



## ۱ جمعه بازار

شما باید فروشگاه اینترنتی **جمعه بازار** را طراحی کنید که در آن سه نقش متفاوت وجود دارد. اولین نقش، مدیر فروشگاه است که می‌تواند کالاهایی را که می‌خواهد در فروشگاهش به فروش برسد، از یک فایل وارد برنامه کند. نقش دوم فروشندگان هستند که می‌توانند روی کالاهایی که توسط مدیر فروشگاه قرارداده شده‌است، تعداد و قیمت واحد پیشنهادی خود را ثبت کنند. دسته سوم خریداران هستند که پس از ثبت نام می‌توانند لیست کالاها را ببینند و سپس از بین پیشنهادهای یک کالا، از یکی خرید انجام دهند.

این پروژه به جمع‌بندی آموخته‌های شما در درس می‌پردازد. انتظار می‌رود مهارت‌هایی را که در تمرین‌های پیشین و سایر بخش‌های درس آموخته‌اید، در پیاده‌سازی این پروژه به کار گیرید. در فاز اول شما منطق برنامه‌ی این فروشگاه را پیاده‌سازی می‌کنید که با استفاده از رابط کاربری خط فرمان<sup>۱</sup> می‌توان از آن استفاده کرد. در ابتدای اجرا برنامه، مدیر فروشگاه به طور خودکار وارد می‌شود. دستوراتی که در ابتدا داخل کنسول وارد شوند، از طرف مدیریت فروشگاه هستند. به این ترتیب، مدیر فروشگاه فایل‌های اطلاعات کالاهایی را که می‌خواهد در فروشگاهش قرار دهد، وارد می‌کند. این فایل‌ها سه فایل متفاوت برای سه دسته‌بندی گوشی موبایل، تلویزیون و خودرو به صورت مجزا وارد خواهند شد. پس از خروج مدیر از فروشگاه، فروشندگان می‌توانند در سایت ثبت نام کنند و سپس روی کالاهایی که مدیر قبلاً وارد کرده است، پیشنهاد فروش خود را با وارد کردن تعداد و قیمت واحدی که آن کالا را ارائه می‌کنند، بگذارند. و پس از آن خریداران می‌توانند با ثبت نام در فروشگاه، ابتدا حساب مالی خود را شارژ کنند، و از میان پیشنهادات فروشندگان، کالایی را به سبد خریدشان اضافه کنند. خریداران در آخر می‌توانند با استفاده از اعتبار مالی خود، کل سبد خرید خود را نهایی کنند که به این ترتیب کل هزینه آن از اعتبار خریدار به فروشندگان هر کالا منتقل می‌شود. هر نوع از کاربران برای استفاده از این فروشگاه دستورات خاصی دارند که در ادامه به تفصیل توضیح داده خواهند شد.

## ۲ شرح تمرین

در این فاز، منطق برنامه در قالب تعدادی دستور در ادامه توضیح داده شده‌است. تمام دستورها پس از اجرا شدن دارای خروجی مشخص هستند که منحصراً ذکر می‌شود.

دقت کنید که خروجی‌های شما به صورت خودکار تست می‌شوند. لذا خروجی شما باید دقیقاً همانند خروجی خواسته شده باشد. در غیر این صورت، نمره‌ی بخش آزمون<sup>۲</sup> را از دست خواهید داد.

## ۳ کالاها

هر کالا دارای نام و شناسه‌ای یکتا به همراه مقادیر ویژگی‌های خاصی است که دسته اش دارد. مدیر فروشگاه، کالاها را در ۳ فایل متفاوت که در کنار پروژه داده شده‌است، وارد می‌کند و همچنین کاربران می‌توانند کالاهای وارد شده را طبق این ویژگی‌ها با هم مقایسه کنند. این دستورات در ادامه توضیح داده خواهند شد. کالاهای داخل سایت ۳ دسته بندی مختلف دارند.

