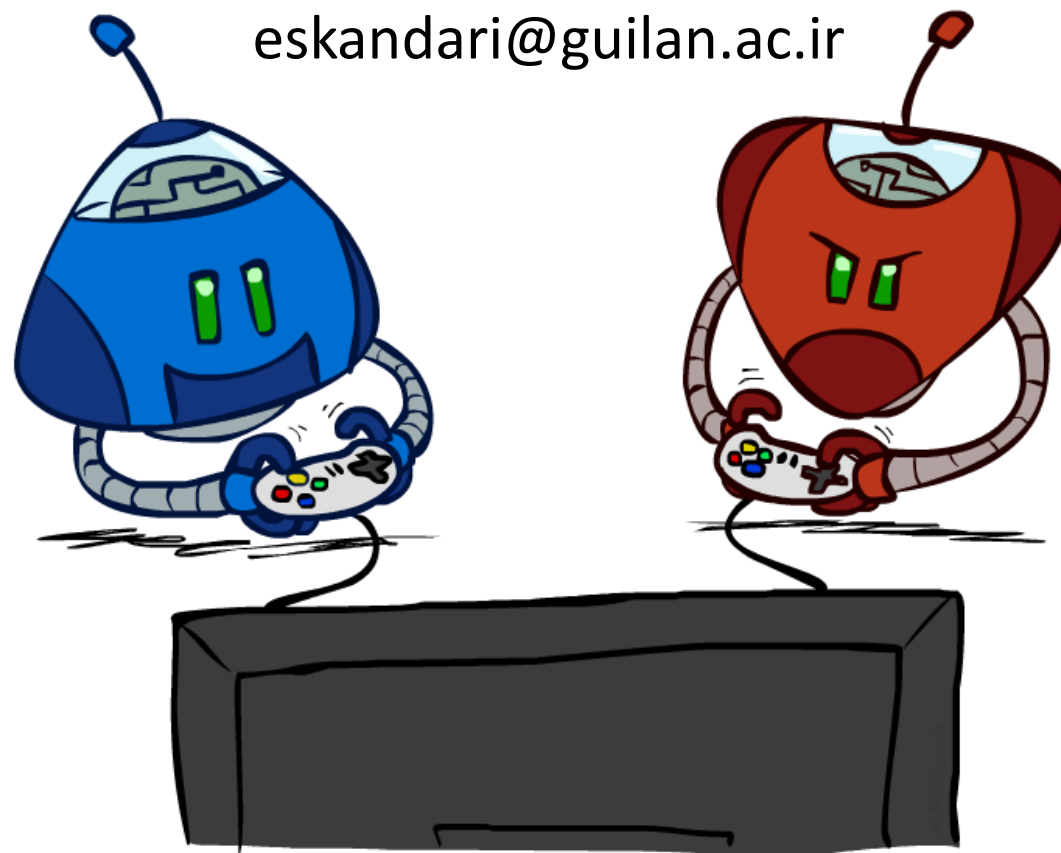


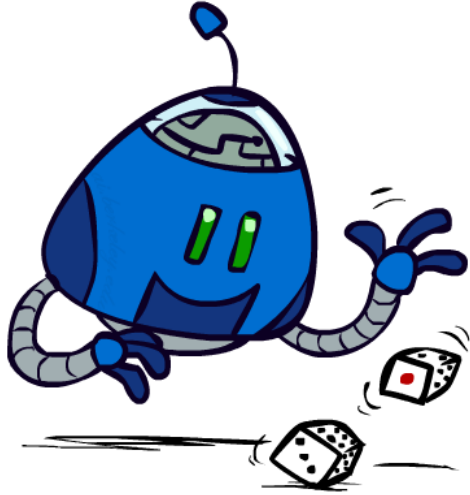
هوش مصنوعی (جستجو در حضور عامل های دیگر- بخش دوم)

صادق اسکندری - دانشکده علوم ریاضی، گروه علوم کامپیوتر

eskandari@guilan.ac.ir



بازی در حضور شانس



تا اینجای کار، با بازی هایی مواجه بودیم که اولاً در آنها محیط قطعی است و دوماً رقیب به شکل ایده آل بازی می کند.

در برقی از بازیها ممکن است شانس نیز دخیل باشد:

Expectimax

○ بازیکن رقیب کاملاً ایده آل نیست و گاهی حرکت های تصادفی انجام میدهد.

Expectiminimax

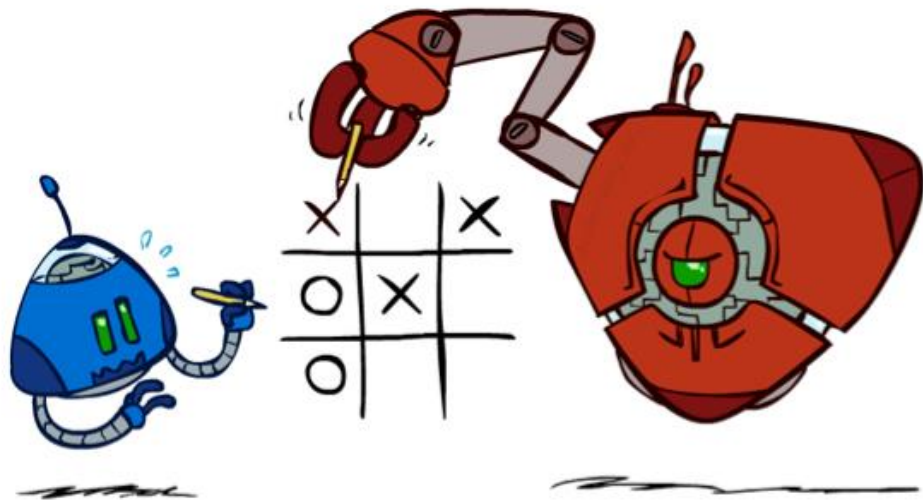
○ ماهیت بازی همراه با شانس است. مانند بازی تخته نرد (Backgammon)

