دانشگاه صنعتی همدان

گروه کامپیوتر

عنوان پروژه: طراحی و پیاده‌سازی سیستم شبیه‌ساز جام جهانی فوتبال ۲۰۲۶

- **نام و نام خانوادگی دانشجو:** سارینا خرام
- **تاریخ تحویل:** **1405/04/30**
- **درس مربوطه:** برنامه‌نویسی پیشرفته

مقدمه و هدف پروژه

هدف از این پروژه، پیاده‌سازی یک نرم‌افزار شبیه‌ساز برای مسابقات جام جهانی ۳۲ تیمی فوتبال با بهره‌گیری از اصول برنامه‌نویسی شیءگرا (OOP) در زبان پایتون است. این برنامه قادر است داده‌های تیم‌ها را از فایل متنی بارگذاری کرده، قرعه‌کشی گروه‌ها را بر اساس رنکینگ فیفا به صورت خودکار انجام دهد، مرحله گروهی و حذفی را با منطق‌های فوتبالی شبیه‌سازی کرده و در نهایت گزارش‌های آماری به کاربر نمایش دهد.

معماری و ساختار شیءگرا (توضیح کلاس‌ها)

- **کلاس Team:**

وظیفه نگهداری اطلاعات پایه هر تیم (نام، قدرت حمله، دفاع و رنکینگ) و متغیرهای پویا (گل های زده ، گل های خورده ، امتیاز ، اسم گروه تیم) در طول تورنمت را بر عهده دارد. همچنین متدهای goals_difference () برای برگرداندن تفاضل گل تیم ، reset_stats() برای پاکسازی داده‌ها در شبیه‌سازی‌های متوالی و simulate_match () برای شبیه سازی مسابقه و در نهایت برگرداندن گل تیم ، گل رقیب ، برنده و نتیجه پنالتی در صورتی که در مرحله حذفی بازی به پنالتی ختم شود در این کلاس قرار دارند.

- **کلاس Match:**

وظیفه مدیریت و برگزاری یک مسابقه فوتبال بین دو تیم را بر عهده دارد. این کلاس در تنها متد خود (play()) آپدیت آمار(نتیجه هر شبیه سازی به این متد برمیگردد) و امتیازدهی مرحله گروهی را انجام میدهد.

- **کلاس Group:**

کلاس مدیریت مسابقات، رتبه بندی و تعیین تیم های صعود کننده هر گروه است. در متغییر های ثابت خود برای هر شی اسم گروه و لیستی شامل تیم های آن گروه را نگه میدارد و در متد play_all_matches() تیم های گروه را برای مسابقه میفرستد. در متد get_ranking () تیم های گروه را بر اساس 1. امتیاز 2. تفاضل گل 3. گل زده 4. بازی مستقیم 5. قرعه کشی تصادفی مرتب میکند و برمیگرداند و در متد advance_team() دو تیم برتر گروه را برمیگرداند.

- **کلاس KnockoutStage:**

ساختار درخت مسابقات حذفی از یک هشتم تا فینال را مدیریت کرده و برندگان هر مرحله را برای صعود به مرحله بعد استخراج می‌کند. در واقع برای هر مرحله طبق براکت حذفی فیفا لیستی از مسابقات آن مرحله می سازد، برای اجرای مسابقه ارسال میکند و تیم های برنده را با متد ()get_winners در لیست ذخیره میکند و به مرحله بعد میفرستد. در پایان متدی برای نمایش نتایج آن مرحله حذفی دارد تا در صورت درخواست نتایج را نمایش دهد.

- **کلاس WorldCupSimulator:**

کلاس هماهنگ‌کننده کل پروژه که ارتباط بین تمام کلاس‌ها، اعمال سیدبندی‌ها، ذخیره‌سازی داده‌های آخرین براکت و شبیه‌سازی‌های انبوه ۱۰۰۰ باره را مدیریت می‌کند. فایل را میخواند و اطلاعات تیم ها را در لیست تیم های خود نگه میدارد سپس سیدبندی و قرعه کشی گروه ها را انجام میدهد و گروه ها را در لیست گروه های خود نگه میدارد. سپس بازی های گروهی را صدا میزند و براکت حذفی را ساخته و بازی های حذفی را صدا میزند.