<sup>۱</sup>CLI(Command Line Interface)

<sup>۲</sup>Test

### ۱.۳ گوشی موبایل

ویژگی‌های این دسته عبارتند از:

- وزن
- فرکانس پردازنده مرکزی
- حافظه داخلی
- مقدار RAM
- اندازه صفحه نمایش
- رزولوشن دوربین
- نوع سیستم عامل

### ۲.۳ تلویزیون

ویژگی‌های این دسته عبارتند از:

- سایز صفحه
- تکنولوژی صفحه
- کیفیت تصویر
- سه بعدی
- HDR

### ۳.۳ خودرو

ویژگی‌های این دسته عبارتند از:

- وزن
- تعداد سرنشین
- تعداد سیلندر
- حجم موتور
- سنسور دنده عقب

**توجه:** تضمین می‌شود که ویژگی‌های دودویی فقط یکی از دو مقدار ۰ یا ۱ را داشته‌باشند و هر کدام از ویژگی‌های عددی حتماً عدد باشند.

## ۴ دستورات

### ۱.۴ احراز هویت

#### ۱.۱.۴ ثبت نام

با این دستور افراد می‌توانند در فروشگاه ثبت نام کنند. ایمیل هر کاربری یک رشته یکتا در سیستم است. توجه کنید که هر کاربر دارای یک شناسه‌ی عددی یکتا است که این شناسه از عدد یک شروع می‌شود. این شناسه به طور خودکار در هنگام ثبت نام کاربر به آن تعلق می‌گیرد. اگر نام کاربری در سیستم موجود باشد، درخواست معتبر نیست و خروجی Bad Request چاپ می‌شود. وجود ویژگی نوع کاربر اختیاری است و در صورت ذکر نشدن خریدار فرض می‌شود. توجه کنید که مدیر فروشگاه نیازی به ثبت نام ندارد و اولین کاربری که برنامه را باز کند، به عنوان مدیر وارد می‌شود.

لازم است برای مدیر خودتان در کد برنامه یک نام کاربری و رمز عبور مشخص برای ورودهای بعدی در نظر بگیرید.

ورودی	خروجی
POST signup ? email <email> username <username> password <password> [type <seller   buyer>]	OK   Bad Request

#### ۲.۱.۴ ورود

اگر فردی قبلاً در فروشگاه ثبت نام کرده باشد، پیش از استفاده از امکانات سامانه باید در آن وارد شود. توجه کنید که کاربران بلافاصله بعد از ثبت نام نیازی به دستور ورود ندارند و پس از ثبت نام به صورت ضمنی با همان دستور ثبت نام وارد سیستم نیز شده‌اند. در صورت درست نبودن اطلاعاتی که کاربر وارد کرده است یا کاربری در سیستم وجود دارد، پیام Bad Request در خروجی چاپ شود.

ورودی	خروجی
POST login ? email <email> password <password>	OK   Bad Request

#### ۳.۱.۴ خروج

اگر شخصی قبلاً در سیستم وارد شده بود، با وارد کردن این دستور از سیستم خارج می‌شود و می‌تواند با یک حساب کاربری دیگر وارد سامانه شود. اگر کاربری قبل از ورود به سیستم این دستور را وارد کرد، پیام Permission Denied باید خروجی داده شود.

ورودی	خروجی
POST logout	OK   Permission Denied

### ۲.۴ مدیر فروشگاه

دستورات قبل به منظور ورود کاربران به سامانه در نظر گرفته شده بود. هر نوع از کاربران قابلیت های خاص خود را دارند که در ادامه قابلیت های هر یک در قالب دستوراتی بیان می‌شود. اگر کاربری به سامانه وارد نشده باشد یا دسترسی را صدا بزند که مربوط به نوع کاربری او نیست، باید پیام خطای Permission Denied داده شود.

## ۱.۲.۴ وارد کردن کالاها

مدیر فروشگاه می‌تواند اطلاعات کالاها را به صورت یک‌جا از یک فایل وارد سامانه کند. آدرس فایل به صورت نسبی<sup>۳</sup> داده می‌شود. نمونه‌ی فایل ورودی در کنار پروژه در اختیاران قرار گرفته است. به ازای هر دسته کالا، یک فایل وارد خواهد شد. در هر فایل ویژگی‌های کالاهای آن دسته در هر سطر نوشته شده‌اند و ستون‌ها نشان‌گر ویژگی کالای موردنظر هستند. اگر در آدرس داده شده، فایلی وجود نداشت، باید خطای Not Found داده شود.

ورودی	خروجی
POST importProducts ? type <mobile   tv   car> filePath <filePath>	OK   Not Found

## ۲.۲.۴ مشاهده پیشنهادات فروشنده‌ها

مدیر باید بتواند تمام پیشنهادات گذاشته شده را به ترتیب مشخصی مشاهده کند. ویژگی order مشخص‌کننده ترتیب خروجی است. ASCEND به معنای صعودی بودن و DESCEND به معنای نزولی بودن براساس آن ویژگی است. Field نیز بیانگر ویژگی است که این ترتیب براساس آن صورت می‌گیرد.