MDP

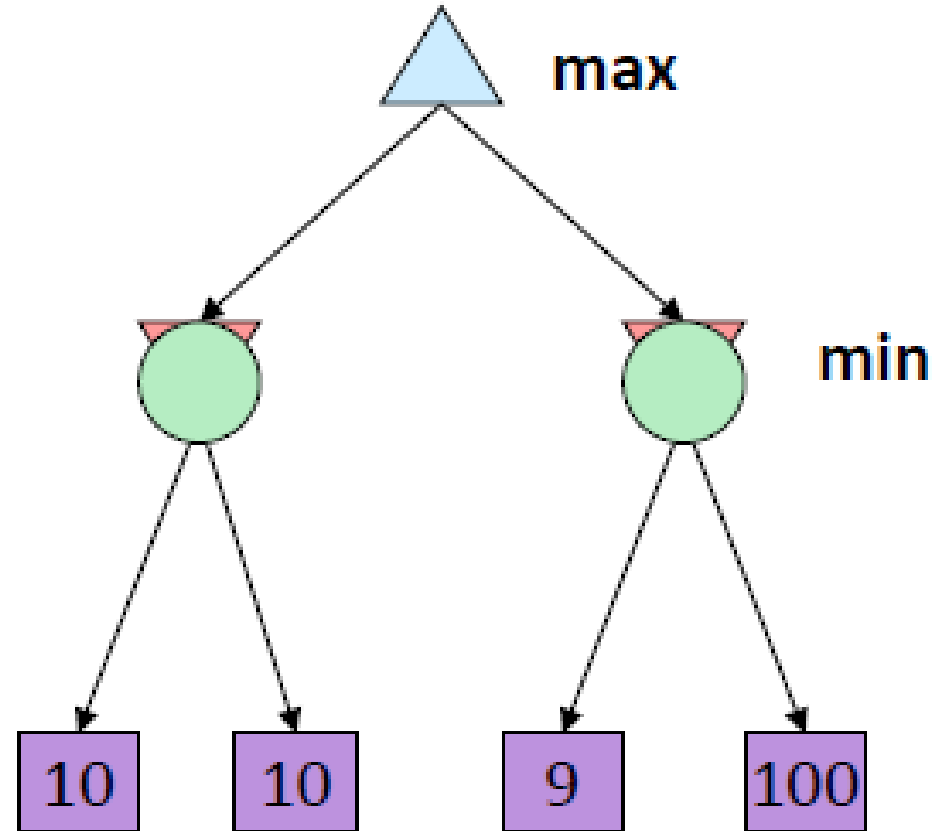
○ خود عمل ممکن است با شکست مواجه شود. رباتی را در نظر بگیرید که تصمیم به حرکت به سمت اتاق مجاور دارد ولی به دلیل لغزنده بودن زمین، احتمال سر خوردن پرفهای آن وجود دارد.

الگوریتم Expectimax

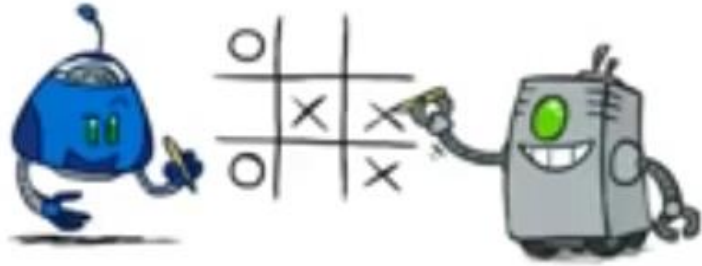
یادآوری: الگوریتم MiniMax



فرض اولیه: MIN بهترین حرکت خود را انجام می دهد.



الگوریتم Expectimax



○ در الگوریتم Expectimax فرض بر این است که رقیب (MIN) خیلی ایده آل نیست و گاهی حرکت های اشتباه هم انجام میدهد.

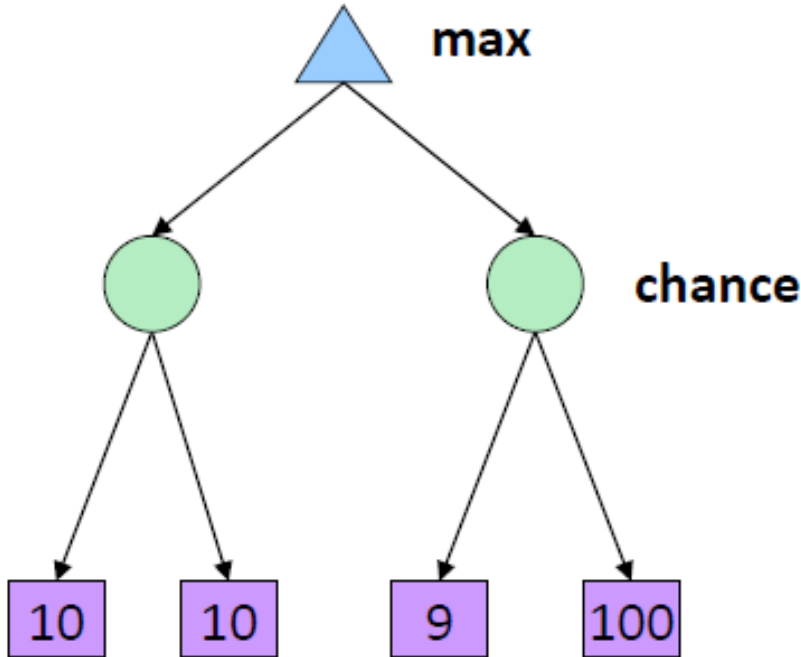
○ در این الگوریتم، به جای در نظر گرفتن بدترین حالت برای MIN، حالت میانگین را در نظر می گیریم. (امید ریاضی: *Expected Value*)

○ در Expectimax:

گره های MAX همانند الگوریتم MiniMax عمل می کنند.

