

به نام خداوند علم و دانش



دانشکده مهندسی کامپیوتر

مبانی کامپیوتر و برنامه نویسی (سی پلاس پلاس)

تمرین دهم (OOP – Class - Method)

دکتر مرضیه داودآبادی

پاییز 1403

طراحان تمرین: آقایان امیرمحمد کمیجانی و متین محمودخانی

- در صورت وجود هرگونه ابهام به طراح پیام دهید.
- باتوجه به وجود تاخیر 10 روزه، امکان تاخیر تحت هیچ شرایطی امکان پذیر نیست.
- انجام تمرین ها تک نفره می باشد.
- زبان برنامه نویسی سی پلاس پلاس است.
- موارد ارسال شده به صورت آنلاین تحویل گرفته خواهند شد.
- ددلاین تمرین: 26 دی ساعت 23:59
- برای دیدن تست کیس های نمونه متنوع به کوئرای درس بپیوندید.
- https://quera.org/course/add_to_course/course/18682
- Password: fcp03
- آیدی طراحان: @MATINN2001 / @AMKomijani

01. مستطیل (محمودخانی)

برنامه‌ای بنویسید تا مساحت مستطیل را با ایجاد یک کلاس به نام `Rectangle` با تابعی به نام `Area` که مساحت را برمی‌گرداند چاپ کند. طول و عرض به عنوان پارامتر به سازنده آن ارسال می‌شود.

➤ *سوالات حتما باید به روش برنامه‌نویسی شی‌گرا حل شوند*.

ورودی

دو عدد صحیح در دو خط ورودی به شما داده می‌شود.

خروجی

مساحت مستطیل به عنوان خروجی باید چاپ شود.

02. دایره (کميجانی)

در این سوال از شما می‌خواهیم یک کلاس `Circle` بسازید. این شیء در هنگام ساخته شدن یک عدد اعشاری (`float`) که مقدار شعاع دایره است به عنوان ورودی می‌گیرد و آن را به صورت محافظت شده ذخیره می‌کند. (از `Constructor` استفاده کنید)

همچنین این کلاس دارای ۲ متود عمومی با نام‌های `getArea` و `getPerimeter` است که مقدار محیط و مساحت دایره را به صورت عددی از جنس `float` برمی‌گرداند.

ورودی

در یک خط شعاع دایره که عددی اعشاری است داده می‌شود.

خروجی

در خط اول مقدار محیط و در خط دوم مقدار مساحت دایره را چاپ کنید. (عدد پی را ۳.۱۴ در نظر بگیرید)

۱.۰۳ اعداد موهومی (محمودخانی)

برنامه‌ای بنویسید که مجموع، تفاضل و حاصل ضرب دو عدد مختلط را با ایجاد کلاسی به نام `Complex` با توابع جداگانه برای هر عملیات که قسمت حقیقی و موهومی آن توسط کاربر وارد می‌شود چاپ کند.

➤ *سوالات حتما باید به روش برنامه‌نویسی شی‌گرا حل شوند*.

ورودی

قسمت‌های موهومی و حقیقی دو عدد در چهار خط به برنامه داده می‌شود که خط اول و دوم به ترتیب قسمت حقیقی و موهومی عدد اول بوده و خط سوم و چهارم هم به ترتیب قسمت حقیقی و موهومی عدد دوم می‌باشد.

خروجی

در خط اول با فاصله، قسمت‌های حقیقی و موهومی حاصل جمع چاپ می‌شوند. در خطوط بعدی مشابه خط اول تفاضل و حاصل ضرب دو عدد گفته شده به همان روش قبلی باید چاپ شود. در هر حالت اگر قسمت مدنظر مساوی صفر شد عدد صفر در خروجی چاپ شود.

04. دانش آموز (کمیجانی)

برنامه‌ای بنویسید که اطلاعات یک دانشجو (نام و شماره تلفن و کد پستی) را گرفته و نمایش دهد. در این برنامه از سازنده (*Constructor*) استفاده شود و کلاس موردنظر یک تابع برای دریافت اطلاعات داشته باشد و یک تابع هم برای نمایش اطلاعات.

سوالات حتما باید به روش برنامه‌نویسی شی‌گرا حل شوند.

ورودی

در سه خط به ترتیب در هر خط نام دانشجو، شماره تلفن دانشجو و کد پستی اش وارد می شود.

