

PacMan Game

برای پروژه این درس قرار است بازی pacman را شبیه سازی کنید.

بازی pacman چیست؟ پکمن موجودی است که در شهری قرار دارد و هدف او خوردن تمام خوراکیها و فرار کردن از دست دشمنانش میباشد. [اینجا](#) میتونید پکمن بازی کنید. (:

نکات کلی که باید در طول پروژه در نظر بگیرید:

۱. هر قسمت از کد را که میزنید در Git خودتان قرار دهید. (در انتها همه فاز های پروژه ها به همراه لینک گیت خودتان باید سابمیت کنید.)

۲. یکی از هدف های اصلی این پروژه، سرچ کردن است. هر سوالی که در طول برنامه به آن بر میخورید سعی کنید خودتان با سرچ کردن در موتورهای جستوجو جواب آن را پیدا کنید.

این کد را : [برنامه کمکی](#) برید اجرا کنید. منظورمون از نمایش حرکت پکمن مشابه خروجی برنامه کمکی می باشد.

در ادامه فاز های پروژه توضیح داده می شود.

فاز یک : کامپیوتر خودش بازی کند.

ورودی :

آدرس فایل حاوی ماتریس. مثال:

```
/home/kalagh/testcases/testcase01.txt
```

Testcase01.txt :

```
{{'0', '1', '*', '1'},
```

{ '1', '#', '1', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '*', '1' }

در فایل testcase01.txt ماتریسی 4×4 قرار دارد. معانی کاراکتر های آرایه :

کاراکتر 0 (صفر): محل فعلی پکمن

کاراکتر #: محل مانع

کاراکتر *: محل خوراکی

کاراکتر 1: محل عادی که میتوان از آن عبور کرد.

خروجی :

مسیری که پکمن باید طی کند که خوراکی ها را بخورد را بصورت انیمیشن نمایش دهید. دقت کنید که چند نمایش برای خروجی امکان پذیر است. خروجی های نمونه :

خروجی نمونه 1:

{{ '0', '1', '*', '1' },

{ '1', '#', '1', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '*', '1' }

{{ '1', '0', '*', '1' },

{ '1', '#', '1', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '*', '1' }

{{ '1', '1', '0', '1' },

{ '1', '#', '1', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '*', '1' }

{{ '1', '1', '1', '1' },

{ '1', '#', '0', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '*', '1' }

{{ '1', '1', '1', '1' },

{ '1', '#', '1', '1' },

{ '1', '1', '0', '1' },

{ '#', '1', '*', '1' }

{{ '1', '1', '1', '1' },

{ '1', '#', '1', '1' },

{ '1', '1', '1', '1' },

{ '#', '1', '0', '1' }

هر مسیر دیگری که به خوراکی ها برسد مورد قبول است.

توجه: خروجی باید **نمایش حرکت پک من روی صفحه نمایش باشد** و نمونه های بالا برای درک بهتر آورده شده اند. و همچنین در خروجی فقط کاراکتر هایی که در بالا معرفی شدند نمایش داده شوند (0, 1, *, #) نیازی به نمایش آکولاد({}) و ویرگول(,) و کوتیشن(") نمی باشد

مسیرها

• سطح یک

پک من در خانه اول ماتریس قرار گرفته و این مسیر ها فقط شامل غذا می باشد و هیچ مانعی در آن قرار ندارد. و سائز ماتریس همان ۴*۴ می باشد.

• سطح دو

پک من در خانه اول ماتریس قرار گرفته است. در این مسیرها هم غذا هم مانع وجود دارد. سائز ماتریس همان ۴*۴ می باشد.

• سطح سه

پک من در هر جای ماتریس می تواند قرار داشته باشد. مسیر ها شامل غذا و مانع می باشد. سائز ماتریس مشخص نمی باشد.

نکات مهم فاز یک

- پیاده سازی برنامه در صورتی که برای مسیر های سطح یک پاسخگو باشد a نمره
- پیاده سازی برنامه در صورتی که برای مسیر های سطح دو پاسخگو باشد b نمره
- مسیر های سطح سه امتیازی می باشد و c نمره می باشد
- در صورتی که مسیر خروجی برنامه، کوتاه ترین مسیر برای پیمایش همه ی خوراکی ها باشد d نمره امتیازی تعلق میگیرد

فاز دو : کاربر بازی کند.

ورودی:

در این فاز از برنامه ورودی آدرس ماتریسی 5×10 می‌باشد و مشابه فاز یک داده میشود. و همچنین کاربر خودش پکمن را با وارد کردن دکمه های Left، Right، Up، Down به عنوان ورودی حرکت می‌دهد.

