دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

برنامه‌سازی پیشرفته

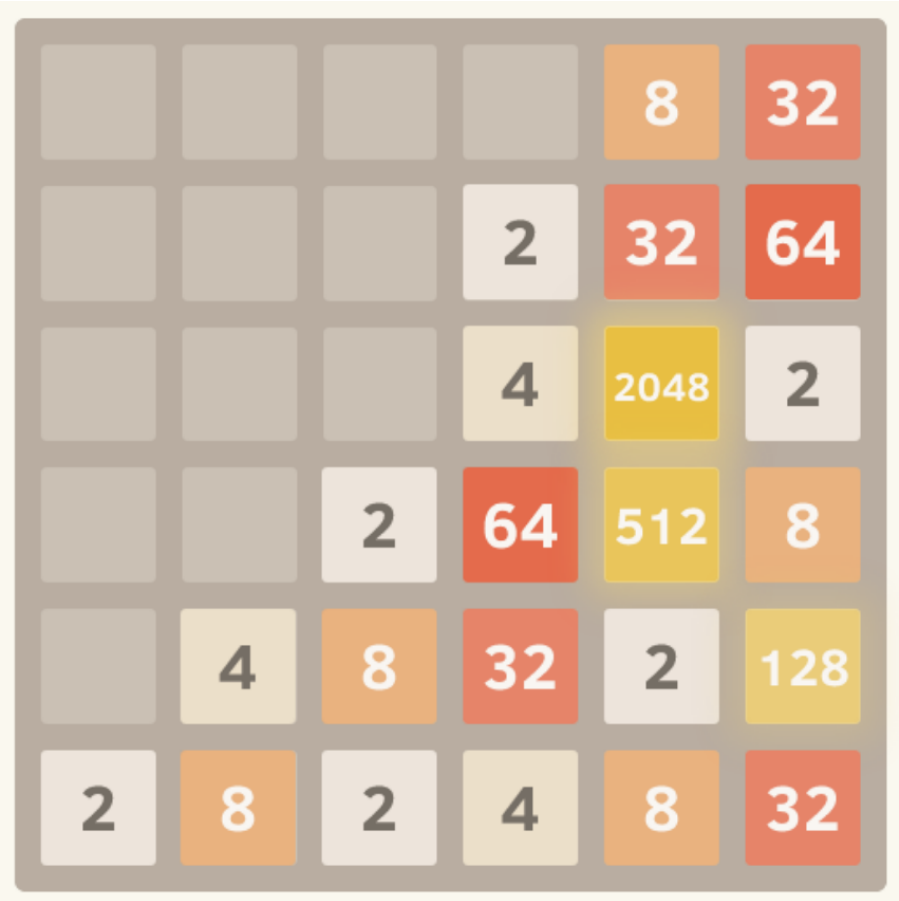
نیم‌سال دوم ۹۸-۹۷

تمرین سه	مفاهیم پیشرفته‌ی جاوا	مهلت ارسال: جمعه ۳ خرداد

به موارد زیر توجه کنید:

- **تنها به یکی از سوالات ۲ یا ۳ پاسخ دهید.**
- پاسخ تمرین را در سامانه‌ی کوئرا بارگذاری نمایید.
- مهلت ارسال پاسخ تمرین تا ساعت ۲۳:۵۹ روز اعلام‌شده است. توصیه می‌شود نوشتن تمرین را به روزهای نهایی موکول نکنید.
- مهلت ارسال پاسخ تمرین تحت هیچ شرایطی تمدید نخواهد شد.
- هم‌کاری و هم‌فکری شما در حل تمرین مانعی ندارد، اما پاسخ ارسالی هر کس حتما باید توسط خود او نوشته شده باشد. ارسال پاسخ تمرین‌ها ۴ روز فرصت ارسال با تاخیر و با جریمه‌ی کسر روزانه ٪۲۰ از نمره‌ی سوال در نظر گرفته شده‌است که برای محاسبه‌ی نمره‌ی سوال در نظر گرفته شده است که برای محاسبه‌ی نمره‌ی تمرین در مجموع ۴ روز از تاخیر تمرین‌ها بخشیده می‌شود.
- به ازای هر روز ارسال زودتر پاسخ هر سوال از تمرین، ٪۵ نمره‌ی مثبت تا سقف ۴ روز (حداکثر ٪۲۰) تعلق خواهد گرفت. (فقط در صورت کسب نمره‌ی کامل سوال که شامل نمره‌ی امتیازی نیز می‌شود)
- مبنای درس، اعتماد به پاسخ ارسالی توسط شماست و در نتیجه در صورت مشاهده‌ی هرگونه مشابهت غیرمتعارف بین کدها، برای هر دو طرف تقلب‌دهنده و تقلب‌گیرنده در مرتبه‌ی اول نمره‌ی ۱۰۰- برای آن تمرین و در صورت تکرار، حذف درس صورت خواهد گرفت.

