**به نام خدا**

|  |  |  |  |
| --- | --- | --- | --- |
| سری نهم تمرینات درس برنامه‌نویسی جاوا | ترم دوم سال تحصیلی 97-96 | مدرس: مظفر بگ محمدی | دانشگاه ایلام |

**حل این سری کاملاً اختیاری بوده اما در صورت حل به عنوان نمره اضافه محاسبه می‏شود، سطح سؤالات این سری ساده است.**

1. **Regex: آدرس سایت**

برنامه‌ای بنویسید که با گرفتن یک‌رشته از کاربر مشخص کند آیا رشته ورودی به فرمت صحیح آدرس یک سایت است یا خیر. برای نوشتن این برنامه باید از پکیج regex جاوا استفاده کنید.

شرایط آدرس یک سایت:

* می‌تواند پیشوند http:// یا https:// یا ftp:// و ... داشته یا نداشته باشد.
* می‌تواند پیشوند www داشته یا نداشته باشد.
* حتماً باید از دو قسمت نام و دامنه تشکیل‌شده باشد که با یک نقطه از هم جداشده باشند.
* کاراکترهایی که نام سایت می‌پذیرد حروف انگلیسی و ارقام هستند.
* حرف اول سایت نمی‌تواند رقم باشد.
* کاراکترهایی که دامنه سایت می‌پذیرد فقط حروف انگلیسی هستند.
* حتماً باید یک دامنه با طول حداقل دو داشته باشد.

ورودی به‌صورت تنها یک‌رشته است و اگر رشته واردشده شرایط آدرس یک سایت را داشته باشد، خروجی true و در غیر این صورت false است.

****

1. **Regex: استخراج‌کننده آدرس**

برنامه‌ای بنویسید که با دریافت کامل یک فایل html در ورودی، تمام لینک‌های هدر h3 در آن را استخراج کند و در خروجی چاپ کند. (هر لینک در یک خط چاپ شود)

* برای دریافت کامل فایل html و ریختن آن در یک‌رشته به‌صورت زیر عمل کنید:

1. String input = "";
2. java.util.Scanner scanner = new java.util.Scanner(System.in);
3. while (scanner.hasNext())
4. input += scanner.nextLine() + "\n";

* فرمت لینک‌های هدر h3:

1. < h3 class = "r"> < a href = "https://www.google.com//"...

ورودی نمونه

1. https://www.google.com/search?ei=BiGUWvCMJM\_fwALqrZKICQ&q=java&oq=java&gs\_l=psy-ab.3..0i67k1l9j0.380940.382000.0.382206.7.6.1.0.0.0.173.639.0j4.4.0....0...1c.1.64.psy-ab..2.5.644....0.ESkDyJdiVUg

خروجی نمونه

1. https://java.com/
2. https://www.java.com/
3. https://docs.oracle.com/javase/tutorial/java/nutsandbolts/
4. https://en.wikipedia.org/wiki/
5. https://google.github.io/styleguide/
6. https://www.javabowlofsoul.com/
7. **Graphic: پونگ**

در این سؤال شما باید بازی پونگ را به‌صورت گرافیکی پیاده کنید.

1. https://pong-2.com/

بازی به‌صورت دو نفرِ انجام می‌شود. بازی شامل یک زمین‌بازی، یک توپ و دو تخته است. زمین‌بازی به شکل یک مستطیل است که بالا و پایین آن دیوار و سمت چپ و راست آن دو دروازه است. دو تخته به‌صورت عمودی در طرفین زمین قرار دارد. هر یک از بازیکن‌ها می‌تواند به‌وسیله‌ی دو کلید روی صفحه‌کلید (!) تخته خود را بالا و پایین کند و مانع ورود توپ به دروازه خود شود (مثلاً بازیکن سمت راست به‌وسیله‌ی I و K و بازیکن سمت چپ به‌وسیله‌ی W و S).

