



مبانی برنامه‌سازی پایتون

نیم‌سال دوم ۹۶-۹۷

استاد: علی طاهری

تمرین سوم

آشنایی با مبانی برنامه‌سازی

مهلت ارسال: ۲۴ اردیبهشت

به موارد زیر توجه کنید:

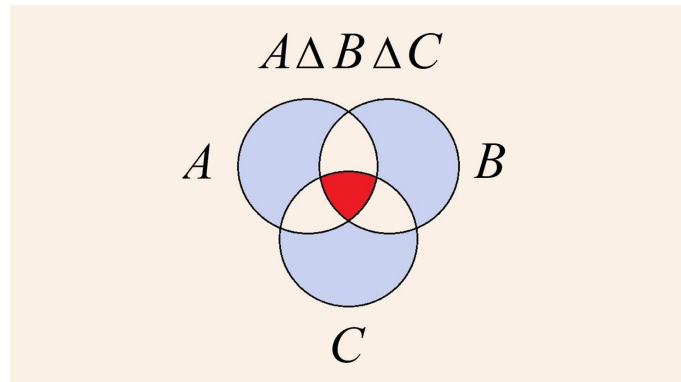
- پاسخ تمرین را در سامانه‌ی کوثرای درس بارگذاری نمایید.
- مهلت ارسال پاسخ تمرین تا ساعت ۲۳:۵۹ روز اعلام‌شده است. بهتر است نوشتن تمرین را به ساعات پایانی موکول نکنید.
- هم‌کاری و هم‌فکری شما در حل تمرین مانعی ندارد، اما پاسخ‌های هر کس حتماً باید توسط خود شخص نوشته شده باشد.
- مبنای درس، اعتماد بر پاسخ‌های ارسالی از سوی شماست؛ بنابراین ارسال پاسخ در سامانه‌ی داوری به این معناست که پاسخ آن تمرین، توسط شما نوشته شده است. در صورت تقلب یا اثبات عدم نوشتار پاسخ حتی یک سوال از تمرین، با هر دو طرف تقلب‌گیرنده و تقلب‌دهنده برخورد شدیدی صورت خواهد گرفت.
- مجموع تأخیر پذیرفته‌شده برای تمرینات، ۳۶ ساعت است. چنانچه تأخیری بیش از این مقدار داشته‌باشید، نمره‌ی یک‌سوم برای آن در نظر گرفته خواهد شد.

سؤال ۱. حاصل جمع تجمعی (۵ نمره)

برنامه ای بنویسید که با گرفتن لیستی از اعداد، لیست "حاصل جمع تجمعی" آنها را بازگرداند. مثلاً با گرفتن لیست [۳، ۶، ۴]، [۴، ۷، ۱۳] را بازگرداند.

سؤال ۲. تفاضل متقارن (۱۰ نمره)

تفاضل متقارن دو مجموعه، مجموعه عناصری است که عضو دو مجموعه هستند ولی در اشتراک آنها قرار ندارند. تابعی بنویسید که تفاضل متقارن دو مجموعه را محاسبه کند. با استفاده از این تابع، برنامه ای بنویسید که سه مجموعه به شکل لیست به عنوان ورودی گرفته و تفاضل متقارن آن سه مجموعه را بازگرداند.



نمونه‌ی ورودی	نمونه‌ی خروجی
1,2,3 3,4,5,6 1,3,5,7	2,3,4,6,7

سؤال ۳. مدیریت استثناها (۱۰ نمره)

برنامه ای بنویسید که با گرفتن یک لیست، لگاریتم آرایه های آن را محاسبه کند. خروجی باید به صورتی باشد که در هر خط، لگاریتم آرایه متناظر نمایش داده شود. در صورتی که آرایه مورد نظر، عددی منفی باشد، پیغام negative integer و در صورتی که از جنس عدد نباشد، پیغام not supported و در انتها پیغام done! را نمایش دهد. از دستورات try و except استفاده نمایید.

نمونه‌ی ورودی	نمونه‌ی خروجی
10,-5,1.2,'apple'	1.0 negative integer 0.07918124604762482 not supported done!