خروجی

با همان ترتیب در سه خط جداگانه اطلاعات دریافت شده را نمایش دهید.

05. برنامه سلامت (محمودخانی)

در دو مدرسه ی A و B ، دو برنامه ی سلامت متفاوت در نظر گرفته شده است. در یک مدرسه شیر توزیع می شود و در مدرسه ی دیگر شیر توزیع نمی شود. کارشناسان تغذیه در نظر دارند با بررسی قد، وزن و سن دانش آموزان، آن ها را با یکدیگر مقایسه کنند.

برنامه ای بنویسید که تعداد دانش آموزان هر کلاس بعلاوه اطلاعات سن، قد و وزن آن ها را بگیرد و در یک لیست هر کدام را ذخیره کند و میانگین سن و قد و وزن هر کلاس را به ترتیب در یک خط جدا در خروجی چاپ کند. (به صورت float چاپ شوند.) و در ادامه کلاسی را که میانگین قد افراد آن کلاس بیشتر است، در خروجی چاپ کند. (در ابتدا تعداد و اطلاعات اعضای کلاس A را وارد کنید و سپس اطلاعات کلاس B . در خروجی نیز به ترتیب اطلاعات ورودی چاپ شود).



نکته: در صورتی که میانگین قد افراد کلاس برابر بود، کلاس با میانگین وزن کمتر باید چاپ شود. در صورت برابر بودن میانگین قد و وزن، عبارت **Same** (دقیقا به همین شکل) در خروجی چاپ شود.

➤ استفاده از **class** در حل تمرین ضروری می باشد.

ورودی

توضیح داده شد.

خروجی

توضیح داده شد.

06. بازیابی پروژه (کميجانی)

کیخسرو که دنبال افزایش نمره‌ی خود در درس ریاضی ۱ بود تصمیم گرفت پروژه‌ای انجام دهد و برای استادش بفرستد. اما متاسفانه قبل از اینکه پروژه را برای استاد ایمیل کند، دستش خورد و هر چیزی که در برنامه‌اش قبل از تابع **main** نوشته بود پاک شد 🤖

everything was here destroyed):

help Keykhosro and insert something here!

```
int main (){

    cout << fixed << setprecision(2)

    double x1, x2, y1, y2;

    cin >> x1 >> y1 >> x2 >> y2;

    Complex c = constructComplex(x1, y1);

    Complex c2 = constructComplex(x2, y2);

    Complex c3 = conjugate(c);

    cout << c3.real << ' ' << c3.img << '\n';

    Complex c4 = add(c, c2);

    cout << c4.real << ' ' << c4.img << '\n';

    Complex c5 = sub(c, c2);

    cout << c5.real << ' ' << c5.img << '\n';

    Complex c6 = mul(c, c2);

    cout << c6.real << ' ' << c6.img << '\n';

    printPolarForm(c);

    return 0;

}
```

کیخسرو که دیگر حال و حوصله‌ی دوباره انجام دادن پروژه را ندارد از شما خواسته تا بجای او

پروژه‌اش را کامل کنید.

در این پروژه قرار بود کیخسرو ساختاری بنویسد که عددی مختلط به فرم $x+iy$ در آن ذخیره شود و به کمک توابع نوشته شده در برنامه عملیات زیر روی آن قابل انجام باشد:

1. ساخت و ذخیره کردن عدد مختلط $x+iy$ در صورتی که x و y را داشته باشیم.

```
Complex c;
```

```
c = constructComplex(1, 1.732); //1 + 1.732i
```

2. بدست آوردن مزدوج عدد مختلط c و ذخیره ی آن در متغیری دیگر.

```
Complex c;
```

```
c = constructComplex(1, 1.732); //1 + 1.732i
```

```
Complex c2;
```

```
c2 = conjugate(c); //1 - 1.732i
```

جمع و تفریق دو عدد مختلط c و $c2$ و ذخیره ی آن در متغیری دیگر.

```
Complex c = constructComplex(2, 3); //2 + 3i
```

```
Complex c2 = constructComplex(5, -2); //5 - 2i
```

```
Complex res = add(c, c2); //(2 + 3i) + (5 - 2i) = 7 + i
```

```
Complex res2 = sub(c, c2); //(2 + 3i) - (5 - 2i) = -3 + 5i
```

ضرب کردن دو عدد مختلط c و c_2 و ذخیره‌ی آن در متغیری دیگر.

```
Complex c = constructComplex(1, 1); //1 + i
```

```
Complex c2 = constructComplex(2, -1); //2 - i
```

```
Complex res = mul(c, c2); //(1+i)(2-i) = 3 + i
```

تبدیل به فرم قطبی و چاپ کردن فرم تبدیل شده.

```
Complex c = constructComplex(1, 1); //1 + i
```

```
printPolarForm(c); //1.41e^(i45.02)
```

در مثال بالا خروجی چاپ شده $1.41e^{i45.02}$ خواهد بود که با تقریب خوبی برابر

$\sqrt{2}e^{i45}$ است.