بازی تا زمانی ادامه پیدا می‌کند که اولین بازیکن 5 گل به ثمر برساند. توپ در شروع بازی یا شروع مجدد بازی در مرکز زمین قرار می گیرد و به‏وسیله‏ی یک زاویه‌ی تصادفی به سمت یکی از بازیکن‌ها پرتاب می‌شود. بازی دارای یک زمان سنج است که زمان مفید بازی را نشان می‌دهد. سرعت توپ با افزایش زمان باید به‌صورت خطی افزایش یابد.

برنامه دارای تعدادی بازیکن است. هر بازیکن به‌وسیله‌ی نامش به‌صورت یکتا مشخص می‌شود. برای شروع بازی باید نام بازیکنان و نام بازی را وارد کرد. با ورود به برنامه، منوی اصلی را می‌بینیم. این منو باید شامل گزینه‌های زیر باشد:

* **اضافه کردن بازیکن:** با انتخاب این گزینه باید نام بازیکن وارد شود و در صورت معتبر بودن بازیکن اضافه می‌شود.
* **شروع کردن یک بازی:** با انتخاب این گزینه نام بازیکن‌ها و بازی وارد می‌شود و اگر نام‌ها معتبر بودند بازی شروع می‌شود.
* **چاپ بازیکن‌های برتر:** نام بازیکن‌ها را بر اساس تعداد بردها نشان می‌دهد.
* **خروج**

همچنین در خود صفحه بازی باید موارد زیر نمایش داده شود:

* **زمین بازی**
* **امتیاز هر بازیکن**
* **زمان بازی**
* **دکمه توقف و ادامه بازی**

همین‌طور می‌توانید موارد زیر را به‌صورت امتیازی پیاده‌سازی کنید:

* امکان توقف و ذخیره کردن یک بازی در فایل و شروع مجدد آن با داشتن یک لیست از بازی‌های ذخیره‌شده.
* استفاده از عناصر مختلف برای زیباسازی منوها و صفحه بازی
* روان بودن و طراحی خوب بازی
* فرض کنیم توپ دارد در یک‌جهت خاصی حرکت می‌کند و کاربر می‌خواهد با فشردن یک کلید رنگ توپ را به رنگ خودش دربیاورد و هر بار که کاربری امتیاز می‌گیرد توپ به رنگ او دربیاید. حال منویی باید داشته باشیم که کاربر بتواند رنگ مخصوص خودش را انتخاب کند.
* قسمتی طراحی کنید که بتوان به بازی آهنگ اضافه، حذف و قطع و وصل کرد، مثلاً در قسمت تنظیمات.
* آیکون‌هایی به‌صورت تصادفی در صفحه ظاهر می‌شوند که در صورت برخورد توپ با آن‌ها توپ خواصی را به خود می‌گیرد؛ مانند این آیتم:
  + دو برابر کردن سرعت توپ
  + نامرئی شدن توپ به‌صورت تناوبی
  + کوچک شدن توپ
  + تغییر جهت توپ به‌صورت تصادفی

هرکدام از این موارد مدت زمان معینی دارند که پس از گذشت آن توپ خواص خود را از دست می‌دهد و به حالت اولیه برمی‌گردد. تمام آیکون‌ها هم مدتی روی صفحه باقی می‌مانند و اگر توپ به آنان برخورد نکرد، از بین می‌روند.

همچنین درباره پیاده‌سازی موارد امتیازی به موارد زیر دقت کنید:

درصورتی‌که موارد امتیازی دیگری به ذهنتان رسید به اطلاع حل تمرین خود برسانید تا درصورت تائید، نمره ی اضافه دریافت کنید.

**دقت کنید** حداکثر نمره امتیازی این تمرین 15 امتیاز است، لذا در پیاده‌سازی موارد امتیازی افراط نکنید.

1. **Socket Programming: پونگ دونفره!**

در این سؤال شما باید با استفاده از مفاهیم مربوط به شبکه، یک سرور بازی برای سؤال قبلی طراحی کنید یا به‌اصطلاح این سؤال را به یک بازی client-server تبدیل کنید.

وظیفه‌ی این سرور به این صورت است که دو کلاینت به آن متصل می‌شوند و سپس از طریق آن می‌توانند با هم بازی کنند. به‌نوعی در این سؤال شما باید قسمت multi-player بازی سؤال سوم را به‌صورت آنلاین پیاده‏سازی کنید.