کتابخانه‌های جانبی استفاده شده و علل آن‌ها

- **کتابخانه random:**

انجام قرعه کشی تصادفی تیم ها با استفاده از (`shuffle.`) و شبیه‌سازی احتمال گل شدن ضربات پنالتی عادی و ناگهانی با (()`random.`) و قرعه‌کشی تصادفی در صورت تساوی مطلق دو تیم در جدول گروهی.

- **کتابخانه csv:**

جهت خواندن هوشمند و ساختاریافته اطلاعات تیم‌ها از فایل `worldcup_2026_teams.cs`

- **کتابخانه os:**

تعامل با سیستم‌عامل. متد `os.path.exists` برای دیباگ و جلوگیری از کرش برنامه در صورت عدم وجود فایل CSV در مسیر اصلی به کار رفته است.

- **کتابخانه numpy:**

برای محاسبات علمی، ریاضی و کار با داده‌ها کاربرد دارد. هسته اصلی این کتابخانه به زبان C نوشته شده است که باعث می‌شود عملیات روی آرایه‌ها و محاسبات آماری با سرعتی بیشتر از لیست‌های عادی پایتون انجام شوند. زیرا زبان C کامپایل مستقیم به زبان ماشین انجام میدهد و ذخیره پیوسته داده ها در حافظه. تولید اعداد تصادفی بر اساس تابع توزیع پواسون (`np.random.poisson`) برای تعیین تعداد گل های هر مسابقه و وقت اضافه. استفاده از این تابع باعث افزایش چشمگیر سرعت اجرا شد.

منطق محاسباتی و فرمول‌های پروژه

الف) شبیه‌سازی تعداد گل‌ها (توزیع پواسون)

برای شبیه‌سازی گل‌ها از مدل ریاضی توزیع پواسون استفاده شده است. نرخ میانگین گل‌زنی هر تیم (λ) بر اساس قدرت حمله خود و قدرت دفاع حریف محاسبه می‌شود.

استفاده از تابع `max` با کف 0.1 به دو دلیل انجام شد؛ اول جلوگیری از منفی شدن نرخ پواسون که باعث کرش کردن کتابخانه NumPy می‌شد، و دوم حفظ شانس حداقلی و برای تیم‌های ضعیف‌تر.

اما مسئله ای که با استفاده از فرمول لامبدا ذکر شده در پروژه در اجرای برنامه پیش می آید تفاوت قدرت تیم‌های مدعی را با تیم‌های ضعیف‌تر در بازه باریکی از 1.2 تا 1.6 نگه میدارد این امر باعث بالا رفتن ضریب شانس و ایجاد نتایج غیرمنتظره در تک‌بازی‌های حذفی می‌شود یا میتوان گفت اگر قدرت دفاع و حمله تیم ها اعداد نزدیک به هم باشند نرخ پواسون نزدیک به هم میشود در نتیجه عامل تصادفی به راحتی تیم ضعیف را برنده میکند.

راهکار برای حل کردن مسئله: رنکینگ تیم ها را که در فایل خوانده ایم در فرمول دخیل کنیم. در این صورت تیم های قوی تر در تصادف گل زنی احتمال بیشتری پیدا میکنن.

ب) سیستم رتبه‌بندی مرحله گروهی و شکستن تساوی‌ها

مرتب‌سازی جدول بر اساس اولویت‌های فیفا انجام می‌شود: ۱. امتیاز کل، ۲. تفاضل گل، ۳. گل زده کل.

- **قانون بازی مستقیم (Head-to-Head)**: اگر دو تیم همسایه در هر ۳ معیار بالا کاملاً برابر بودند، الگوریتم به سراغ مسابقه مستقیم آن‌ها می‌رود و در صورت برنده داشتن تیم دوم، جایگاه آن‌ها را در جدول تعویض می‌کند.
- **قرعه‌کشی تصادفی** : برای پیاده‌سازی قانون قرعه‌کشی فیفا در تساوی‌های مطلق (زمانی که بازی مستقیم هم مساوی شده)، لیست تیم‌ها قبل از مرتب‌سازی یک بار کپی شده و شافل تصادفی (`shuffle`) روی آن انجام می‌شود تا شانس کاملاً عادلانه 50٪ رعایت شود.