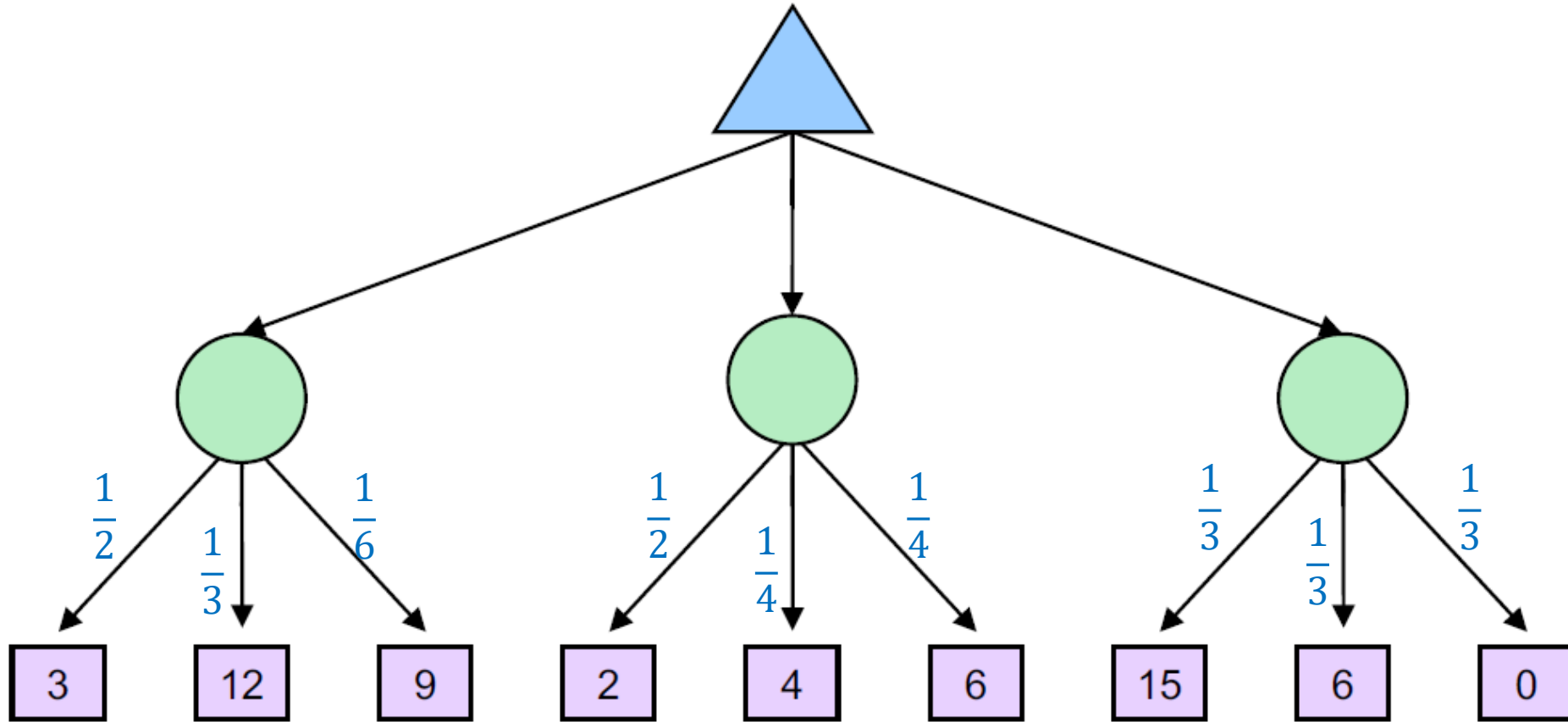
به جای گره های MIN از گره هایی تحت عنوان Chance استفاده می شود.

این گره ها، به جای در نظر گرفتن کمترین مقدار فرزندان خود، میانگین وزنی (امید) گره های فرزند را مناسبه می کنند.



