```

بازیکن مورد نظر را از تیم مبدا به تیم مقصد با مبلغ و طول قرارداد مشخص شده منتقل می کند. در صورتی که تیم مبدا نامعتبر باشد `invalid source team`، چاپ می شود. در غیر این صورت، اگر بازیکن مورد نظر در تیم مبدا وجود نداشته باشد، `invalid player` چاپ می شود. در صورتی که تیم مبدا و بازیکن معتبر باشند ولی تیم مقصد نامعتبر باشد `invalid destination team` چاپ می شود. در صورتی که تیم مقصد بودجه ی لازم برای انتقال را نداشته باشد `insufficient budget`، چاپ می شود.

```
loan player firstName lastName from sourceTeam to destinationTeam with  
#years years contract
```

بازیکن مورد نظر از تیم مبدا به تیم مقصد و با مدت قرارداد مشخص شده قرض داده می شود. پیام های چاپ شده در صورت نامعتبر بودن ورودی ها مانند `sell` هستند. اگر طول قرارداد قرضی از طول قرارداد اصلی بیشتر بود پیام `invalid loan contract` چاپ شود.

```
put player firstName lastName from teamName in main squad
```

بدون این دستور امکان ندارد بازیکن ذخیره به ترکیب اصلی برود. بازیکن مورد نظر از تیم مشخص شده را در ترکیب اصلی قرار می دهد. در صورتی که تیم نامعتبر باشد `invalid team` چاپ می شود.



در صورتی که بازیکن در ترکیب اصلی تیم باشد
 player is currently in the main squad، چاپ می شود. در صورتی که بازیکن نه
 در ترکیب اصلی و نه در ذخیره ها باشد invalid player، چاپ می شود. در غیر این صورت
 ،بازیکن به ترکیب اصلی (و به پست خودش) منتقل می شود. نحوه ی قرار گرفتن بازیکن
 ها در ترکیب تیم ها به صورت First in-First out است، یعنی در صورتی که بازیکنان
 مربوط به یک پست تکمیل باشند، اولین بازیکنی که در آن پست به ترکیب اضافه شده، از
 ترکیب خارج می شود و بازیکن مورد نظر به ترکیب اضافه می شود.

friendly match between team1 and team2

یک بازی دوستانه میان تیم اول و تیم دوم برگزار می کند. در این حالت از بازی، اثرات
 مربوط به بازی در خانه حریف اعمال نمی شود. در صورتی که حداقل یکی از تیم ها نامعتبر
 باشد invalid team، چاپ می شود. در صورتی که هر تیم، ترکیب ناقصی داشته باشد، عبارت
 زیر (به نسبت هر تیمی که ترکیش ناقص است) چاپ می شود:

team1/team2 squad isn't complete

در پایان بازی، نتیجه بازی به صورت زیر چاپ می شود:

team1 #team1Goals – #team2Goals team2

next season

با این دستور، بازی های مربوط به فصل بعد انجام می شود و اعمال انتهای فصل (توضیح
 داده شده در بالا) انجام می شوند. در صورتی که ترکیب اصلی تیمی ناقص بود، به ترتیب
 الفبایی برای نام تیم ها، به ازای آن ها در خروجی عبارت

teamName squad isn't complete چاپ می شود. در انتها جدول لیگ چاپ می
 شود. ترتیب قرار گرفتن تیم ها در جدول به این صورت است که اولویت ابتدا با امتیاز است
 . سپس تفاضل گل و گل زده شده بررسی می شود و در صورتی که تمام موارد ذکر شده برابر
 بودند، تیم ها بر اساس ترتیب نام هایشان در جدول قرار می گیرند! یک نمونه برای جدول:

1. Chelsea 95 points W: 29 D: 8 L: 1 GF: 72 GA: 15 GD: +57

2. ...

اعداد ذکر شده در جدول، به ترتیب امتیاز تیم، تعداد بردها، تعداد تساوی ها، تعداد



باخت‌ها، تعداد گل‌های زده، تعداد گل‌های خورده و تفاضل گل هستند.

end

با خواندن این دستور، بازی پایان می‌یابد.
نکته : هر دستور غیر از موارد ذکر شده منجر به تولید invalid command می‌شود.

ورودی نمونه

```
create team liverpool
create team tottenham
create team everton
create team tottenham
delete team barcelona
delete team tottenham
create team arsenal
create player Mohammad Salah 28 ST
70
76
88
create player Lionel Messi 33 ST
65
83
92
create player Willian Borges 32 MF
80
64
70
create player Abbas Mammadian 26 GK
80
75
50
create player Willian Borges 32 ST
create player Neymar Jr 24 ST
75
74
88
```



```
create player Mehdi Shiri 30 DF
72
80
70
create player Sadio Mane 28 ST
78
75
80
create player Pashmak Pashmakian 22 GK
80
75
50
create player Hafez Hafezian 27 DF
75
74
88
create player Akbar zeydan 29 DF
72
87
70
create player Sardar Azmoon 23 ST
78
75
80
create player Adrian San 22 GK
60
60
40
create player Ali Parvin 18 DF
75
74
88
create player Dave RA 25 MF
72
82
73
create player Jordan Henderson 30 MF
78
75
80
```



