

پوشه football را باز کرده و run.cmd را اجرا کنید. ابعاد زمین بازی به صورت زیر است:



شما باید تابع play داخل پوشه team1 و فایل team1.py را پیاده سازی کنید. شما تیم قرمز هستید. ورودی‌های این تابع شامل اطلاعات بازیکن‌های قرمز، اطلاعات بازیکن‌های آبی، تعداد گل‌های بازیکنان قرمز، تعداد گل‌های بازیکنان آبی، اطلاعات توپ و زمان سپری شده است.

اطلاعات ورودی

اطلاعات مورد نظر را به صورت زیر باید به دست آورید.

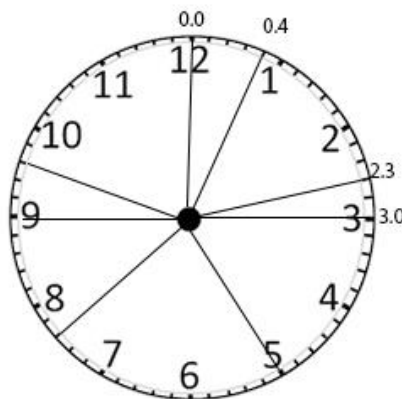
red_players[i]['x']	موقعیت بازیکن i ام
red_players[i]['y']	
blue_players[i]['x']	
blue_players[i]['y']	
red_players[i]['name']	نام بازیکن i ام
blue_players[i]['name']	
red_players[i]['number']	شماره بازیکن i ام !!!
blue_players[i]['number']	
ball['x']	موقعیت توپ
ball['y']	
ball['speed']	سرعت حرکت توپ
ball['direction']	جهت حرکت توپ
ball['owner_color']	توپ دست کدام تیم است
ball['owner_number']	توپ دست بازیکن شماره چند است

به نکات زیر توجه کنید

- موقعیت x و y به صورت زیر است:



- یعنی x و y نقطه وسط زمین برابر صفر است. به سمت راست که حرکت کنیم x زیاد و به سمت چپ حرکت کنیم x کم می شود. به سمت بالا حرکت کنیم y زیاد شده و به سمت پایین حرکت کنیم y کم می شود.
- تعداد بازیکنان برابر با ۶ است. در نتیجه طول لیست `red_players` و `blue_players` برابر با ۶ است.
- شماره هر بازیکن برابر با اندیس قرار گرفتنش در لیست `red_players` و `blue_players` است.
- جهت حرکت توپ یک عدد اعشاری است که قسمت صحیح آن از ۰ تا ۱۱ و قسمت اعشاری آن از ۰ تا ۴ است. این جهت ها با توجه به جهت های ساعتی مشخص می شوند. به شکل زیر توجه کنید.



- مقدار `ball['owner_color']` یکی از سه رشته `red` و `blue` و `white` است. در صورتی که `white` باشد یعنی توپ دست هیچ کدام از تیم ها نیست.

حرکت‌ها

شما در این برنامه باید از ۳ تابع move، grab و kick برای انجام حرکت‌های خود استفاده کنید.

move(decisions, player_number, destination, speed)

با استفاده از تابع move بازیکن‌ها حرکت می‌کنند. ورودی‌های آن به این صورت است:

۱. decisions: باید همیشه همین رو بنویسین و کاری نداشته باشین چیه.
۲. player_number: از جنس عدد صحیح است. نشان دهنده شماره بازیکنی است که می‌خواهید حرکت کند.
۳. destination: از جنس دیکشنری است و حتما باید کلیدهای x و y را داشته باشد و نشان دهنده مقصدی است که می‌خواهید بازیکن به سمت آن حرکت کند.
۴. speed: از جنس عدد صحیحی است و نشان دهنده سرعتی است که می‌خواهید بازیکن حرکت کند و باید عددی کوچکتر یا مساوی ۱۸ باشد. (بر حسب پیکسل)

به مثال زیر توجه کنید:

```
maghsad = {'x': 350, 'y': -100}
move(decisions, 10, maghsad, 18)
```

grab(decisions, player_number)

با استفاده از تابع grab می‌توانید یک بازیکن را صاحب توپ کنید. ورودی‌های آن به این صورت است:

۱. decisions: باید همیشه همین رو بنویسین و کاری نداشته باشین چیه.
۲. player_number: از جنس عدد صحیح است. نشان دهنده شماره بازیکنی است که می‌خواهید صاحب توپ شود.

به نکات زیر دقت کنید.

- تنها در صورتی که فاصله بازیکن با توپ کمتر ۱۸ پیکسل باشد بازیکن ممکن است صاحب توپ شود. البته در بازه‌بان (یعنی کسی که شماره‌اش صفر است) می‌تواند در شعاع ۲۳ پیکسلی grab کند.
- هنگام grab کردن اگر توپ دست شخص دیگری نباشد قطعاً بازیکن مورد نظر صاحب توپ خواهد شد. در غیر این صورت به احتمال ۵۰ درصد ممکن است مالک توپ شود.

kick(decisions, player_number, direction, power)

با استفاده از تابع kick می‌توانید به توپ ضربه بزنید. ورودی‌های آن به این صورت است:

۱. decisions: باید همیشه همین رو بنویسین و کاری نداشته باشین چیه.
۲. player_number: از جنس عدد صحیح است. نشان دهنده شماره بازیکنی است که می‌خواهید به توپ ضربه بزنند.
۳. direction: از جنس اعداد اعشاری است و جهتی است که می‌خواهید توپ حرکت کند. (نحوه جهت دهی ساعتی توضیح داده شد)
۴. از جنس عدد صحیح است و قدرتی است که می‌خواهید به توپ ضربه بزنید و باید کمتر از ۶۰ باشد.

توابع کمکی

توابعی هستند که در برنامه خود می‌توانید از آنها کمک بگیرید.

`get_direction(a, b)`

اگر بخواهید توپ را در جهت خاصی شوت کنید می‌توانید با استفاده از این تابع، جهت مناسب را پیدا کنید. ورودی‌های آن باید از جنس دیکشنری باشند و حتماً کلیدهای `x` و `y` را داشته باشند. به مثال زیر توجه کنید.

```
hadaaf = {'x': 400, 'y': 100}
player = red_players[2]
jahat = get_direction(player, hadaaf)
kick(decisions, 2, jahat, 15)
```

دقت کنید که ترتیب ورودی‌ها مهم است وگرنه جهت برعکس را به دست می‌آورید.

`get_distance(a, b)`

اگر بخواهید فاصله بین دو بازیکن یا فاصله یک بازیکن با توپ و یا فاصله هر چیز دیگری را پیدا کنید، می‌توانید از این تابع استفاده کنید. ورودی‌های آن باید از جنس دیکشنری باشند و حتماً کلیدهای `x` و `y` را داشته باشند. به مثال زیر توجه کنید.

```
player = red_players[2]
print(get_distance(player, ball))
point = {'x': 100, 'y': 200}
print(get_distance(player, point))
```