راهنمایی:

- برای \tan^{-1} گرفتن در C++ می‌توانید کتابخانه‌ی `cmath` را `include` کنید و با

استفاده از `atan2(y,x)` مقدار θ را در $\tan\theta = \frac{y}{x}$ برحسب رادیان بدست آورید و سپس

به واحد درجه تبدیل کنید). برای سادگی کار تضمین می‌شود x یا همان بخش حقیقی

عدد هرگز صفر نیست)

- برای رادیکال گرفتن (بعد از `include` کردن `cmath`) از تابع `sqrt` استفاده کنید.

- در تبدیل از رادیان به درجه عدد π را 3.14 در نظر بگیرید.
- در تابع main هیچ تغییری ایجاد نکنید و هر چیزی اضافه می کنید به قبل از main اضافه کنید.

ورودی

توضیح داده شد.

خروجی

توضیح داده شد.

07. کیف پول (محمودخانی)

➤ توجه: این سؤال دارای داوری خودکار نیست و نمره‌ی آن پس از چک کردن دستی کدها، اعلام می شود.

کیانوش در حال گذراندن تعطیلات نوروز است و به این نتیجه رسیده اگر فقط یک چیز باشد که خیلی در شرایط کنونی به آن نیاز داشته باشد، آن یک کیف پول هوشمند است! کیانوش از شما کمک می خواهد تا برنامه‌ای بنویسید که به او میزان موجودی کیف پولش را بگوید و در صورتی که می خواهد خریدی بیشتر از بودجه‌اش بکند، به او هشدار دهد. هم چنین بتواند

مقداری پول پس انداز کند و هر زمان که تنها ۵۰ هزار تومان موجودی برایش باقی ماند، به او اطلاع رسانی کند.

در این سوال، شما باید کلاسی با نام `Wallet` بنویسید به طوری که:

- دو ویژگی `total` و `saving` داشته باشد `total`. متغیری از جنس `int` در واقع کل موجودی کیف پول به غیر از پول های پس انداز شده است `saving`. نیز متغیری از جنس `float` است که مقدار پول پس انداز شده است. دقت کنید هر دو متغیر `total` و `saving` نباید از خارج از کلاس قابل دسترسی باشند.
- یک متد به نام `account_balance` داشته باشد که موجودی کیف پول را برگرداند.
- یک متد به نام `add` داشته باشد که به وسیلهی آن بتوانیم به حساب خود پول واریز کنیم. پس از موفقیت آمیز بودن عملیات واریز نیز پیام مناسب چاپ شود.
- یک متد به نام `save` داشته باشد که مقدار پول و درصد پولی که می خواهد از آن پول ذخیره کند، به کاربر داده شود و آن پول را به ویژگی `saving` اضافه کند.
- یک متد به نام `withdraw` داشته باشد که به وسیلهی آن بتوان از کیف پول برداشت کرد.
- برای مثال، اگر تابع `add` را صدا بزنیم:

```
int main() {  
  
    Wallet mywal(1000,0);  
  
    mywal.add(100);  
  
}
```

ورودی اول مقدار پول اولیه است که به جای **total** قرار می گیرد. مقدار پس انداز اولیه را صفر در نظر بگیرید. به عنوان خروجی خواهیم داشت:

100 added!

تابع **withdraw** وظیفه‌ی برداشت از کیف پول شما را بر عهده دارد.

```
int main() {  
  
    Wallet mywal(1000,0);  
  
    mywal.withdraw(4000);  
  
    mywal.withdraw(100);  
  
}
```

این تابع ابتدا چک می کند اگر مقداری که می خواهید برداشت کنید بیشتر از موجودی باشد، در خروجی عبارت **low account balance** را چاپ می کند. در غیر این صورت، به اندازه‌ی داده شده از کل پول (**total**) برداشت می کند و در نهایت **withdrawed successfully** را چاپ می کند.