نحوه پیاده‌سازی این موضوع می‌تواند به این شکل باشد که شما در منوی بازی سؤال پیش، یک قسمت multiplayer اضافه کنید که پس از ورود به آن بازیکن انتخاب می‌کند که server باشد یا client.

* درصورتی‌که کاربر گزینه server را انتخاب کند، بایستی منتظر بماند تا دو client به او متصل شوند و بازی بین آن‌ها را شروع کند.

سرور باید اطلاعات لازمی که در سؤال قبلی مشخص‌شده بودند را از دو کاربر دریافت و ذخیره کند. همچنین باید بتوان بر روی سرور میزان امتیاز هدف برای برنده شدن در بازی را مشخص کرد. سایر بخش‌های اجباری که بازی در این حالت باید داشته باشد دقیقاً مطابق با بخش‌های اجباری سؤال قبلی هستند.

* درصورتی‌که کاربر گزینه client را انتخاب کند، بایستی منویی برای او بیاید که نام خود، IP و Port سروری که می‌خواهد به آن وصل شود را در آن وارد کند تا بتواند با کاربر دیگری که به آن سرور متصل شده بازی کند. همین‌طور دقت کنید که اگر می‌خواهید هر دو client را روی یک دستگاه اجرا کنید، بایستی یا در کد خود این تدبیر را بیندیشید که دو client با دو Port مختلف به server وصل شوند یا این‌که در منویی که برای کاربر client می‌آید فیلدی وجود داشته باشد که کاربر خودش Port ی که می‌خواهد از طریق آن بازی کند را انتخاب کند.

همچنین می‌توانید موارد زیر را به‌صورت امتیازی پیاده‌سازی کنید:

* هر کاربر درخواست بازی خود را به سرور بفرستد و سپس سرور لیستی از افراد درخواست‌کننده بازی را به او نشان دهد تا از بین آن‌ها انتخاب کند.
* نگه‏داشتن لیستی از بازیکن‌هایی که تاکنون بازی کرده‏اند به همراه نتایج بازی‌های آن‌ها و leaderboard بازی.
* پیاده‌سازی شبکه به صورتی که یک کاربر بتواند هم‌زمان server و client باشد، یعنی لازم نباشد که یک نسخه از بازی به‌صورت server اجرا شود و نتواند کار دیگری بکند، بلکه دو نسخه از بازی بتوانند باهم بازی کنند به شکلی که یک نفر client باشد و یک نفر دیگر هم server و client.

درصورتی‌که موارد امتیازی دیگری به ذهنتان رسید به اطلاع حل تمرین خود برسانید تا درصورتی‌که مورد تائید بود نمره اضافه دریافت کنید. همچنیندقت کنید که مانند سؤال قبل در پیاده‏سازی موارد امتیازی افراط نکنید، نمره امتیازی سؤال نهایتاً می‌تواند به‌اندازه‌ای که ذکرشده (حداکثر 15 نمره) باشد.

1. **Multithreading: آجیل‌فروشی احمد‏رضا و برادران به‌جز حمیدرضا**

احمدرضا و برادرانش چون خیلی فکر اقتصادی خوبی دارند تصمیم گرفتند دانشگاه را رها کنند و باهم یک آجیل‌فروشی باز کنند تا بتوانند خرج زندگی خود را دربیاورند و به ریش هر مهندسی که وارد مغازه اشان می‌شود بخندند (البته حمیدرضا به ادامه تحصیل ادامه داد برای همین بالای آجیل‌فروشی‌شان نوشتند " آجیل‌فروشی احمدرضا و برادران به‌جز حمیدرضا"). قرار شد هر کس فروش بخشی از آجیل‌ها را بر عهده بگیرد و درآمد بخش خود را به داخل یک حساب مشترک بریزد. ازآنجایی که همه کارهای احمدرضا حساب‌وکتاب دارد، تصمیم گرفته برنامه‌ای بنویسد تا تغییرات مالی حساب آجیل‌فروشی را داشته باشد. در اول تعداد برادران (n) از کاربر گرفته می‌شود و بعد تعداد تغییرات مالی هر روز.