روند دیباگ و بهینه‌سازی کدهای پروژه

1. **مشکل پر شدن ترمینال در شبیه‌سازی انبوه:**در ابتدا متد شبیه‌سازی ۱۰۰۰ باره، تمام براکت‌ها را چاپ می‌کرد که باعث کندی شدید برنامه می‌شد. با اضافه کردن آرگومان `show_results=False` این مشکل برطرف شد و چاپ براکت فقط به گزینه ۴ و ۶ منو محدود گشت.

2. **جلوگیری از عوارض جانبی:** برای جلوگیری از تغییر چیدمان اصلی تیم‌ها در حافظه در طول شبیه‌سازی‌های مکرر، متد رتبه‌بندی گروه‌ها تغییرات را روی یک کپی مستقل از لیست تیم‌ها انجام می‌دهد.

نتیجه گیری و خروجی‌ها

برنامه دارای یک منوی متنی با ۷ گزینه است . شبیه‌سازی ۱۰۰۰ باره به صورت پویا از کاربر تعداد دفعات را دریافت کرده (با اینتر پیش‌فرض ۱۰۰۰ بار) و با تحلیل آماری دقیق، شانس قهرمانی تیم‌ها را استخراج و درصد قهرمآنی آنها را میدهد.

```
================ World Cup Simulator ==========================
1) Loading teams from CSV file
2) Draw groups (Seed Automatic)
3) Run Group Stages and display tables
4) Run tournament (group and knockout stage) and show champion
5) 1000 simulation and championship probability report
6) Display Bracket of Last Simulation
7) Exit
===============================================================
Select an option (1-7): 1
loading 32 teams from the file was completed successfully
```

خروجی های نمونه :

```
----- GROUP STAGE TABLES -----

Group A:
Team            | Pts | GD  | GF  | GA
-----------------------------------
Iran            | 7   | 2   | 5   | 3
Belgium         | 6   | 2   | 4   | 2
USA             | 2   | -1  | 5   | 6
Nigeria         | 1   | -3  | 3   | 6

Group B:
Team            | Pts | GD  | GF  | GA
-----------------------------------
Ecuador         | 7   | 2   | 6   | 4
Mexico          | 3   | 0   | 1   | 1
Brazil          | 2   | -1  | 4   | 5
Senegal         | 2   | -1  | 3   | 4
```

1) گزینه سه منو و نمایش جدول

```
Starting full tournament simulation...

===== Round of 16 =====
Iran 1 - 1 (4-0 pens) Ecuador -> winner : Iran
Morocco 0 - 2 France -> winner : France
Japan 0 - 1 Peru -> winner : Peru
England 2 - 0 Wales -> winner : England
Brazil 1 - 0 Belgium -> winner : Brazil
Germany 1 - 0 Croatia -> winner : Germany
Italy 0 - 1 Netherlands -> winner : Netherlands
Denmark 4 - 5 Sweden -> winner : Sweden

===== Quarterfinals =====
Iran 0 - 1 France -> winner : France
Peru 3 - 1 England -> winner : Peru
Brazil 4 - 0 Germany -> winner : Brazil
Netherlands 2 - 3 Sweden -> winner : Sweden

===== Semifinals =====
France 2 - 1 Peru -> winner : France
Brazil 2 - 0 Sweden -> winner : Brazil

===== Final =====
France 1 - 2 Brazil -> winner : Brazil

🏆 World Cup Champion: Brazil 🏆
```

2) گزینه چهار منو و انجام یک دور شبیه سازی نمایش نتیجه بازی های قهرمان

```
Enter number of simulations (Press Enter for default 1000):

Running 1000 simulations, please wait...
Team`s championship probabilities have been simulated 1000 times
Argentina:  7.90%
France:  7.00%
Brazil:  8.70%
England:  7.70%
Belgium:  6.40%
Croatia:  3.30%
```

3) گزینه پنج منو اجرای1000 باره شبیه سازی و نشان دادن درصد قهرمان