مثلا خروجی تابع بالا به صورت زیر است:

low account balance

100 withdrew successfully

تابع `account_balance` مقدار `total` را برمی گرداند.

```
int main() {  
  
    Wallet mywal(1000,0);  
  
    mywal.withdraw(100);  
  
    cout<< mywal.account_balance();  
  
}
```

در خروجی خواهیم داشت:

100 withdrew successfully

900

تابع `save` مقداری از پول داده شده که به عنوان اولین ورودی می گیرد را به ویژگی

`save` اضافه می کند.

```
int main() {  
  
    Wallet mywal(1000,0);  
  
    mywal.save(4000,0.1);  
  
}
```



```
cout<< mywal.account_balance();  
  
}
```

در خروجی خواهیم داشت:

400.000000saved!

4600

ورودی

توضیح داده شد

خروجی

توضیح داده شد.

08. بقالی (کميجانی)

➤ توجه: این سؤال دارای داوری خودکار نیست و نمره‌ی آن پس از چک کردن دستی کدها، اعلام می‌شود.

در بقالی سر کوچه سیستم جدیدی قرار است پیاده سازی شود که با استفاده از آن صاحب مغازه می‌تواند اجناس موجود در انبار و قفسه‌های خود را مدیریت کند. صاحب مغازه فکر تبدیل شدن



به یک فروشگاه زنجیره‌ای را در سر دارد. به او کمک کنید تا بتواند با مدیریت بهتر منابعش به هدف خود نزدیک‌تر شود.

در این تمرین شما باید برنامه‌ای را پیاده‌سازی کنید که بتواند از محیط کنسولی (command line) کاربر ورودی بگیرد. هر ورودی در یک خط داده می‌شود که شامل چند بخش است. در بخش ابتدایی نام دستور می‌آید و سپس پارامترهای مربوط به آن داده می‌شود. مانند فرم زیر:

```
COMMAND argument1 argument2 ....
```

➤ نکته: نام دستور و پارامترها با فاصله (Space) از یک دیگری جدا شده‌اند.

دستورات مورد نیاز در جدول زیر داده شده است.

Command Name	Argument 1	Argument 2	Argument 3
Define	Name	Buying Cost	Selling Cost
Delete	Name	-	-
Sell	Name	Count	-



Command Name	Argument 1	Argument 2	Argument 3
Buy	Name	Count	-
Status	-	-	-
Financial	-	-	-
Exit	-	-	-

➤ توضیحات مربوط به هر دستور:

***Define*:**

این دستور مشخصات یک کالا را تعریف می کند. در پارامتر اول نام محصول مشخص می شود. پارامتر دوم قیمت خرید آن از کارخانه را نشان می دهد و پارامتر سوم قیمت فروش آن به خریدار است.

***Delete*:**

این دستور مشخصات یک کالای تعریف شده را حذف می کند. در پارامتر اول این دستور اسم کالایی که مشخصاتش باید حذف شود داده می شود. توجه کنید که این دستور زمانی می تواند

مشخصات محصول را حذف کند که هیچ نمونه‌ای از آن در انبار موجود نباشد. اگر نمونه‌ای از آن موجود بود باید در خروجی و در یک خط جداگانه یک علامت تعجب (!) چاپ کنید.

***Sell*:**

این دستور به معنای فروختن یک نمونه جنس به خریدار است. در پارامتر اول اسم جنس می‌آید و در پارامتر بعدی تعداد آن می‌آید. اگر این تعداد محصول در انبار موجود نبود باید در یک خط جداگانه یک علامت تعجب نمایش داده شود. اگر این دستور اجرا شود تعداد کالاهای موجود در انبار کاهش می‌یابد. میزان کاهش برابر تعداد مشخص شده در پارامتر دوم است.

***Buy*:**

این دستور به معنی خرید یک محصول از کارخانه است. با اجرای این دستور تعداد اجناس موجود در انبار افزایش می‌یابد. توجه شود که نام محصول خریداری شده در پارامتر اول می‌آید و تعداد خریداری شده در پارامتر دوم.

***Status*:**

این دستور لیست تمام کالاهای تعریف شده را با اطلاعات آن‌ها نمایش می‌دهد. این اطلاعات شامل نام، قیمت خرید، قیمت فروش و تعداد موجود از آن در انبار است. به مثال زیر توجه کنید .
(بخش \$ Status خط ورودی است که کاربر وارد کرده است.)