خروجی:

- نقشه شهر پکمن با توجه به هر ورودی کاربر نمایش داده شود.
- برنامه تا زمانی ادامه دارد که کاربر 1- به عنوان ورودی بدهد یا تمام خوراکی‌ها توسط پکمن خورده شود.

نمونه ورودی و خروجی

فایل testcase02.txt را در نظر بگیرید:

```
{{'0', '1', '1', '*', '1', '#', '#', '1', '1', '#'},
```

```
{'1', '1', '1', '1', '1', '1', '#', '1', '1', '1'},
```

```
{'1', '1', '1', '*', '1', '1', '1', '1', '1', '1'},
```

```
{'1', '1', '1', '1', '1', '1', '#', '1', '1', '1'},
```

```
{'1', '1', '1', '1', '1', '*', '1', '1', '1', '#'}}}
```

User: /home/kalagh/testcases/testcase01.txt

Comp:

```
{{'0', '1', '1', '*', '1', '#', '#', '1', '1', '#'},
```

```
{'1', '1', '1', '1', '1', '1', '#', '1', '1', '1'},
```

```
{'1', '1', '1', '*', '1', '1', '1', '1', '1', '1'},
```

{'1','1','1','1','1','1','#','1','1','1'},

{'1','1','1','1','1','1','1','1','1','#'}}

User: Right Arrow

Comp:

{{'1','0','1','1','1','#','#','1','1','#'},

{'1','1','1','1','1','1','#','1','1','1'},

{'1','1','1','*','1','1','1','1','1','1'},

{'1','1','1','1','1','1','#','1','1','1'},

{'1','1','1','1','1','1','1','1','1','#'}}

User: Right Arrow

Comp:

{{'1','1','0','*','1','#','#','1','1','#'},

{'1','1','1','1','1','1','#','1','1','1'},

{'1','1','1','*','1','1','1','1','1','1'},

{'1','1','1','1','1','1','#','1','1','1'},

{'1','1','1','1','1','1','1','1','1','#'}}

User: Right Arrow

Comp:

```
{{'1','1','1','0','1','#','#','1','1','#'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','*','1','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','1','1','1','1','1','1','#'}}}
```

Pacman ate food!

User: Right Arrow

Comp:

```
{{'1','1','1','1','0','#','#','1','1','#'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','*','1','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','1','1','1','1','1','1','#'}}}
```

دقت کنید در این مرحله پس از گذشتن از *، مقدار خانه تبدیل به ۱ شد.

User: Right Arrow

Comp:

```
{{'1','1','1','1','0','#','#','1','1','#'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','*','1','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','1','1','1','1','1','1','1','#'}}
```

You can't move to right! There is an obstacle

دقت کنید در این مرحله کاربر با وارد کردن دستور right خطای بالا را گرفته. زیرا در راست مانع قرار داشته است. و نقشه آن عوض نشده است.

User: Down Arrow

Comp:

```
{{'1','1','1','1','1','#','#','1','1','#'},
```

```
{'1','1','1','1','0','1','#','1','1','1'},
```

```
{'1','1','1','*','1','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','1','1','1','1','1','1','1','#'}}
```

User: Down Arrow

Comp:

```
{{'1','1','1','1','1','#','#','1','1','#'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','*','0','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','#','1','1','1'},
```



```
{'1','1','1','1','1','1','1','1','1','1','#'}}
```

User: Left Arrow

Comp:

```
{{'1','1','1','1','1','1','#','#','1','1','1','#'},
```

```
{'1','1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','0','1','1','1','1','1','1'},
```

```
{'1','1','1','1','1','1','1','#','1','1','1'},
```

```
{'1','1','1','1','1','1','1','1','1','1','#'}}
```

Pacman finished eating!

- دقت کنید که کلمات User و Comp قرار نیست در خروجی باشند، صرفاً برای نمایش بهتر مثال نوشته شده اند. خروجی قرار است نمایش حرکت باشد.
- توجه: خروجی برنامه باید مانند فاز یک باشد

نکات مهم فاز دو

- ماتریسی که در آدرس داده شده قرار دارد شامل چندین مانع و خوراکی می‌باشد و سائز آن 10×4 می‌باشد.
- نمره این بخش a می‌باشد.
- در صورت نمایش هر چه زیباتر خروجی با استفاده از کتابخانه graphics نمره امتیازی بسته به خروجی‌تان در نظر گرفته می‌شود.
- **نکته کلی:** اگر برنامه شما قابلیت های بیشتری نسبت به پروژه فعلی داشته باشد، نمره امتیازی متناسب با کار شما در نظر گرفته میشود.

موفق باشید!

