



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده ریاضی و علوم کامپیوتر

گزارش سوم درس هوش مصنوعی (بازی *Pacman*)

نگارش

مهرشاد حسنی

استاد راهنما

دکتر مهدی قطعی

استاد مشاور

استاد بهنام یوسفی مهر

پاییز ۱۴۰۲

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

چکیده

بازی *Pacman* یکی از بازی‌های ویدئویی محبوب است که در آن شخصیت *Pacman* به دنبال خوردن نقاط و دور شدن ارواح است. در این تحقیق، ما قصد داریم *Pacman* را با استفاده از الگوریتم *Minimax* پیاده‌سازی کنیم. الگوریتم *Minimax* یک الگوریتم تصمیم‌گیری در بازی‌ها است که با فرض هوشمندی حریف به بازیکن اجازه می‌دهد بهترین حرکت ممکن را انجام دهد. در این گزارش تحقیقاتی، ابتدا به معرفی بازی *Pacman* و الگوریتم *Minimax* پرداخته و سپس به معرفی پیاده‌سازی می‌پردازیم.

واژه‌های کلیدی:

بازی *Pacman*، الگوریتم *Minimax*، عامل هوشمند، پیاده‌سازی

فهرست مطالب

عنوان	صفحه
۱ معرفی بازی <i>pacman</i>	۱
۱-۱ مقدمه	۲
۲-۱ نحوه بازی	۲
۳-۱ بازی پیاده شده	۲
۲ معرفی الگوریتم <i>Minimax</i>	۳
۱-۲ الگوریتم <i>Minimax</i>	۴
۲-۲ نحوه کار الگوریتم <i>Minimax</i>	۵
۳ پیاده‌سازی بازی <i>Pacman</i>	۷
۱-۳ نحوه پیاده‌سازی	۸
۲-۳ نحوه حرکت روح‌ها	۸
۳-۳ نحوه حرکت <i>Pacman</i>	۸
۴-۳ نحوه محاسبه <i>Score</i>	۹
۵-۳ جمع‌بندی	۹
۱۰ منابع و مراجع	۱۰

فصل اول

معرفی بازی *pacman*

۱-۱ مقدمه

شخصیت پَک-مَن یکی از شناخته‌شده‌ترین و البته بانفوذترین شخصیت‌های تاریخ بازی‌های ویدیویی به حساب می‌آید، شخصیتی که پروفیسور تورو ایواتا به آن جان بخشید تا این موجود زرد کوچک و البته نقطه‌خوار! امروز ۳۵ ساله شود.

اولین نسخه پَک-مَن سال ۱۹۸۰ در کشور ژاپن و برای دستگاه‌های آرکید عرضه شد. وظیفه نشر این بازی را شرکت نامکو (بندای نامکو امروزی) بر عهده داشت. ناشر پَک-مَن در آمریکا نیز شرکت میدوی بود. پَک-مَن از روز اول عرضه تا به امروز محبوب بود و این موضوع یکی از نکات شگفت‌انگیز این بازی است. پَک-مَن اگرچه در روزهای اول فقط به عنوان یک بازی ساده برای دستگاه‌های آرکید عرضه شد و شاید حتی خالق آن یعنی ایواتانی نیز چنین آینده درخشانی را برای این بازی متصور نبود اما با گذشت سال‌ها از عرضه پَک-مَن، کم‌کم پتانسیل‌های موجود در این بازی مشخص شد تا جایی که به یکی از نمادهای کلاسیک بازی‌های ویدیویی تبدیل شد، به یک تعریف از بازی‌های ویدیویی و این قدرت و نفوذ تا جایی پیش رفت که به یک پدیده اجتماعی تبدیل شد. [۱]