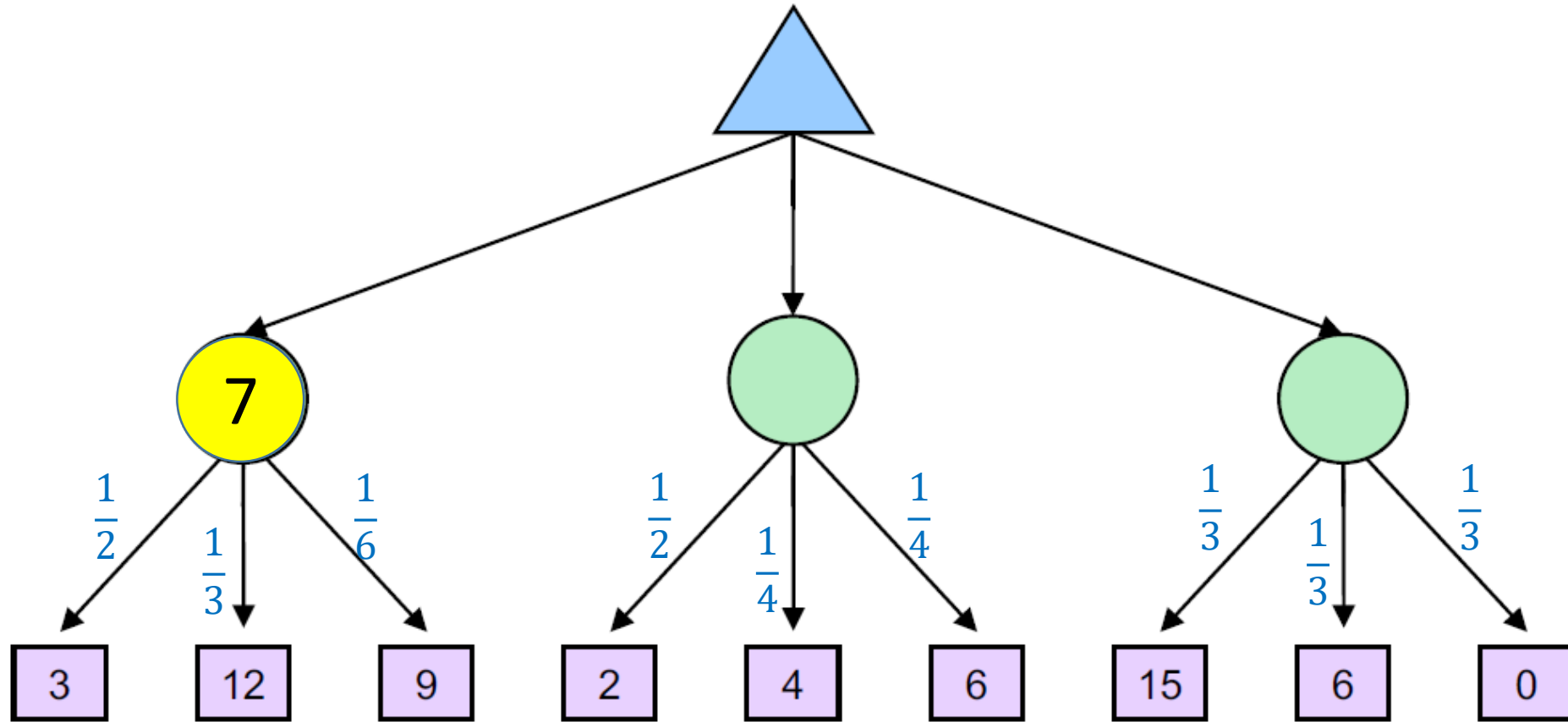
Expectimax الگوریتم

مثال



الگوریتم Expectimax

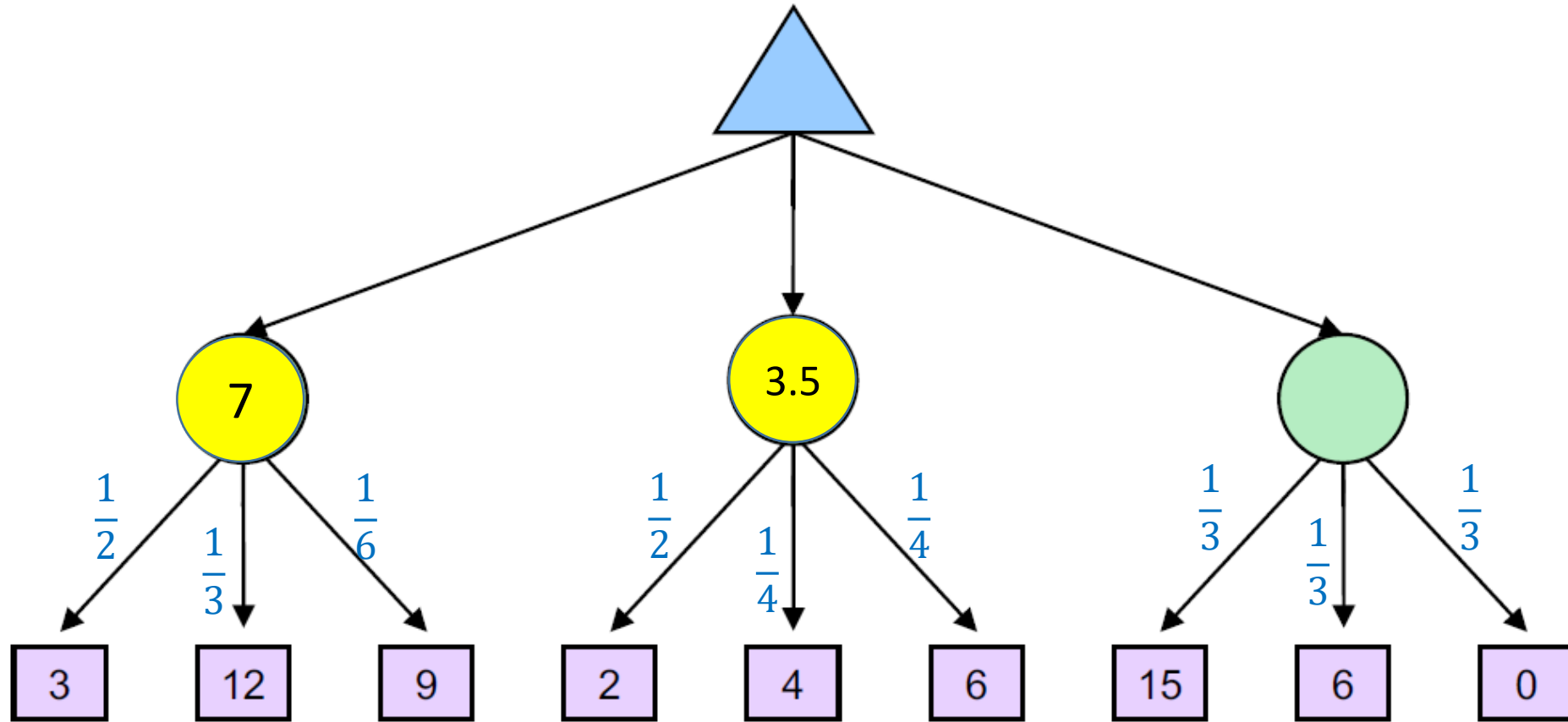
مثال



$$\frac{1}{2} \times 3 + \frac{1}{3} \times 12 + \frac{1}{6} \times 9 = 7$$

الگوریتم Expectimax

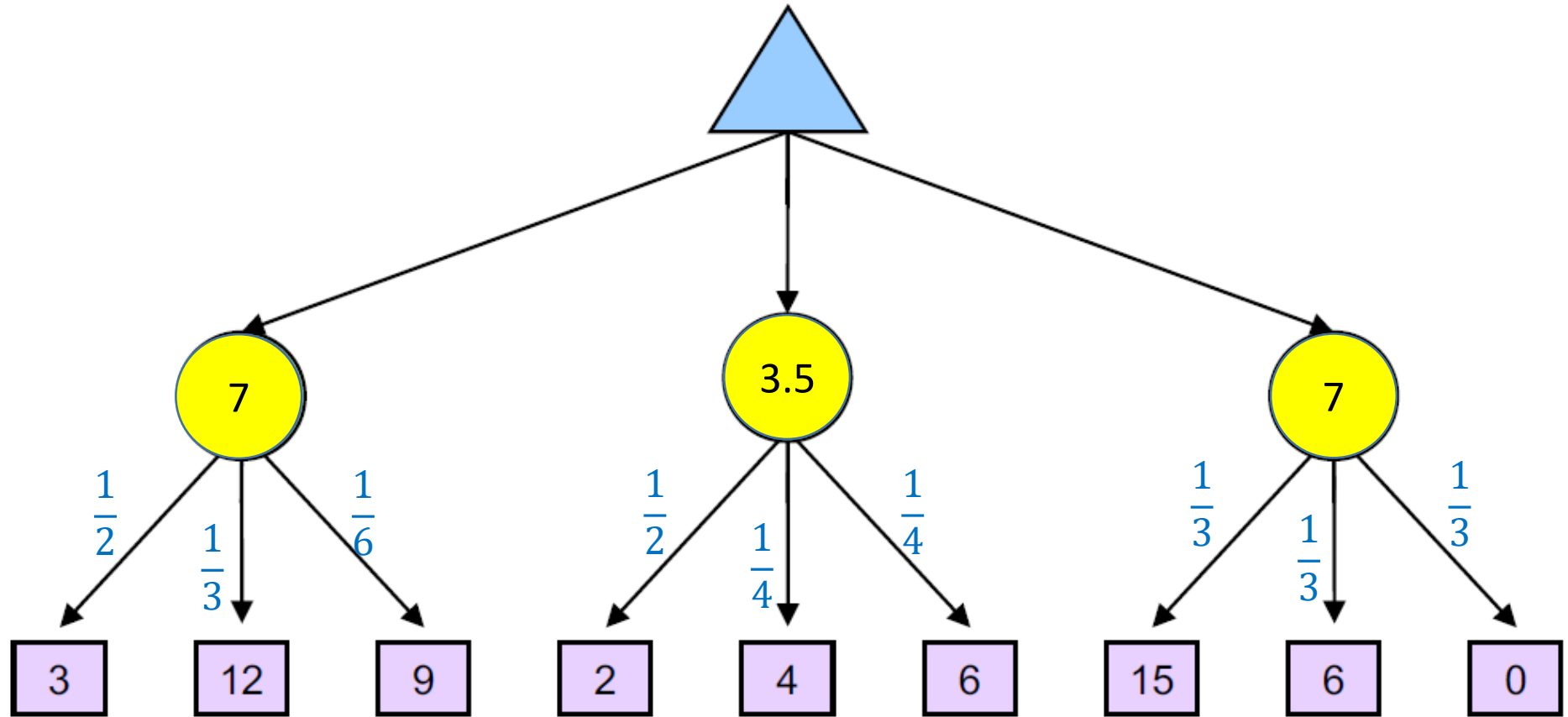