سؤال ۴. تابع یک به یک (۱۰ نمره)

برنامه ای بنویسید که تابعی با دامنه و برد اعداد را به صورت یک دیکشنری بگیرد. سپس معکوسش را بسازید و نیز اعلام کنید که آیا تابع اولیه یک به یک است و یا خیر. ورودی یک دیکشنری به شما داده خواهد شد که کلیدهای آن دامنه ی تابع و مقادیر آن برد تابع هستند. خروجی تابع معکوس را چاپ کنید، به طوری که هر مولفه آن معکوس مولفه نظیرش در دیکشنری اولیه باشد. سپس در خط بعدی اگر تابع اولیه یک به یک بود True و در غیر این صورت False چاپ کنید. به بزرگ بودن حرف اول این کلمات دقت کنید.

نمونه‌ی ورودی	نمونه‌ی خروجی
1:2,2:3,5:3,4:5	2:1,3:2,3:5,5:4 False

سؤال ۵. ضرب ماتریسی (۱۵ نمره)

برنامه ای بنویسید که دو ماتریس به عنوان ورودی گرفته و حاصلضرب اولی در دومی را محاسبه کند. نوع نمایش ماتریس‌های ورودی، ابتدا به برنامه داده خواهد شد. اگر ماتریس‌های ورودی به صورت لیست تو در تو باشند، ابتدا عبارت list داده خواهد شد و سپس دو ماتریس. در صورتی که ماتریس‌های ورودی به صورت فرهنگ داده ای باشند ابتدا عبارت sparse داده خواهد شد. توجه کنید که شکل نمایش ماتریس خروجی مانند ماتریس‌های ورودی باشد.

نمونه‌ی ورودی	نمونه‌ی خروجی
list [1,0],[1,1] [2,0],[-1,2]	[2,0],[1,2]
sparse (0,0):1,(1,0):1,(1,1):1 (0,0):2,(1,0):-1,(1,1):2	(0,0):2,(1,0):1,(1,1):2

سؤال ۶. پشته (۱۵ نمره)

میخواهیم عملکرد یک پشته را شبیه سازی کنیم. فرض کنید پشته ای به عمق ۱۰ داریم که از دو دستور push و pop پشتیبانی میکند.

push X : مقدار X را روی پشته قرار می دهد. (X یک عدد صحیح است.)

pop : بالاترین عنصر پشته را حذف کرده و سپس برمی گرداند.

طبیعی است در صورتی که پشته پر باشد، دستور push و هنگامی که پشته خالی باشد، دستور pop بی معنی است. در نتیجه، در صورتی که دو خطای بالا اتفاق افتاد، باید به ترتیب پیغامهای خطایی به دو شکل زیر چاپ شود.

پر بودن پشته : Stack is full.

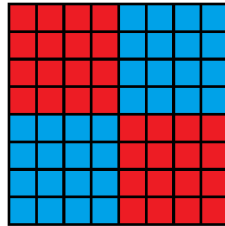
خالی بودن پشته : Stack is empty.

نمونه‌ی ورودی	نمونه‌ی خروجی
push 1	
push 2	
pop	2
pop	1
pop	Stack is empty.
push 3	
pop	3
push 4	

سؤال ۷. بازی چهارگوش (۱۵ نمره)

یک صفحه شطرنجی مربعی به طول ضلع 2^n داریم. این صفحه 2^n در 2^n خانه دارد که در هر خانه آن یک عدد صحیح نوشته شده است. بازی چهارگوش به این صورت تعریف می‌شود که در هر مرحله کارهای زیر را انجام می‌دهیم:

- صفحه را به چهار صفحه مربعی مساوی کوچک تقسیم می‌کنیم. (اگر هم از صفحه تنها یک خانه باقی مانده بود، آن را دور ریخته و به مرحله شماره ۴ می‌رویم.)
- دو صفحه مقابل از این چهار صفحه را برای خود نگه داشته و دوتای دیگر را حذف می‌کنیم. مثلاً در شکل زیر با دو صفحه آبی و یا دو صفحه قرمز که روبروی هم هستند را برای خود نگه می‌داریم:



- حال از دو صفحه باقی مانده دوباره یکی را نگه می‌داریم و دیگری را دور می‌ریزیم.
- به اندازه بیشترین عدد از صفحه دور ریخته شده به خود امتیاز می‌دهیم!