هر تغییر مالی به این شکل وارد می‌شود:

1. 1 1000

به این معنا که برادر اول 1000 تومن به‌حساب آجیل‌فروشی اضافه کرده است. بعد از گرفتن همه دستورات، n نخ شروع به کار می‌کنند که هر نخ مربوط به یک برادر است و نخ شماره، i تغییرات مالی مربوط به برادر i را انجام می‌دهد. همچنین بازهم ازآنجایی‌که همه کارهای احمدرضا حساب‌وکتاب دارد، معتقد است که نخها باید "سِیف" باشند (چون خودش تمرین نخی که برای برنامه‌نویسی پیشرفته زده بود "سِیف" نبود، TA های نوکیسه نمره‌اش را نداده بودند). بعد از اجرای هر دستور شماره برادر، مقدار تغییر حساب و پول کل حساب را چاپ کنید تا احمدرضا ببیند که هرکدام از برادرانش چقدر درآمدزایی دارند.

1. **Generic: امر خیر**

پادشاه گوگولستان دختر زیبایی به نام فرنوش داشت که همه‌ی جوانان شهر آرزوی ازدواج با او را در سر می‌پروراندند. روزها یکی پس از دیگری سپری می‌شد و عده‌ی هواخواهان فرنوش هر روز بیش‏تر و بیشتر می‌شد و تصمیم‌گیری برای او سخت‏تر و سخت‌تر. تا این‌که ماجرایی رخ داد که سبب آشنایی فرنوش را با عده‏ای فراهم کرد که می‌توانستند مشکل او را حل‌وفصل نمایند. خوشبختانه او توانسته بود شما را که همواره مشتاق همیاری و کمک در انجام امور خیر هستید، پیدا کند! پس بیایید برای خوشبخت شدن هرچه بیشتر فرنوش بکوشید و دستان پرتوان خود را به‌کارگیرید تا شاید روزی کسی این کار نیک را در حق شما جبران نماید چنان‌که گفته‌اند: "تو نیکی کن و در دجله انداز که ایزد در بیابانت دهد باز". برای آن‌که در این امر ما نیز سهمی داشته باشید و از برکات این امر خیر ذره‌ای چشیده باشیم چگونگی این کمک را تا حدی شرح می‌دهیم، "باشد که مقبول افتد ": در گام اول اطلاعات خواستگاران فرنوش را در یک داده ساختار ذخیره کنید. مطابق علایق شخصی فرنوش، این داده ساختار یک "درخت جست‌وجوی دودویی" است. از آنجا که معیارهای فرنوش در انتخاب همسرش برای ما به‌طور دقیق مشخص نیست این داده ساختار باید به‌صورت Generic تعریف شود. برای مثال اگر قلی خواستگار فرنوش باشد می‌تواند ویژگی‌هایی همچون قد، وزن، رنگ چشم و ... داشته باشد و برنامه شما باید آن‌ها را تا حد خوبی پشتیبانی نماید. نام داده ساختار و تعریف آن این‌گونه است:

1. Public class BST < T extends Comparable > {}

عملیاتی که این کلاس باید بتواند انجام دهد به‌صورت توابعی به همراه تعریف هریک، در زیر آورده شده است:

* این متد تعداد عناصر موجود در داده ساختار را نمایش می‌دهد:

1. public int size();

* باید بتوان عنصری جدید را به این داده ساختار افزود:

1. Public void add(T element);

* این متد عنصری از جنس T گرفته و اگر آن عنصر داخل داده ساختار موجود بود true و در غیر این صورت false برمی‌گرداند:

1. Public T containts(T element);

* کوچک‌ترین عنصر موجود در داده ساختار را برمی‌گرداند (در درخت جستجوی دودویی این عنصر همان چپ‌ترین برگ است):

1. Public T getMinElement() throws IllegalStateException;

* بزرگ‌ترین عنصر موجود در درخت که همان راست‌ترین برگ را برمی‌گرداند:

1. Public T getMaxElement() throws IllegalStateException;

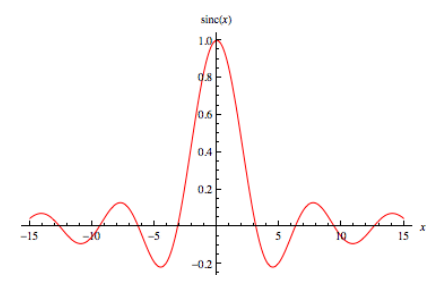
در پیاده‌سازی این داده ساختار به نکات زیر توجه کنید:

* در توابع getMaxElement و getMinElement اگر هیچ عنصری وجود نداشته باشد باید یک استثنا از نوع IllegalStateException پرتاب شود.
* پیاده‌سازی بدنه‌ی متدها اختیاری بوده و بر عهده‌ی خودتان است.
* برای آن‌که از درستی تست‌های نوشته‌شده مطمئن شوید باید تعدادی تست در قالب Unit Test (حداقل 2 تست برای هر متد) طراحی کنید. سعی کنید تست‌های جامع و کاملی طراحی کنید چراکه در نمره دهی شما مؤثر است.

1. **MultiThreading: ریسه بازی**

در این سؤال می‌خواهیم انتگرال تابع sinc را با استفاده از روش چندریسگی به دست آوریم.

*این تابع شکلی مانند زیر دارد:*



*برای این کار می‌خواهیم از سه نخ استفاده کنیم:*

*یک نخ باید نقاط صفر شدن تابع را پیدا کند، نخ دوم وظیفه‏ی حساب کردن انتگرال در قسمت‌هایی را بر عهده دارد که مقدار تابع مثبت است و نخ سوم قسمت‌های منفی را حساب می‌کند.*

*دقت کنید که هر سه نخ کار خود را هم‌زمان پیش می‌برند و نباید تا حد امکان وقتی تلف کنند.*

*ابتدا نخ اول شروع می‌کند و نخهای دوم و سوم هیچ کاری انجام نمی‌دهند. پس از مدتی یک بازه برای نخ دوم به دست می‌آید و در این لحظه هر دو نخ اول و دوم به کار خود ادامه می‌دهند. پس از مدتی اولین بازه برای نخ سوم نیز پیدا می‌شود و در این لحظه نخ سوم شروع به کار می‌کند. نخ اول همچنان در حال پیدا کردن بازه‏ی بعدی برای نخ دوم است، اگر تا زمان پیدا شدن این بازه کار نخ دوم تمام شود، نخ دوم باید صبر کند و منتظر نخ اول باشد تا بازه‏ی بعدی را پیدا کند، در غیر این صورت اگر نخ اول بازه‏ی بعدی را پیداکرده باشد این نخ بلافاصله به سراغ محاسبه بازی بعدی می‌رود.*

*نخ اول بعد از پیدا کردن هر بازه، آن بازه را به همراه زمان پیدا شدن آن بازه چاپ می‌کند.*

[ta, tb][a, b] + / -

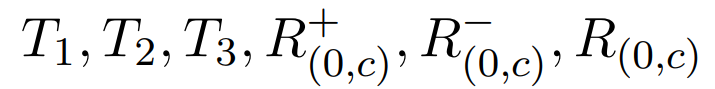
*که در آن tx زمان پیدا شدن x ، a ابتدای بازه و b انتهای بازه و + یا – نشان‏دهنده‏‏‏‏‏‏ای این است که مقدار تابع در این بازه مثبت یا منفی است.*

*نخهای دوم و سوم هم پس از محاسبه‌ی هر بازه مقدار جمع و زمان به پایان رسیدن محاسبه‏ی آن بازه را چاپ می‌کند.*

[ta, tb][a, b]R (a, b)