مثال



$$\frac{1}{2} \times 2 + \frac{1}{4} \times 4 + \frac{1}{4} \times 6 = 3.5$$

الگوریتم Expectimax

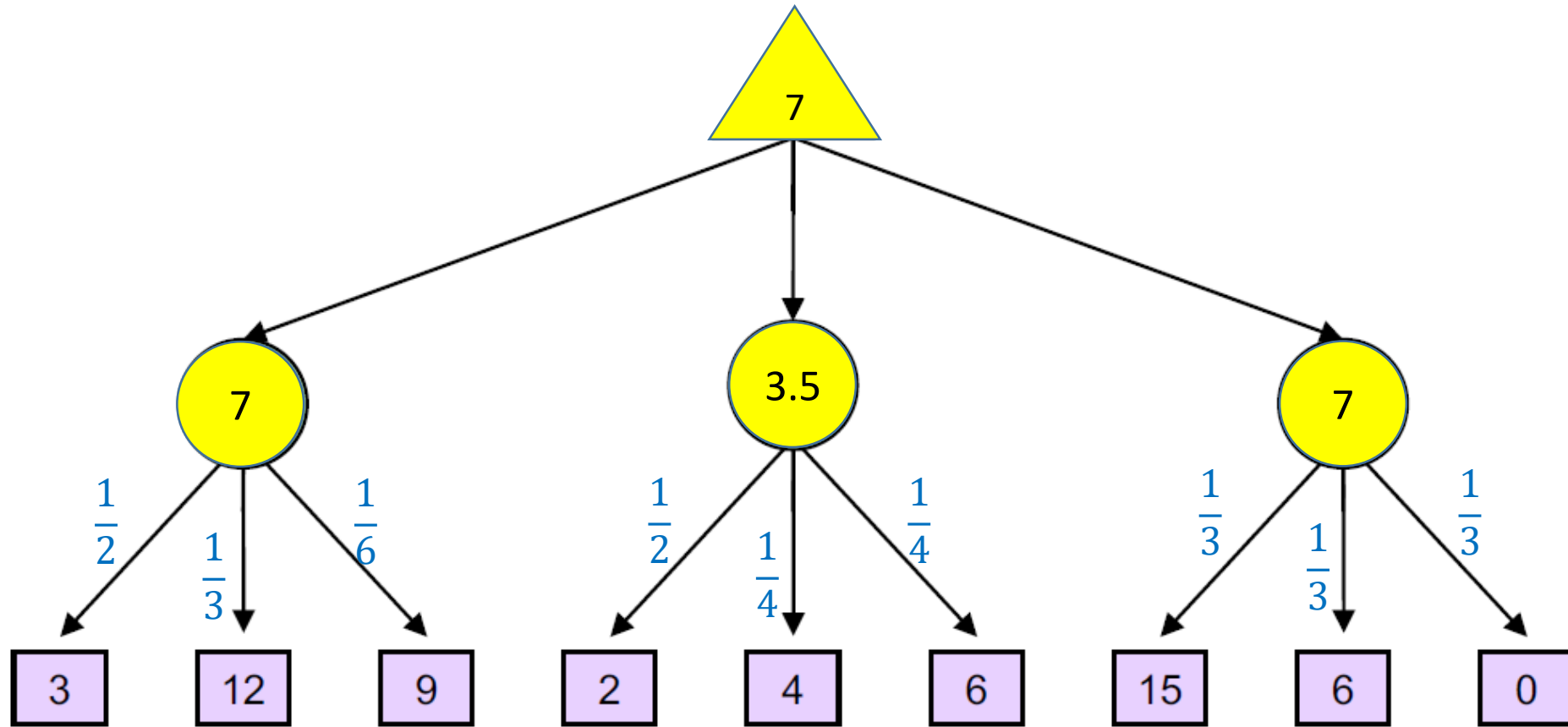
مثال



$$\frac{1}{3} \times 15 + \frac{1}{3} \times 6 + \frac{1}{3} \times 0 = 7$$

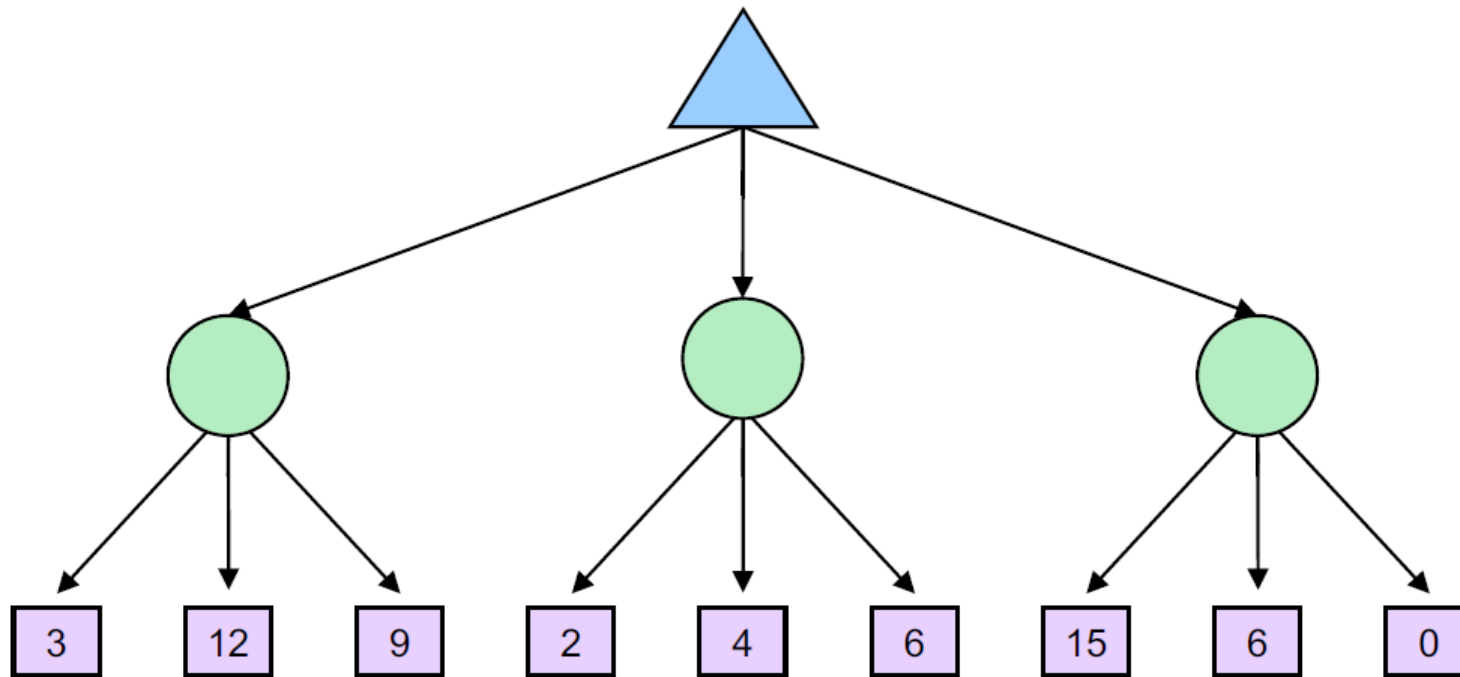