در حالتی که دو مقدار قیمت برابر باشند، اول پیشنهادی می‌آید که شناسه پیشنهادش کوچکتر است.

ورودی	خروجی
GET offers ? order <ASCEND/DESCEND> field <offerId/offerPrice>	<productId>   <offerId>   <offerUnitPrice>   <offerAmount>
GET offers ? order ASCEND field offerId	productId   offerId   offerUnitPrice   offerAmount 5   1   250.99   10 11   2   23.99   23

## ۳.۴ فروشنده

هر فروشنده می‌تواند روی کالاهایی که مدیر وارد کرده است، تعداد موجود و قیمت واحد پیشنهادی خود که با آن کالا را ارائه می‌دهد بگذارد. با خرید کاربران از این فروشنده، هزینه کالاها به حساب مالی فروشنده افزوده و از حساب مالی خریدار کاسته می‌شود. همچنین پس از خرید موفق از تعداد کالاهای موجود در این پیشنهاد کاسته می‌شود.

## ۱.۳.۴ مشاهده کالاهای فروشگاهها

در پاسخ این دستور باید کالاهای موجود در فروشگاه که توسط مدیر وارد شده است، به ترتیب شماره id آنها چاپ شوند.

ورودی	خروجی
GET getProducts	<productId>   <productName>
GET getProducts	productId   productName 1   iPhone XS Max 3   Galaxy Note 10 4   Samsung 65" 4K UHD

<sup>3</sup>relative

#### ۲.۳.۴ ثبت تعداد و قیمت واحد پیشنهادی

هر فروشنده می‌تواند روی کالاهایی که مدیر وارد کرده‌است، تعداد و قیمت واحد پیشنهادی خود برای فروش را بگذارد. اگر شناسه کالا وجود نداشته باشد، یا فروشنده قبلاً روی این کالا پیشنهاد گذاشته باشد، باید خطای Bad Request داده شود. برای تغییر پیشنهاد گذاشته شده، دستور خاصی در ادامه توضیح داده خواهد شد. دقت کنید که به ازای هر offer نیز یک شناسه عددی یکتا وجود دارد.

ورودی	خروجی
POST offer ? productId <productId> offerUnitPrice <offerUnitPrice> offerAmount <offerAmount>	OK

#### ۳.۳.۴ مشاهده پیشنهادات گذاشته شده توسط خود فروشنده

پیشنهادهای گذاشته شده توسط فروشنده بر اساس شناسه یکتای پیشنهاد خروجی داده شود.

ورودی	خروجی
GET myOffers	productId   offerId   offerUnitPrice   offerAmount 5   1   250.99   10 11   2   23.99   23

#### ۴.۳.۴ تغییر پیشنهادهای گذاشته شده

فروشنده می‌تواند پیشنهادی که قبلاً روی یک کالا گذاشته را تغییر دهد. اگر شناسه پیشنهاد وجود نداشته باشد، باید خطای Bad Request داده شود.

ورودی	خروجی
POST changeOffer ? offerId <offerId> offerUnitPrice <offerUnitPrice> offerAmount <offerAmount>	OK

#### ۵.۳.۴ مشاهده گردش‌های مالی

فروشنده می‌تواند چند گردش آخر حساب مالی خود را ببیند. مقدار count مشخص کننده تعداد این گردش‌هاست. توجه کنید که در ابتدای کار، داخل حساب مالی هر کاربری مقدار ۰ قرار دارد. خط اول، موجودی کنونی کاربر است. اگر تعداد داده شده در ورودی برای مشاهده، بیشتر از تعداد گردش‌های کاربر بود، به همان تعدادی که گردش وجود دارد، موجودی چاپ می‌شود.

ورودی	خروجی
GET wallet ? count <count>	credit 22 20 10 0

#### ۶.۳.۴ ایجاد کد تخفیف

فروشنده می‌تواند برای هر پیشنهاد خود، توسط سیستم تعدادی کد تخفیف ایجاد کند. کد تخفیف یک رشته random ولی یکتا است. سپس خریداران می‌توانند با استفاده از آن‌ها، کالاها را با تخفیف بخرند. مقدار تخفیف به درصد است و از قیمت اصلی کالا کاسته می‌شود. دقت کنید که هر کد تخفیف پس از یک بار استفاده نامعتبر می‌شود.