۲-۱ نحوه بازی

ماموریت اصلی در بازی *Pacman*، جمع‌آوری تمام نقاط غذای موجود در ماز است. همچنین، بازیکن باید از ارواح مختلف (*Inky*, *Pinky*, *Blinky* و *Clyde*) فرار کند. با برخورد با روح‌ها، یکی از جان‌های خود را از دست می‌دهید و پس از از دست دادن تمام جان‌ها، بازی را می‌بازید. هرچند بعد از رسیدن به امتیاز ۱۰۰۰۰، بازی به صورت خودکار به شما یک جان خواهد داد. [۱]

۳-۱ بازی پیاده شده

در این گزارش تحقیقاتی، نسخه ساده شده این بازی پیاده شده‌است. در این گزارش، پَک‌من فقط یک جان دارد و با برخورد با روح‌ها جان خود را از دست می‌دهد. در این بازی تمامی خانه‌ها دارای نقطه هستند و پَک‌من باید با استفاده از الگوریتم *Minimax* عملکرد خود را بیشینه کند و عملکرد روح‌ها را کمینه کند و سپس به ادامه بازی بپردازد. در این بازی، ابتدا پَک‌من یک حرکت می‌کند و سپس روح‌ها به صورت رندوم یک حرکت می‌کنند.

فصل دوم

معرفی الگوریتم *Minimax*

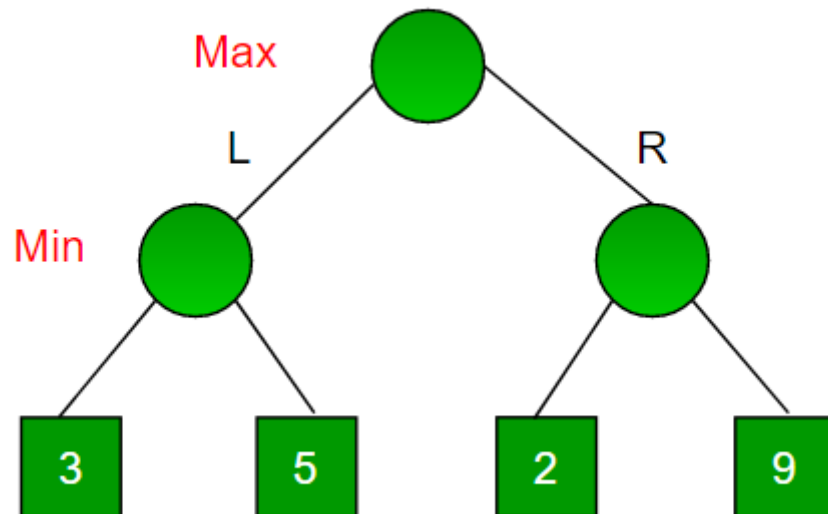
در این قسمت، به معرفی الگوریتم *Minimax* می‌پردازیم.

۱-۲ الگوریتم *Minimax*

در واقع این الگوریتم سعی می‌کند با کمترین هزینه (مینیمم) تصمیمی بگیرد که بیشترین سود (ماکزیمم) را نصیب ما کند. با فرض این که حریف به شکل منطقی بازی کند از این الگوریتم می‌توان در بازی‌های دونفره نوبتی به شکل گسترده استفاده کرد. برای مثال، در بازی‌های مثل بازی دوز (*Tic-Tac-Toe*) یا منطق شطرنج یا تخته نرد و امثالهم می‌توان از این الگوریتم استفاده کرد.

در الگوریتم *Minimax* به دو بازیکن حداکثرکننده و حداقل‌کننده گفته می‌شود، حداکثر کننده سعی می‌کند که بالاترین امتیاز ممکن را بگیرد و در مقابل حداقل‌کننده سعی می‌کند که کمترین امتیاز ممکن را به دست آورد در حالت معین اگر حداکثرکننده برتری داشته باشد امتیاز مثبت خواهد بود و اگر حداقل‌کننده برتری داشته باشد امتیاز منفی خواهد بود.