الگوریتم Expectimax

مثال



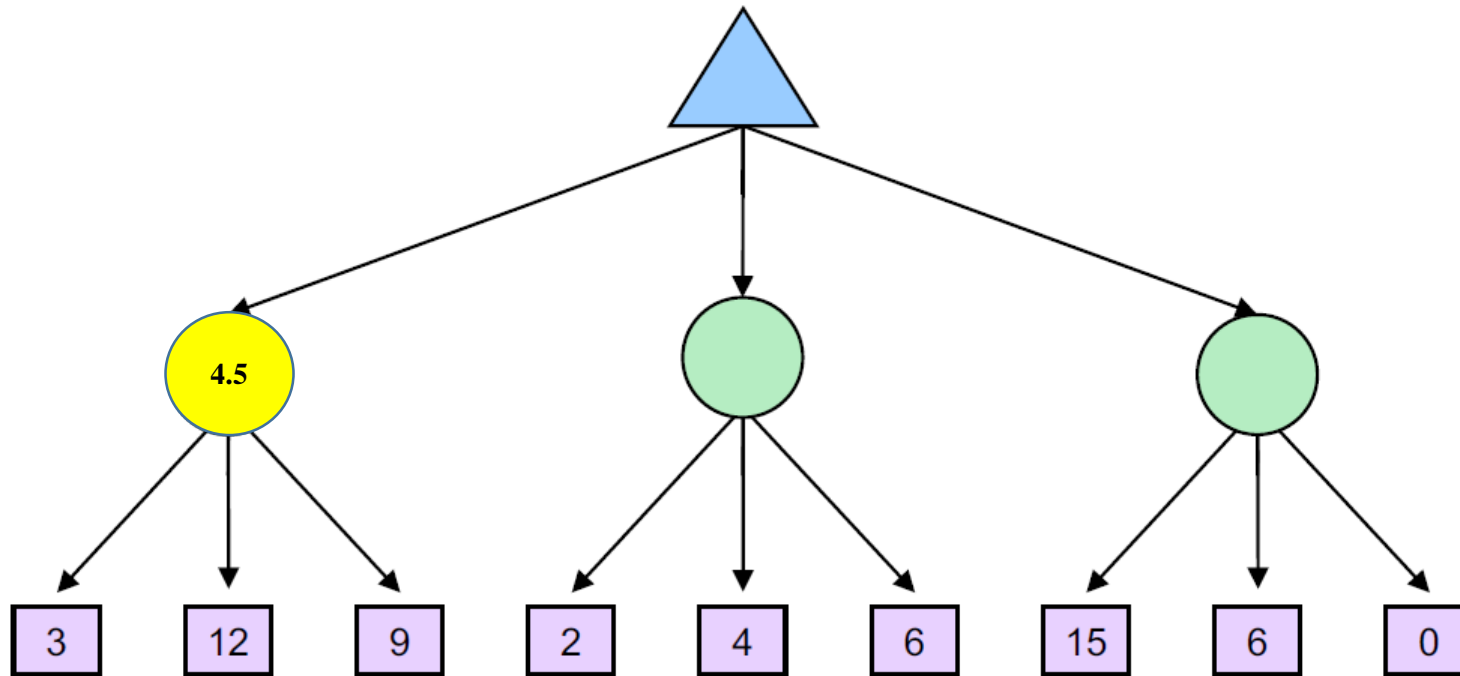
الگوریتم Expectimax

مثال: فرض کنید که در حال بازی با رقیبی هستید که در ۷۰ درصد تصمیمات خود، از الگوریتم **MiniMax** و در ۳۰ درصد دیگر، از تصمیمات کاملاً تصادفی استفاده می کند. مقادیر **Expectimax** را برای درخت زیر مشخص کنید.