ورودی	خروجی
POST generateDiscountCode ? offerId <offerId> discountPercent <discountPercent> discountNumber <discountNumber>	<discountCode1> <discountCode2> ...

#### ۴.۴ خریدار

خریدار می‌تواند یکی از کارهای زیر را انجام دهد.

#### ۱.۴.۴ مشاهده کالاهای فروشگاهها

در پاسخ این دستور باید کالاهای موجود در فروشگاه که توسط مدیر وارد شده است، به ترتیب شماره id آنها چاپ شوند.

ورودی	خروجی
GET getProducts	<productId> <productName>
GET getProducts	productId   productName 1 iPhone XS Max 3 Galaxy Note 10 4 Samsung 65" 4K UHD

#### ۲.۴.۴ مشاهده پیشنهادات روی یک کالا

خریدار با دادن شناسه یک کالا، می‌تواند پیشنهادات فروشندگان را به ترتیب صعودی شناسه پیشنهاد دریافت کند.

ورودی	خروجی
GET offersOnProduct ? productId <productId>	<productId>   <productName>   <offerId>   <offerUnitPrice>   <offerAmount>
GET offersOnProduct ? productId 5	productId   productName   offerId   offerUnitPrice   offerAmount 5   iPhone XS Max   1   250.99   98

#### ۳.۴.۴ مشاهده ویژگی‌های یک کالا

خریدار با دادن شناسه یک کالا می‌تواند ویژگی‌های آن را دریافت کند. ترتیب چاپ ویژگی‌ها، به همان ترتیب ورود آنها در فایل‌های وارد شده توسط مدیر فروشگاه است.

ورودی	خروجی
GET productDetail ? productId <productId>	<productName> <feature>: <value>
GET productDetail ? productId 5	iPhone XS Max Weight: 110 CPUFrequency: 4.1 ...

#### ۴.۴.۴ مشاهده نظرات یک کالا

کاربر می‌تواند با این دستور، تمام نظرات روی یک کالا را مشاهده کند. اولین خط، جدیدترین نظر گذاشته شده است.

ورودی	خروجی
GET comments ? productId <productId>	<productName> <userName>   <comment>
GET comments ? productId 5	iPhone XS Max Ali   Innovative design Reza   Android only Ali   RIP steve jobs

#### ۵.۴.۴ افزودن کالا به سبد خرید (به همراه کد تخفیف)

خریدار می‌تواند با دستور زیر یک کالا را به تعدادی از یکی از پیشنهادهای به سبد خریدش اضافه کند. همچنین خریدار می‌تواند در انتها کد تخفیف برای خرید وارد کند. اگر کد تخفیف وارد شده اشتباه بود، باید خطای Bad Request چاپ شود.

اگر تعداد درخواست داده‌شده از تعداد موجود کالا بیشتر باشد، باید خطای Bad Request داده‌شود.

نوشتن کد تخفیف اختیاری است و در صورت درست بودن کد، باید هزینه با محاسبه تخفیف در نظر گرفته شود و آن کد نامعتبر شود. (حتی اگر خریدار سبد خریدش را ثبت نهایی نکند.)

ورودی	خروجی
POST addToCart ? offerId <offerId> amount <amount> [discountCode <discountCode>]	OK   Bad Request

#### ۶.۴.۴ مشاهده سبد خرید

با این دستور خریدار می‌تواند مواردی را که به سبد خریدش اضافه کرده‌است، مشاهده کند. اولین خط، آخرین مورد اضافه شده از نظر زمانی است. اگر سبد خرید خالی است، پیام Empty چاپ می‌شود.

ورودی	خروجی
GET cart	productId   productName   offerId   sellerId   totalPriceConsideringDiscount   amount 11   iPhone XS Max   2   3   21.99   1 5   Galaxy Note 10   1   3   249.99   2 Empty

#### ۷.۴.۴ ثبت سفارش

خریدار پس از اضافه کردن یک یا چند کالا به سبد خریدش می‌تواند با این دستور سفارش خود را نهایی کند. اگر خریدار اعتبار لازم برای خرید آن سبد خرید را نداشت، خطای Bad Request داده می‌شود و در غیر این صورت، مجموع هزینه کل سبد خرید از حساب خریدار کسر می‌شود و قیمتی که هر کالا داشته، به فروشنده آن اضافه می‌شود. (در این قسمت به استفاده کاربران از کد تخفیف توجه کنید.)