این مقادیر و محاسبه توسط روش‌های اکتشافی (پیش‌بینی حرکت‌های ممکن در بازی) که برای هر بازی (منظور نوع بازی است) منحصر به فرد خواهد بود محاسبه می‌شود. [۲]

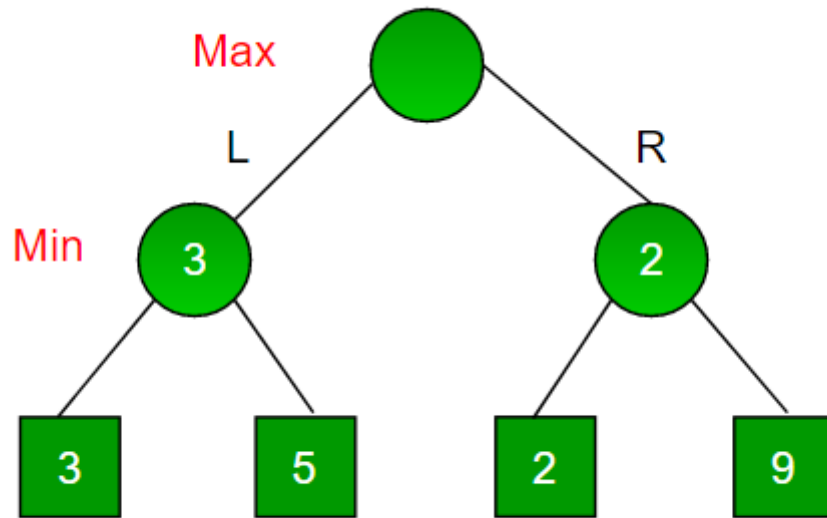
۲-۲ نحوه کار الگوریتم *Minimax*

با استفاده از این الگوریتم تصور کنید درخت حرکات ما به شکل بالا است. اکنون حداکثرکننده *Maximizer* دو حالت دارد:

۱. شما به عنوان یک بازیکن حداکثر کننده حرکت *L* را انتخاب می کنید، آنگاه حریف شما دو انتخاب خواهد داشت یا ۳ را انتخاب کند یا این که ۵ را انتخاب کند که با توجه به نقش حداقل کننده ۳ را انتخاب خواهد کرد.

۲. شما به عنوان یک بازیکن حداکثر کننده حرکت *R* را انتخاب می کنید، آنگاه حریف شما دو انتخاب خواهد داشت یا ۲ را انتخاب کند یا این که ۹ را انتخاب کند که با توجه به نقش حداقل کننده مسلماً ۲ را انتخاب خواهد کرد.

باتوجه به شرایط بالا که توضیح دادیم منطقی تر است که شما به عنوان حداکثرکننده حرکت به سمت چپ (*L*) را انتخاب کنید که در نهایت در بدترین حالت ممکن ۳ امتیاز خواهید گرفت. در نهایت درخت بازی به شکل زیر تصویر خواهد شد. [۳]

