

به نام خدا

گزارش تمرین سری 6

برنامه نویسی پیشرفته(1)

رضا مرادی 9623100

• Apha6.h

پروتوتایپ های توابع مورد نظر به صورت صحیح وارد شده است و چون به جای cpp فایل hpp داریم در آخر باید آنرا اینکلود کنیم. کتابخانه ی algorithm برای استفاده از sort اینکلود شده است.

• Apha6.hpp

خط 1 تا 4- در این تابع قصد داریم هر نوع ورودی را به هر نوع خروجی مورد نظر convert کنیم برای اینکار لازم است که template این تابع دوتا ورودی داشته باشد یکی a و دیگری O که به ترتیب تایپ ورودی و خروجی می باشد که خروجی دقیقا از ابتدا تا انتها باید ایتريت کند که همه ی sequence ها یک همچین constructor دارن که فقط کافيست المان ابتدایی و انتهایى را به آن بدهيم.

خط 6 تا 11- ابتدا یک متغير ostream_iterator میسازيم که بين هر ايتريت آن کارکتر " " را پرينت میکند سپس با استفاده از کپی هر ايتريت را چاپ میکنيم و در آخر به خط بعد میرويم.

خط 13 تا 18- برای اینکه رنگ n را چاپ کنیم چاره ای نداريم جز اینکه دنباله را sort کنیم برای اینکار ابتدا دنباله را به شکل وکتور در می آوريم تا در استفاده از sort به مشکل برنخوريم سپس با کمک lambda function باید مشخص کنیم که به چه ترتیبی میخواهيم sort کنیم یا به اصلاح به چه منطقی؛ سپس بعد از اینکه آن هارا از کوچک به بزرگ چاپ کردیم عضو nام را از آخر ريترن میکنيم.

خط 20 تا 25- مشابه قبل است فقط باید نزولی مرتب کنیم و سپس خود sequence را ريترن کنیم.

خط 27 تا 32- دقیقا مشابه قبل است.

• League.py

• Class Match

خط 4 تا 9- مطابق با آن صورتی که باید باشد عضو های کلاس را مقدار دهی میکنيم از split() برای تبدیل یک متن به string های جدا و قابل استخراج و قابل صدا زدن استفاده میکنيم.

خط 12 تا 25- مطابق با آنچه صحیح است خروجی ها به صورت tuple ساخته میشوند که اکثرا لاجیکال هستند از int() برای cast کردن استفاده میکنيم تا با ارور مواجه نشویم برای away_tuple میتوانيم از home_tuple استفاده کنیم در نهایت هر دو را ريترن میکنيم.

خط 27 تا 31- در یک فرمت زیبا خروجی را چاپ میکنیم در واقع وقتی پرینت بر روی آبجکت کلاس صدا زده میشود این توابع به کار می آیند.

• Class Weeks

خط 34 تا 43- ابتدا متغیر های کلاس با استفاده از متغیر های ورودی مقدار دهی میکنیم سپس در تابع `add_match` ابتدا بررسی میکنیم که مسابقه مورد نظر در لیست مسابقه ها موجود نباشد سپس آنرا اضافه میکنیم.

خط 45 تا 55- در یک فرمت زیبا و خط به خط مسابقات هر هفته را پرینت میکنیم در اینجا توابع `__str__` و `__repr__` کلاس قبل صدا زده میشوند.

• Class Team

در ابتدا تمام متغیر های کلاس را بر اساس متغیر های ورودی مقدار دهی میکنیم و به شمار تیم ها هر بار که کانستراکتور صدا زده میشود یکی اضافه میکنیم سپس با اضافه کردن مسابقه بر اساس تابع `data` که در کلاس `match` تعریف شد متغیر های کلاس را مقدار دهی میکنیم اینرا در نظر میگیریم که اسم تیم فعلی برابر عضو اول `tuple` است یا دوم؛ در تابع `destructor` از شمار تیم ها یکی کم میکنیم و در انتها در یک فرمت درست خروجی هارا پرینت میکنیم.

• Class League

ابتدا متغیر های کلاس را `initialize` میکنیم و بر اساس ورودی مقدار دهی میکنیم سپس در تابع `__add_match__` به هر تیم مسابقه را اضافه میکنیم این تابع در تابع بعدی کال میشود. در تابع `add_week` ابتدا هفته را به لیست هفته های مسابقه اضافه میکنیم سپس برای تمام مسابقه های یک هفته تابع قبل را صدا میزنیم حال باید تیم هارا `sort` کنیم برای اینکار از `lambda function` استفاده میکنیم و اینکار به صورت نزولی انجام میدهیم با دستور `reverse = True` سپس رنک های هر تیم را مشخص میکنیم مشخص است وقتی که تیم ها `sort` شده است با حلقه روی اعضا `index` هر عضو همان رنکش است در انتها به لیست کل تیم ها هر تیم را اضافه میکنیم؛ در پایان در یک فرمت مناسب متغیر های لازم را پرینت میکنیم.

• League.py

در ابتدا تمام کلاس‌ها را از league کال می‌کنیم؛

در تابع `read_data` فایل را به نام `fs` می‌خوانیم از `"r"` استفاده می‌کنیم تا فایل ورودی در حالت خواندن باشد سپس یک حلقه روی هر خط فایل اجرا می‌کنیم و هر خط را می‌خوانیم سپس محتوای بین دو براکت در هر خط را داخل `str_list` میریزیم با کمک دستور `split()` و حلقه هر کلمه را در متغیر مذکور صدا می‌زنیم سپس با کمک دستور `strip()` هر کلمه بین دو متغیر ' (کاما) را بدست می‌آوریم سپس بر اساس اینکه در فایل `data.txt` اسم تیم‌ها بین `"` و `"` یا `"` هستند به لیست تیم‌ها تیم جدید را اضافه می‌کنیم. دقیقاً روند مشابهی را برای اضافه کردن هفته داریم. سپس با استفاده از داشته‌ها تیم‌ها را می‌سازیم و در نهایت لیگ را به لیگ هفته‌ها را اضافه می‌کنیم و ریترن می‌کنیم.

در نهایت همه‌ی تست‌ها پاس می‌شوند:

```
..[Esteghlal 3 - 1 Zobahan, Golgozar 0 - 1 Perspolis, Nassaji 4 - 2 Peykan]
....
-----
Ran 6 tests in 0.008s
OK
```