ورودی	خروجی
POST submitCart	OK   Bad Request

#### ۸.۴.۴ مشاهده سفارش‌های انجام شده

خریدار می‌تواند خریدهای گذشته خود را ببیند. تعداد این موارد با استفاده از count مشخص می‌شود. خریدها برحسب تاریخ خرید به صورت نزولی چاپ می‌شوند. اولین خط، آخرین خرید را نشان می‌دهد.

اگر سفارشی یافت نشد، خطای Not Found داده‌شود. اگر تعداد سفارش‌ها از count کمتر بود، تمام سفارش‌ها به عنوان پاسخ برگردانده‌شود.

سفارش‌هایی متوالی با استفاده از \*\*\*\* از یکدیگر جدا می‌شوند.

ورودی	خروجی
GET orders ? count <count>	productId   productName   offerId   sellerId   soldPrice   amount
	11   iPhone XS Max   2   3   23.99   1
	5   Galaxy Note 10   1   3   250.99   2
	****
	11   iPhone XS Max   2   3   24.99   2

#### ۹.۴.۴ گرفتن چند گردش مالی آخر

خریدار می‌تواند چند گردش آخر حساب مالی خود را ببیند. تعداد این گردش‌ها براساس count محدود می‌شود. توجه کنید که در ابتدای کار، حساب مالی کاربری مقدار ۰ دارد. خط اول، موجودی کنونی کاربر است. اگر تعداد داده شده در ورودی برای مشاهده، بیشتر از تعداد گردش‌های کاربر بود، به همان تعدادی که گردش وجود دارد، موجودی چاپ می‌شود.

ورودی	خروجی
GET wallet ? count <count>	credit
	8
	20
	30
	0

#### ۱۰.۴.۴ افزایش اعتبار کیف پول

خریدار می‌تواند به مقداری که وارد می‌کند (عدد اعشاری بزرگتر از ۰)، حساب مالی خود را شارژ کند. اگر مقدار داده شده کوچکتر یا مساوی ۰ بود، باید خطای Bad Request داده شود.



ورودی	خروجی
POST chargeWallet ? amount <amount>	OK   Bad Request

#### ۱۱.۴.۴ اضافه کردن نظر به کالا

کاربر با دادن شناسه یک کالا می‌تواند روی یک کالا نظر اضافه کند. اگر شناسه وجود نداشته باشد، باید خطای Not Found داده شود.

ورودی	خروجی
POST comment ? productId <productId> comment <comment>	OK   Not Found

#### ۱۲.۴.۴ مقایسه دو کالا

کاربر می‌تواند با دادن شناسه دو کالا، مقایسه‌ی ویژگی‌های آن دو را ببیند. اگر دو کالا در یک دسته بندی نباشند، باید خطای Bad Request داده شود. اگر شناسه هر یک از دو کالا موجود نباشد، باید خطای Not Found داده شود.

ورودی	خروجی
GET compare ? productId1 <productId1> productId2 <productId2>	<productName1>   <productName2> <featureName>: <product1Value>   <product2Value>
GET compare ? productId1 2 productId2 5	iPhone XS Max   Galaxy Note 10 Weight: 250   264 CPUFrequency: 3.1   2.9 ...

#### ۵ خطاها

#### ۱.۵ خطای عدم وجود

اگر دستور<sup>۴</sup>ی که کاربر وارد می‌کند (قسمت پس از GET یا POST)، در لیست دستورهای تعریف شده نباشد یا کاربر با وارد کردن شناسه‌ی ناموجود در سیستم قصد دسترسی به موجودیتی را داشته باشد، این خطا در خروجی چاپ می‌شود.

خروجی
Not Found

#### ۲.۵ خطای درخواست اشتباه

اگر اولین قسمت خط ورودی کاربر یکی از دو مقدار POST و GET نباشد یا این دستور اطلاعات کافی برای اجرا را در خود نداشته باشد، سیستم خروجی زیر را چاپ می‌کند.

خروجی
Bad Request

<sup>4</sup>Command

### ۳.۵ خطای عدم دسترسی

شرایط رخداد این خطا در دستورها توضیح داده شده است. در صورت رخداد این خطا پیام زیر چاپ می شود.