۱۰۲۴ × ۲ (۳۰ + ۱۰ امتیاز)



در این سوال از شما خواسته می‌شود که بازی 2048 را به صورت گرافیکی پیاده سازی کنید. احتمالا با این بازی آشنا هستید اما برای آشنایی بیشتر می‌توانید از این لینک استفاده کنید. :

http://2048game.com/

منطق بازی

در هر بار فشار دادن یکی از جهت‌های کیبورد، اعداد تا جای ممکن در آن جهت حرکت می‌کنند و در صورتی که حرکت در آن جهت قابل انجام نباشد، تغییری در جدول مشاهده نمی‌شود. همچنین اعداد برابر، با حرکت در آن جهت در صورت نبودن مانع (عدد دیگر بین آنها) با هم جمع می‌شوند و توانی دیگر از ۲ را می‌سازند. در هر بار فشار دادن جهت‌ها، اگر خانه یا خانه‌هایی خالی در جدول موجود باشند، به صورت رندوم یکی از اعداد ۲ یا ۴ در یکی از خانه های خالی قرار می‌گیرد (در صورت وجود دو یا چند خانه‌ی خالی، انتخاب خانه‌ی خالی نیز رندوم خواهد بود). همچنین در هر مرحله، به اندازه‌ی حاصل‌جمع اعداد به دست آمده در آن مرحله (به جز عدد رندوم تولیدی) به امتیاز کاربر اضافه می‌شود. در نهایت، اگر جدول بازی پر بود و امکان حرکت در هیچ جهتی وجود نداشت، بازی با پیغام *Game Over* پایان می یابد و امتیاز نهایی کاربر نمایش داده می‌شود.

موارد اجباری

۱. پیاده سازی منطق بازی

۲. حرکت با استفاده از کلیدهای جهت (*arrow keys*)

۳. قرار دادن رنگ های متفاوت برای اعداد

۴. قرار دادن *scoreboard* و نمایش امتیاز کاربر پس از هر حرکت

۵. طراحی یک منوی ساده قبل از ورود به بازی (مثلا دارای گزینه‌هایی مانند $Play$ و $Quit$ باشد و با کلیک روی هر گزینه، عملیات مطلوب را انجام دهد)

موارد امتیازی

۱. گرفتن مقدار n از کاربر و طراحی بازی در جدول $n \times n$

۲. در نظر گرفتن پروفایل‌های مختلف برای کاربران و قراردادن نام کاربری برای هر کاربر و افزودن قابلیت تغییر نام کاربری (نام کاربری باید یکتا باشد و در صورت وجود نام کاربری مشابه، پیغام مناسب چاپ شود)

۳. جدول $Ranking$ که در آن کاربران بر اساس $high\ score$ مرتب شده اند.

۴. استفاده از انیمیشن برای حرکت‌ها و هر نکته‌ی گرافیکی دیگری که باعث زیباتر شدن طراحی شود.

چت‌روم ملی!! (۴۰ امتیاز)

در این سوال شما باید یک چت روم پیاده سازی کنید!
پیاده سازی چت روم با استفاده از $socket\ programming$ است و محیط آن باید به صورت کاملا گرافیکی باشد.
برنامه شما باید یک سرور و چند کلاینت داشته باشد که هرکدام به سرور متصل میشوند و از طریق آن با هم ارتباط برقرار میکنند.
(برای مثال شما باید یک کد بزنید و چندبار آنرا اجرا کنید، به طوری که بار اولی که اجرا میشود سرور بسازد و دفعه های بعدی کلاینت بسازد)

- هر کاربر باید نام خود را در هنگام ورود وارد کند.
- یک صفحه گفتگو ها در برنامه باشد که با کلیک بر روی نام هر کاربر، وارد صفحه چت با آن کاربر شویم.
- کاربرها باید قابلیت ارسال پیام متنی، اموجی و یا عکس به یکدیگر، و همچنین قابلیت ریپلای کردن پیام ها را داشته باشند.
- یک کاربر باید بتواند گروه بسازد و با وارد کردن نام کاربر های دیگر آنهارا در هر زمان به گروه اضافه کند (هم در هنگام تشکیل گروه و هم بعد از آن. فقط سازنده گروه میتواند کاربر ها را به گروه اضافه کند). نام گروه هم برای اعضای هر گروه، به صفحه گفتگو های آن کاربر اضافه میشود و با کلیک بر روی آن میتوانند وارد صفحه چت آن گروه شوند

نکته : برای ارسال عکس میتوانید چند فایل عکس در کنار برنامه خود قرار دهید و برای مثال با نوشتن $send\ example.jpg$ آنرا ارسال کنید. یکی از راه های ارسال عکس از طریق $socket$ ، خواندن بایت به بایت فایل عکس و سپس ارسال و بازسازی آن است.
اگر از هر روش دیگری هم میخواهید استفاده کنید باید عکس را به طور کامل بفرستید، برای مثال فرستادن تنها نام عکس و لود کردن آن از یک فایل کنار برنامه نمره ای نخواهد داشت.