الگوریتم Expectimax

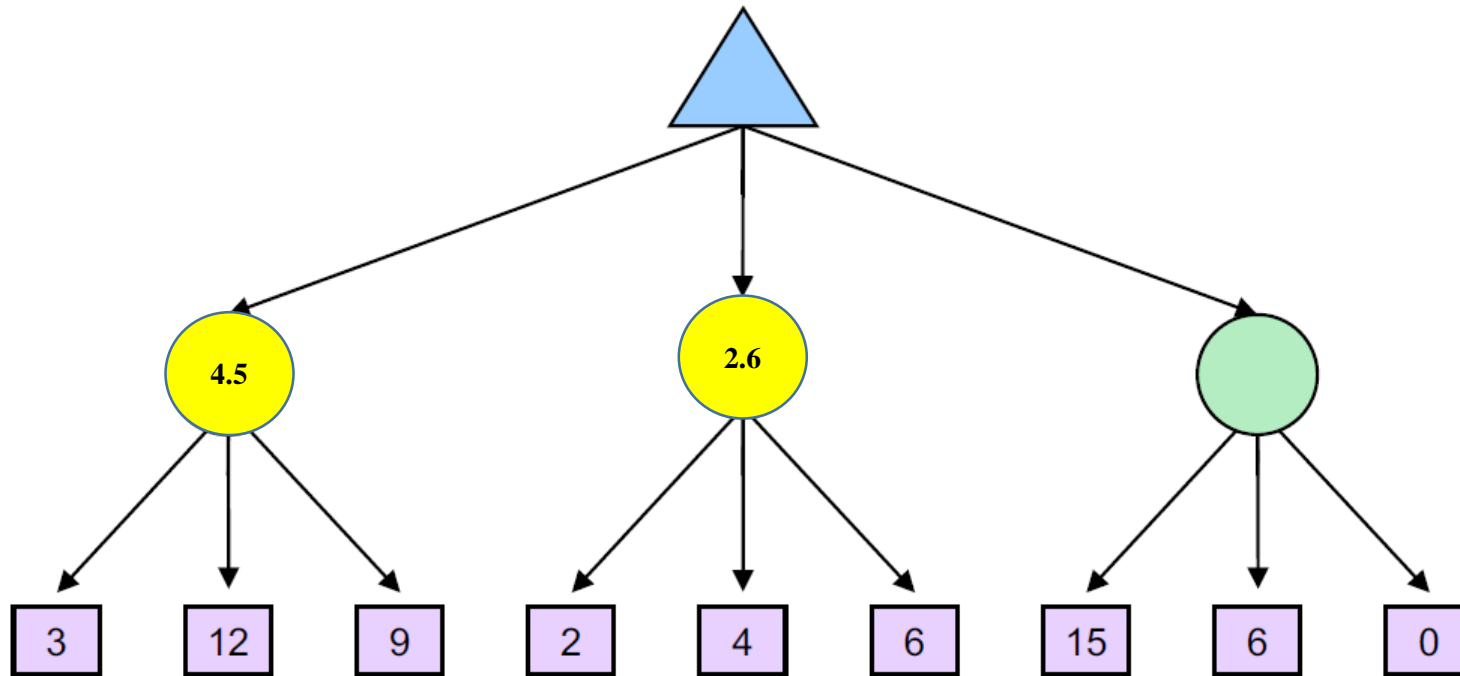
مثال: فرض کنید که در حال بازی با رقیبی هستید که در ۷۰ درصد تصمیمات خود، از الگوریتم MiniMax و در ۳۰ درصد دیگر، از تصمیمات کاملاً تصادفی استفاده می کند. مقادیر Expectimax را برای درخت زیر مشخص کنید.