```
create player Alisson Becker 28 GK
80
84
50
create player Mash Rajab 38 DF
61
74
70
create player Ali dayi 29 ST
72
87
89
create player Akbar Asghari 28 MF
78
75
80
create player Carlos Puyol 39 DF
70
75
87
create player Reza Rezayi 24 ST
75
74
88
create player Mehdi Abdi 30 DF
72
80
70
create player Mahdi Taromi 25 ST
78
75
80
create player Hamed Lak 31 GK
80
75
50
create player Jalal Hoseini 33 DF
75
74
88
```




```
create player Gerard Pique 33 DF
72
87
70
create player Cristiano Ronaldo 35 ST
78
75
80
create player Manuel Neuer 34 GK
76
80
54
create player Vahid Amiri 29 DF
75
74
88
create player Thiago Alcantara 30 MF
72
82
73
create player Shoja KhalilZade 30 MF
78
75
80
create player Luis Suarez 32 ST
80
84
68
create player Mash Gholam 36 DF
61
74
70
create player Mahdi Torabi 29 ST
72
87
89
create player Ali Mahdavi 28 MF
78
75
80
```



```
create player Trent Arnold 21 DF
70
75
87
create player Robert Lewandowski 32 ST
75
74
88
create player Thiago Silva 35 DF
72
80
70
create player Angel DiMaria 34 ST
78
75
80
create player namak namakian 24 MF
80
75
50
create player David Luiz 33 DF
75
74
88
create player M D 29 DF
72
87
70
create player N T 23 ST
65
67
50
create player M A 22 MF
70
86
88
create player K A 37 DF
75
74
88
```



create player Y S 25 MF

72

82

73

create player P M 30 MF

78

75

80

create player A M 28 DF

80

84

50

create player Mash Akbar 38 DF

61

74

70

create player Ali Akbari 29 MF

72

87

89

create player M J 28 MF

50

66

67

create player Y Z 20 DF

70

75

87

move free player Y Z to liverpool with 3 years contract

move free player M A to PSG with 2 years contract

move free player Y Z to arsenal with 1 years contract

move free player M A to liverpool with 6 years contract

move free player Mohammad Salah to liverpool with 3 years contract

move free player Alisson Becker to liverpool with 4 years contract

move free player Sadio Mane to liverpool with 3 years contract

move free player Ali Parvin to liverpool with 5 years contract

move free player Jordan Henderson to liverpool with 6 years contract

move free player Robert Lewandowski to liverpool with 3 years contract

move free player Carlos Puyol to liverpool with 3 years contract

move free player Thiago Alcantara to liverpool with 3 years contract



```
move free player David Luiz to liverpool with 3 years contract
move free player Trent Arnold to liverpool with 3 years contract
move free player Abbas Mammadian to arsenal with 2 years contract
move free player Lionel Messi to arsenal with 3 years contract
move free player Sardar Azmoon to arsenal with 4 years contract
move free player N T to arsenal with 3 years contract
move free player M J to arsenal with 3 years contract
move free player Ali Akbari to arsenal with 4 years contract
move free player namak namakian to arsenal with 2 years contract
move free player Mash Akbar to arsenal with 1 years contract
move free player M D to arsenal with 2 years contract
move free player K A to arsenal with 3 years contract
move free player Thiago Silva to arsenal with 2 years contract
move free player Pashmak Pashmakian to arsenal with 1 years contract
put player Y Z from liverpool in main squad
put player M A from liverpool in main squad
put player Mohammad Salah from liverpool in main squad
put player Alisson Becker from liverpool in main squad
put player Sadio Mane from liverpool in main squad
put player Ali Parvin from liverpool in main squad
print stats of player Mohammad Salah
print stats of team liverpool
put player Jordan Henderson from liverpool in main squad
put player Robert Lewandowski from liverpool in main squad
put player Carlos Puyol from liverpool in main squad
put player Thiago Alcantara from liverpool in main squad
put player Trent Arnold from liverpool in main squad
put player Abbas Mammadian from arsenal in main squad
put player Lionel Messi from arsenal in main squad
put player Sardar Azmoon from arsenal in main squad
put player N T from arsenal in main squad
put player M J from arsenal in main squad
put player Ali Akbari from arsenal in main squad
put player namak namakian from arsenal in main squad
put player Mash Akbar from arsenal in main squad
put player M D from arsenal in main squad
put player K A from arsenal in main squad
put player Thiago Silva from arsenal in main squad
put player Pashmak Pashmakian from arsenal in main squad
move free player Hamed Lak to everton with 3 years contract
```