با این اوصاف اگر در انتخاب صفحه‌ها دقت به خرج دهیم می‌توانیم امتیاز بیشتری کسب کنیم. برنامه‌ای بنویسید که بیشترین امتیازی که می‌توان در این بازی بدست آورد را محاسبه کند. در خط اول عدد Π (حداکثر ۸) را دریافت کنید. در 2^n خط بعدی، در هر خط 2^n عدد که متناظر با اعداد خانه‌های صفحه شطرنجی هستند، می‌آید. در خروجی نیز باید بیشترین امتیاز ممکن را چاپ کنید. راهنمایی: از تابع بازگشتی استفاده کنید.

نمونه‌ی ورودی	نمونه‌ی خروجی
1 3 1 4 3	6
2 1 3 2 1 3 4 1 -1 2 -2 -3 2 8 7 0 6	12

سؤال ۰۸ رمزگذاری (۲۰ نمره)

برنامه ای بنویسید که عمل رمزگذاری را روی یک فایل متنی انجام دهد. روش رمز کردن هم به این صورت است که فایلی به عنوان کلید به شما داده می شود که در آن مشخص شده است که کدام کاراکتر باید با کدام کاراکتر دیگر جابه جا شود. سایر کاراکترهای متن که در فایل کلید جایگزین آن ها مشخص نشده است، به همان صورت اولیه در متن خروجی آورده می شوند. متن اصلی که شما باید آن را رمزگذاری کنید در فایل input.txt می باشد و شما به وسیله ی فایل key.txt که محتوی کلید می باشد ، متن اصلی را تغییر داده و حاصل را در فایل output.txt ذخیره کنید.

فایل ورودی	فایل کلید	فایل خروجی
Hello, this is input file content!	t : b c : w H : m , : ? i : 0	Mello? bh0s 0s 0npub f0le wonbenb!

تذکر : این تمرین را فقط در سایت quera آپلود کنید و نمره دهی آن در تحویل حضوری صورت خواهد گرفت.

سؤال ۹. مرتب‌سازی درجی (۲۰ نمره) (سوال امتیازی)

در این سوال می‌خواهیم مرتب‌سازی درجی را به صورت بازگشتی (صعودی) پیاده‌سازی کنیم. برای این کار از دو تابع `insert` و `sort` استفاده خواهیم کرد که هر دو به صورت بازگشتی خواهند بود. توضیح این دو تابع در پایین آمده است. تابع `insert` به این صورت عمل می‌کند که یک لیست مرتب شده از اعداد و یک عدد به عنوان ورودی می‌گیرد و عدد را در جای درست خود در لیست درج می‌کند. این کار به این صورت انجام می‌پذیرد که اگر عدد داده شده، از همه ی اعداد لیست کوچکتر بود، آن را در ابتدای لیست درج می‌کند و لیست نهایی را برمی‌گرداند. در غیر این صورت، به طور بازگشتی عدد مورد نظر را در لیست درج می‌کند.

تابع `sort` به این صورت عمل می‌کند که دو لیست از اعداد به عنوان ورودی می‌گیرد که یکی لیست اعدادی است که تا به این جای کار مرتب شده اند و دیگری لیست اعدادی است که هنوز در لیست نهایی مرتب درج نشده اند. این تابع اولین عنصر لیست اعداد نامرتب را در لیست اعداد مرتب شده `insert` می‌کند و به طور بازگشتی بقیه ی لیست را نیز مرتب می‌کند. حال، بعد از پیاده سازی این دو تابع، برنامه ای بنویسید که ابتدا عدد N را به عنوان تعداد عناصر لیست دریافت کند، سپس N عنصر لیست را از کاربر گرفته و لیست را به صورت مرتب شده و در یک خط چاپ کند. عناصر باید با `space` از هم جدا شوند.

فایل ورودی	فایل خروجی
5 5 -2 -3 10 0	-3 -2 0 5 10

تذکر : استفاده از کدهای آماده به عنوان تقلب در نظر گرفته خواهد شد.