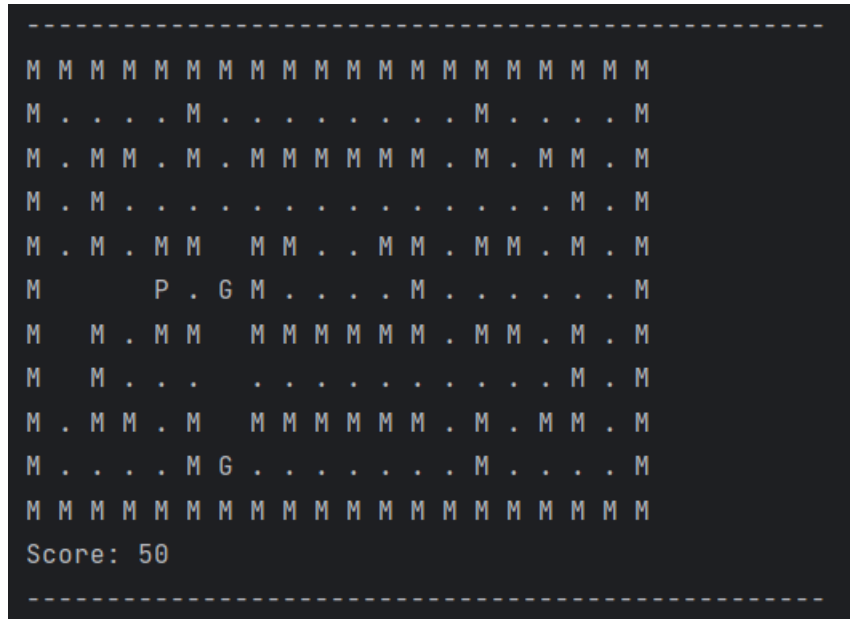


فصل سوم

پیاده‌سازی بازی *Pacman*

۱-۳ نحوه پیاده‌سازی

در این بخش به بررسی نحوه پیاده‌سازی بازی Pacman می‌پردازیم. در این پیاده‌سازی، موانع با حرف M ، روح‌ها با حرف G و پکمن با حرف P نمایش داده شده‌است.



۲-۳ نحوه حرکت روح‌ها

در این بازی طبق صورت سوال، روح‌ها دارای حرکت رندوم هستند. حرکت رندوم روح‌ها به شکل زیر پیاده‌سازی شدند:

```
direction = random.randint(0, 3)
is_changed, x, y = self.move(x1, y1, direction)
while not is_changed:
    is_changed, x, y = self.move(x1, y1, random.randint(0, 3))
```

۳-۳ نحوه حرکت Pacman

در این بخش، پکمن با استفاده از الگوریتم *Minimax* سعی در بهینه کردن حرکت خود و کمینه کردن حرکت روح دارد. بنابراین، با استفاده از الگوریتم *BFS* نزدیک‌ترین نقطه را پیدا می‌کند و در صورتی

که در آن نقطه روحی وجود نداشت به آن جا می‌رود. همچنین در صورتی که نقطه بعدی بیشتر از یک خانه با خانه کنونی فاصله داشت، در هر مرحله یک بار حرکت می‌کند تا به آن خانه برسد.

۴-۳ نحوه محاسبه Score

در این پیاده‌سازی، به ازای هر حرکتی که نقطه‌ای خورده نشود یک امتیاز کم می‌شود و به ازای هر حرکتی که یک نقطه خورده شود ۱۰ امتیاز اضافه می‌گردد. همچنین اگر با روح برخورد انجام شود، ۵۰۰ امتیاز از Score کاهش می‌یابد.
در این تابع داریم:

```
def update_score(self, score, x, y):
    if self.ground[y][x] == 1:
        score += 10
    else:
        score -= 1
    return score
```

شما می‌توانید از طریق [این لینک](#) به کد این سوال دسترسی داشته باشید.

۵-۳ جمع‌بندی

در این تحقیق، ما به پیاده‌سازی بازی Pacman با استفاده از الگوریتم Minimax پرداختیم. از طریق مدل‌سازی بازی، تعریف تابع Minimax و پیاده‌سازی الگوریتم آن، ما توانستیم Pacman را به یک بازیکن هوشمند تبدیل کنیم که بهینه‌ترین حرکت را در هر مرحله انجام می‌دهد.
این پروژه نه تنها به ما کمک کرد تا با الگوریتم Minimax و روند تصمیم‌گیری در بازی‌ها آشنا شویم، بلکه ما را با مفاهیم مهمی همچون مدل‌سازی بازی، تابع ارزیابی و انتخاب بهترین حرکت آشنا کرد. پس از معرفی بازی Pacman و الگوریتم Minimax، به معرفی پیاده‌سازی بازی با استفاده از این الگوریتم پرداختیم. سپس قسمت‌هایی از کد را مشاهده کردیم. کد کامل این بازی در [این لینک](#) قابل مشاهده است.

منابع و مراجع

- [1] Ayari, Arash. everything about pacman. Zoomg, 2015.
- [2] Zeus. implementing minimax algorithm. Sisoog.
- [3] L, Akshay. minimax algorithm in game theory. GeeksforGeeks, 2022.