خروجی
Permission Denied

### ۶ امتیازی

#### ۱.۶ مشاهده لیست خریداران کالاهای فروشنده

در پاسخ این دستور باید تمام خریدهای خریداران پیشنهاد این فروشنده همراه با تعداد و قیمت واحدی که خریده اند، (ممکن است به علت استفاده از کد تخفیف کمتر از قیمت پیشنهادی باشد)، به ترتیب زمانی چاپ شوند. در اولین خط، جدید ترین خرید چاپ می شود.

ورودی	خروجی
GET offerBuyers	productId   offerId   buyerId   soldPrice   amount 11   2   1   23.99   2 5   1   1   250.99   1

#### ۲.۶ مشاهده تمام پیشنهادات فروشنده ها توسط خریدار

خریدار باید بتواند تمام پیشنهادات گذاشته شده را به ترتیب شناسه آنها مشاهده کند.

ورودی	خروجی
GET offers	productId   productName   offerId   offerUnitPrice   offerAmount 5   iPhone XS Max   1   250.99   98 11   Galaxy Note 10   2   23.99   32

### ۳.۶ هش<sup>۵</sup> کردن گذرواژه

همان طور که می دانید گذرواژه<sup>۶</sup> ویژگی ای است که افراد تمایل دارند از دسترسی سایرین حفظ شود. به این دلیل است که هنگام ذخیره سازی گذرواژه آن را به صورت خام در حافظه نگهداری نمی کنند. هش کردن روشی که توسط آن می توان یک رشته را طی فرآیندی غیرقابل بازگشت به رشته ای دیگری تبدیل کرد؛ یعنی هش هر رشته یک رشته ی یکتا است و با هر رشته ی دیگر تفاوت دارد و همچنین از طریق هش یک رشته نمی توان به آن رشته دسترسی پیدا کرد.

در این قسمت باید گذرواژه های افراد را به صورت هش شده در حافظه نگهداری کنید تا از دسترسی به گذرواژه خام پرهیز شود. هش کردن الگوریتم های متفاوتی دارد که می توانید در اینترنت درباره ی آنها جستجو کنید.

<sup>5</sup>Hash

<sup>6</sup>Password

## ۷ نحوه‌ی تحویل و نکات پایانی

- پرونده‌های برنامه‌ی خود را با نام A7-1-SID.zip در صفحه‌ی CECM درس بارگذاری کنید که SID شماره‌ی دانشجویی شماست؛ برای مثال، اگر شماره‌ی دانشجویی شما ۸۱۰۱۹۷۹۹۹ باشد، نام پرونده‌ی شما باید A1-810197999.zip باشد.
- در تخصیص شناسه‌ها به موجودیت‌های سیستم دقت کنید. تمام شناسه‌ها به صورت خودکار و از عدد یک شروع می‌شوند و در صورت حذف شدن یک شناسه مقدار آن جایگزین نمی‌شود؛ حتی اگر مقدار حذف‌شده آخرین مقدار تخصیص داده‌شده باشد.
- برای بررسی خطاها استفاده از مدیریت استثنا<sup>۷</sup> را در نظر بگیرید.
- درستی برنامه شما از طریق آزمون خودکار سنجیده می‌شود. بنابراین، از صحت قالب<sup>۸</sup> ورودی‌ها و خروجی‌های برنامه‌ی خود مطمئن شوید. پیشنهاد می‌شود با استفاده از ابزار diff خروجی خود را با خروجی نمونه داده‌شده مطابقت دهید. همچنین از کپی کردن از متن PDF خودداری کنید زیرا ممکن است قالب ورودی اشتباه شود.
- این پروژه حتماً باید به روش شی‌گرایی و به صورت Multi File پیاده‌سازی شود. همچنین استفاده از makefile اجباری است.
- برنامه‌ی شما باید در سیستم عامل لینوکس و با مترجم g++ با استاندارد c++11 ترجمه و در زمان معقول برای ورودی‌های آزمون اجرا شود. دقت کنید که باید در multifile خود مشخص کنید که از استاندارد c++11 استفاده می‌کنید.
- هدف این تمرین یادگیری شماست. لطفاً تمرین را خودتان انجام دهید. در صورت کشف تقلب مطابق قوانین درس با آن برخورد خواهد شد.

---

<sup>۷</sup>Exception Handling

<sup>۸</sup>format