دوز کذایی II (۴۰ + ۱۰ امتیاز)

در این سوال باید از کد خودتان در سوال دوز کذایی در تمرین دوم استفاده کنید. با بازی دوز در تمرین دو آشنایید. در این سوال قرار است بازی را به صورت $multi\ player$ و با استفاده از $socket\ programming$ پیاده سازی کنید.
برنامه شما باید یک سرور و چند کلاینت داشته باشد که هرکدام به سرور متصل میشوند و از طریق آن با هم ارتباط برقرار میکنند.
(برای مثال شما باید یک کد بزنید و چندبار آنرا اجرا کنید، به طوری که بار اولی که اجرا میشود سرور بسازد و دفعه های بعدی کلاینت بسازد)
مواردی که علاوه بر کد قبلی باید پیاده سازی شوند به شرح زیرند:

- در ابتدای اجرای برنامه کاربر باید نام خود را وارد کند. نام های کاربری نباید تکراری باشند.
- برای بازی کردن دو کاربر با یکدیگر، یکی از آنها دستور$new\ game\ playerName$ را وارد میکند و درخواست را به سرور میفرستد. در صورتی که کاربر وجود نداشت یا در حال بازی بود پیغام خطا به بازیکن اول ارسال میشود. در غیر این صورت دو بازیکن وارد بازی میشوند. دقت کنید که ممکن است همزمان چند بازی در حال انجام باشند. دستور $resume$ هم که در منوی مخصوص به خود وارد میشود، به همین شیوه کار میکند
- با وارد شدن به یک بازی، محیط گرافیکی دوز برای بازیکنان نشان داده شود که بتوانند جدول را پس از هر حرکت در آن ببینند. دقت کنید که نیازی به محیط گرافیکی برای باقی بخش ها نیست و حرکات بازیکن ها در بازی (دستورput) هم میتواند از طریق همان ورودی متنی باشد.
- با وارد کردن دستور $pause$ توسط یکی از بازیکن ها بازی متوقف میشود.

بقیه دستورات به صورت کاملا مشابه با سوال قبلی پیاده سازی میشوند.

موارد امتیازی (حداکثر ۱۰ نمره)

- پیاده سازی کامل محیط برنامه به صورت گرافیکی.
- قابلیت ارسال پیام بین دو بازیکن در هر حال بازی.

تردمیل (۳۰ امتیاز)

هدف این سوال آشنایی با $multi\,threading$ در جاوا است.
شما باید برنامه ای پیاده سازی کنید که ۳ نویسنده و ۳ خواننده داشته باشد که هرکدام یک $thread$ اند. هرکدام از نویسندگان در فایل متنی $data.txt$ که در کنار برنامه تان قرار داده شده مینویسند و هرکدام از خوانندگان هم اطلاعات این فایل را میخوانند. سه نویسنده داریم. این نویسندگان با هم کار میکنند و **هر ثانیه ۵ کاراکتر** در فایل مینویسند (لازم نیست دقیق باشد، صرفا به صورت میانگین این سه نویسنده روی هم این تعداد کاراکتر را در هر ثانیه بنویسند). نویسنده اول کاراکتری تصادفی از حروف الفبا (A تا Z) ، نویسنده دوم عددی تصادفی از ۱ تا ۱۰۰ و نویسنده سوم یکی از علامت های !،@،# ،$ ، % ، & و یا * را مینویسند.
خواننده ها نیز با هم همکاری کرده و نوشته های درون فایل را میخوانند و خواننده اول اطلاعاتی که میخواند را در فایل $1.txt$ ، نویسنده دوم در فایل $2.txt$ و نویسنده سوم در فایل $3.txt$ قرار میدهند. همچنین خواننده ها نباید اطلاعات فایل $data.txt$ را تغییر دهند. در صورتی که متنی برای خواندن وجود نداشت (یعنی نویسنده ها از خواننده ها عقب افتادند)، خواننده باید صبر کند (با استفاده از متد $wait$) تا نویسنده ها چیزی در فایل بنویسند و سپس خواننده ها را مطلع سازند (با استفاده از متد $notify$).