```
move free player Adrian San to everton with 2 years contract
move free player Mehdi Abdi to everton with 4 years contract
move free player Vahid Amiri to everton with 1 years contract
move free player Hafez Hafezian to everton with 3 years contract
move free player Jalal Hoseini to everton with 3 years contract
move free player Akbar Asghari to everton with 2 years contract
move free player Willian Borges to everton with 4 years contract
move free player Shoja KhalilZade to everton with 5 years contract
move free player Y S to everton with 3 years contract
move free player Angel DiMaria to everton with 2 years contract
move free player Cristiano Ronaldo to everton with 2 years contract
move free player Mahdi Torabi to everton with 3 years contract
put player Hamed Lak from everton in main squad
put player Adrian San from everton in main squad
put player Mehdi Abdi from everton in main squad
put player Vahid Amiri from everton in main squad
put player Hafez Hafezian from everton in main squad
put player Jalal Hoseini from everton in main squad
put player Akbar Asghari from everton in main squad
put player Willian Borges from everton in main squad
put player Shoja KhalilZade from everton in main squad
put player Angel DiMaria from everton in main squad
put player Cristiano Ronaldo from everton in main squad
put player Mahdi Torabi from everton in main squad
terminate contract of Y S
print stats of team everton
next season
end
```

خروجی نمونه

```
team created
team created
team created
a team exists with this name
invalid team
team created
Heading:
```



Finishing:
Penalties:
player created
Heading:
Finishing:
Penalties:
player created
Passing:
Shooting:
Crossing:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
a player exists with this name
Heading:
Finishing:
Penalties:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
Strength:
Aggression:
Heading:
player created
Strength:
Aggression:
Heading:
player created



Heading:
Finishing:
Penalties:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
Strength:
Aggression:
Heading:
player created
Passing:
Shooting:
Crossing:
player created
Passing:
Shooting:
Crossing:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Passing:
Shooting:
Crossing:
player created
Strength:
Aggression:
Heading:
player created



Heading:
Finishing:
Penalties:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
Strength:
Aggression:
Heading:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Shot Saving:
Reactions:
Penalty Saving:
player created
Strength:
Aggression:
Heading:
player created
Passing:
Shooting:
Crossing:
player created



Passing:
Shooting:
Crossing:
player created
Heading:
Finishing:
Penalties:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Passing:
Shooting:
Crossing:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Passing:
Shooting:
Crossing:
player created



Strength:
Aggression:
Heading:
player created
Strength:
Aggression:
Heading:
player created
Heading:
Finishing:
Penalties:
player created
Passing:
Shooting:
Crossing:
player created
Strength:
Aggression:
Heading:
player created
Passing:
Shooting:
Crossing:
player created
Passing:
Shooting:
Crossing:
player created
Strength:
Aggression:
Heading:
player created
Strength:
Aggression:
Heading:
player created
Passing:
Shooting:
Crossing:
player created

Passing:
Shooting:
Crossing:
player created
Strength:
Aggression:
Heading:
player created
free player moved
invalid team
invalid free player
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
free player moved
Mohammad Salah ST
Age: 28
Heading: 70
Finishing: 76
Penalties: 88
Team: liverpool



Budget: 100

Squad players:

1. Alisson Becker GK Age: 28 Contract: 4 years
2. Ali Parvin DF Age: 18 Contract: 5 years
3. Y Z DF Age: 20 Contract: 3 years
4. M A MF Age: 22 Contract: 6 years
5. Sadio Mane ST Age: 28 Contract: 3 years
6. Mohammad Salah ST Age: 28 Contract: 3 years

Reserve players:

1. Trent Arnold DF Age: 21 Contract: 3 years
2. David Luiz DF Age: 33 Contract: 3 years
3. Carlos Puyol DF Age: 39 Contract: 3 years
4. Thiago Alcantara MF Age: 30 Contract: 3 years
5. Jordan Henderson MF Age: 30 Contract: 6 years
6. Robert Lewandowski ST Age: 32 Contract: 3 years

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

free player moved

Team: everton

Budget: 100

Squad players:

1. Adrian San GK Age: 22 Contract: 2 years
2. Mehdi Abdi DF Age: 30 Contract: 4 years
3. Vahid Amiri DF Age: 29 Contract: 1 years
4. Hafez Hafezian DF Age: 27 Contract: 3 years
5. Jalal Hoseini DF Age: 33 Contract: 3 years
6. Akbar Asghari MF Age: 28 Contract: 2 years
7. Willian Borges MF Age: 32 Contract: 4 years
8. Shoja KhalilZade MF Age: 30 Contract: 5 years
9. Angel DiMaria ST Age: 34 Contract: 2 years



10. Cristiano Ronaldo ST Age: 35 Contract: 2 years

11. Mahdi Torabi ST Age: 29 Contract: 3 years

Reserve players:

1. Hamed Lak GK Age: 31 Contract: 3 years

1. liverpool 12 points W: 4 D: 0 L: 0 GF: 47 GA: 7 GD: +40

2. arsenal 6 points W: 2 D: 0 L: 2 GF: 33 GA: 33 GD: 0

3. everton 0 points W: 0 D: 0 L: 4 GF: 20 GA: 60 GD: -40