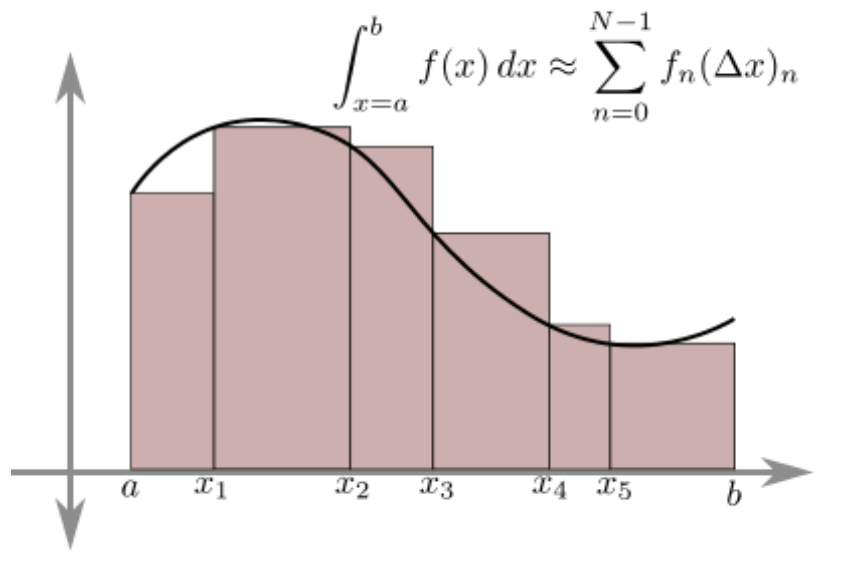
*که در آن R(a,b)  مقدار جمع تابع در این بازه است.*

*در انتها پس از پایان یافتن محاسبات هر سه نخ مقدار مثبت و منفی را جداگانه نمایش دهید و حاصل کل را نیز محاسبه کنید. همین‌طور زمان مفید هر نخ (زمانی که مشغول محاسبه بوده است) را نیز چاپ کنید.*



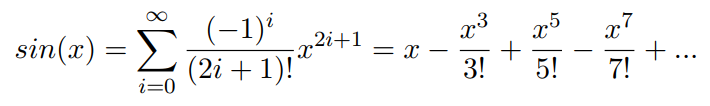
*که در این رابطه Ti زمان کل نخ i است.*

*برای محاسبه انتگرال از روش جمع ریمان استفاده می‌کنیم. در این روش بازه‌ی موردنظر را به مستطیل‌هایی تقسیم می‌کنیم و مساحت هر مستطیل را حساب می‌کنیم و باهم جمع می‌کنیم.*



*نکات زیر را در نظر داشته باشید:*

* *عرض هر مستطیل را یکسان در نظر بگیرید و مقدار 0.01 یا کمتر را برای آن انتخاب کنید.*
* *در نظر داشته باشید به‌محض این‌که یک بازه‌ی قابل انتگرال‌گیری (مثبت یا منفی) توسط نخ اول مشخص شد نخهای دیگر در همان لحظه باید شروع به محاسبه بکنند.*
* *نحوه‌ی کار نخ اول این‌طور است که از 0 شروع به پیمایش محور x می‌کند تا جایی که تغییر علامتی متوجه شد آن نقطه را به‌عنوان نقطه‌ی 0 شدن تابع گزارش می‌کند. مقدار طی شده در هر پیمایش را هم برابر عرض مستطیل‌ها در نظر بگیرید.*
* *دقت کنید که حساب کردن تابع sinc در اطراف 0 کمی مشکل‌دار است برای دوری از این مشکل می‌توانید تابع sinc را در نزدیکی‌های 0 با بسط تیلور آن تقریب بزنید. حساب کردن این «بازه‏ی مشکل دار» و دقت تقریب با شماست. (می‌توانید محاسبات این قسمت را در کاغذ انجام دهید ولی بهتر است که در صورت امکان برنامه شما خودش بتواند آن‌ها را حساب کند).*



* *می‌توانید از تابع‌های Sin و Cos که در کتابخانه‌های محاسباتی جاوا موجود است استفاده کنید.*
* *هر چه جواب انتگرال شما به واقعیت نزدیک‌تر باشد نمره امتیازی بیشتری خواهید گرفت.*