\$Status

A 1000 1350 10

B 2500 2700 0

C 5300 5600 2

***Financial*:**

این دستور اطلاعات مالی را نمایش می دهد. شما باید میزان فروش، میزان خرید و سود حاصل را نمایش دهید. (سود حاصل همان تفاضل میزان فروش از میزان خرید است) فرمت نمایش به همان ترتیب گفته شده می باشد و هر بخش با فاصله از هم جدا شده اند.

***Exit*:**

این دستور اجرای برنامه را متوقف می کند.

➤ نکته

سعی کنید از تمام امکانات Object-Oriented و در پیاده سازی خود استفاده کنید. لازم به ذکر است که پیاده سازی این تمرین تنها در یک کلاس با یک تابع main مجاز نمی باشد. برای حل مسئله حتما باید از روش Object-Oriented استفاده شود.

➤ نحوه ارسال

کل فایل‌های مورد نیاز برای اجرا شدن تمرین را به همراه یک فایل PDF که شامل توضیحات برنامه است را در قالب یک فایل ZIP ارسال کنید.

ورودی

توضیح داده شد.

خروجی

توضیح داده شد.

09. فروشگاه کتاب (محمودخانی)

➤ توجه: این سؤال دارای داوری خودکار نیست و نمره‌ی آن پس از چک کردن دستی کدها،

اعلام می‌شود.

یک فروشگاه کتاب به تازگی افتتاح شده است و می‌خواهد سیستمی برای ذخیره و جستجوی

کتاب‌های موجود در فروشگاه ایجاد کند.

سیستم مورد نظر دارای قابلیت‌های زیر خواهد بود:

➤ اضافه کردن کتاب جدید



در این بخش، سیستم با گرفتن `name` و `id` و `publisher` و `price` و `category` از ورودی یک کتاب با مشخصات داده شده به لیست کتاب ها اضافه میکند.

```
add name id publisher price category
```

توجه کنید که `id` برای هر یک از کتاب ها یکتا است در صورت وجود کتاب با شناسه یکسان خطای `Book with same id already exist` نشان داده خواهد شد. همچنین شناسه هر کتاب پس از مقدار دهی اولیه تغییر نخواهد کرد.

➤ تغییر قیمت کتاب ها

در این بخش، سیستم با دریافت `id` و `newPrice` در صورت وجود کتاب با شناسه وارد شده مقدار قیمت آن کتاب به مقدار جدید تغییر خواهد کرد و در صورت وجود نداشتن کتاب عبارت `id did not match a book` نشان داده شود.

```
change id price
```

جستجو بر اساس دسته بندی کتاب

در این بخش، سیستم با گرفتن یک `category` کتاب هایی که در دسته بندی یکسان قرار دارند را در خروجی چاپ می کند.

```
search category
```

توجه کنید که در هنگام جستجوی دسته بندی ها کلمات وارد شده حساس به حروف (case sensitive) نیستند (بطور مثال کلمات comedy و CoMedy برابر خواهند بود).

➤ حذف کتاب

در این بخش، سیستم با دریافت id در صورت وجود کتاب با شناسه وارد شده، آن را از لیست کتاب ها حذف می کند و دیگر در جستجو یا ذخیره سازی نشان داده نمیشود.

```
delete id
```

#ذخیره سازی برای نگه داری اطلاعات وارد شده توسط کاربران و تغییرات اعمال شده، از فایل های **Csv (Comma-separated values)** استفاده می کنیم. بدین صورت که در اولین خط از فایل اسامی ستون های ذخیره شده جدا شده با کاراکتر ',' می آید.

```
id,name,publisher,price,category
```

سپس در هریک از خطوط بعدی مقادیر فیلد های یک کتاب با ترتیب مشخص شده برای ستون ها به صورت جدا شده با کاراکتر ',' قرار می گیرد.

#اجرا برنامه ی شما باید در ابتدای اجرای برنامه مقادیر ذخیره شده در فایل را بخواند و کتاب های ذخیره شده را بازبینی کند تا تغییرات را بر روی آنها انجام دهد.



همچنین در صورت وارد شدن دستور `exit` فایل ذخیره را با استفاده از مقادیر جدید بازنویسی کرده و به برنامه پایان دهد.

ورودی

توضیح داده شد.

خروجی

توضیح داده شد.