$$\frac{7}{10} \times 3 + \frac{3}{10} \times \left(\frac{1}{3} \times 12 + \frac{1}{3} \times 3 + \frac{1}{3} \times 9 \right) = 2.1 + 2.4 = 4.5$$

الگوریتم Expectimax

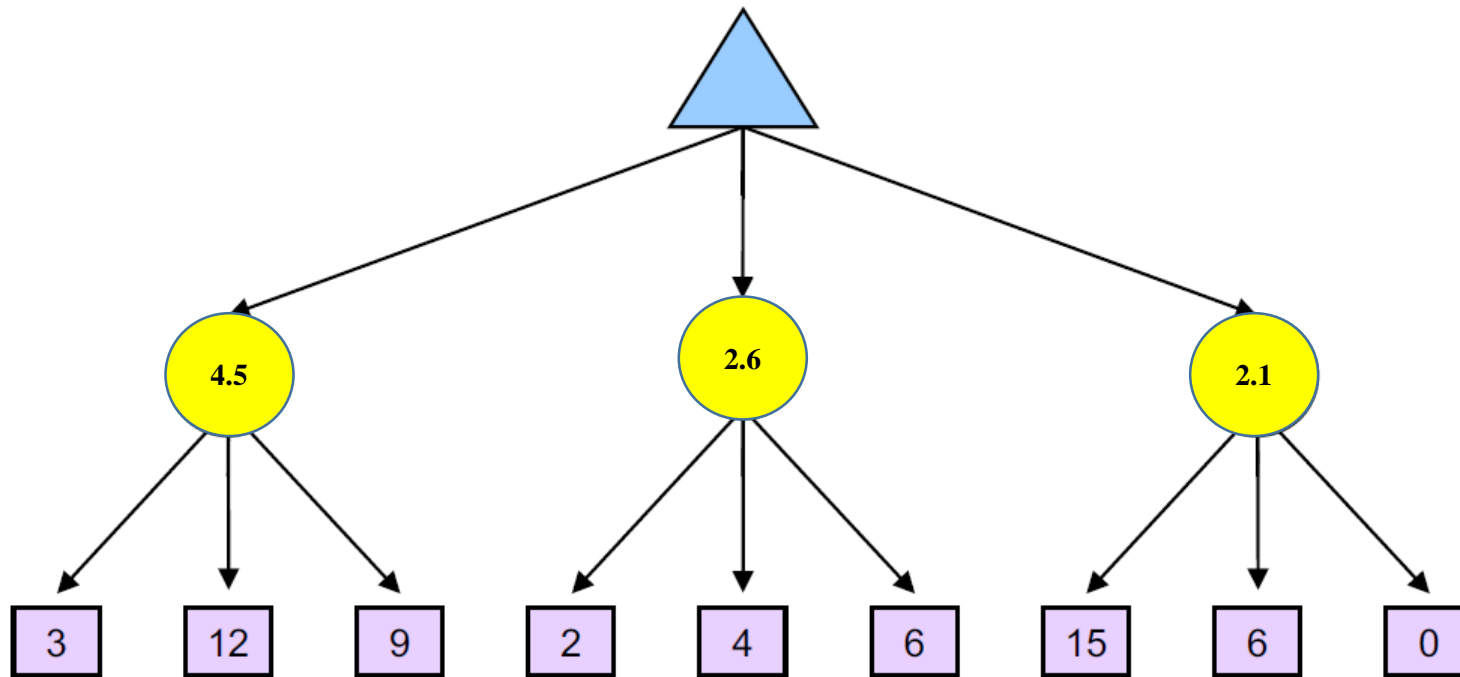
مثال: فرض کنید که در حال بازی با رقیبی هستید که در ۷۰ درصد تصمیمات خود، از الگوریتم MiniMax و در ۳۰ درصد دیگر، از تصمیمات کاملاً تصادفی استفاده می کند. مقادیر Expectimax را برای درخت زیر مشخص کنید.



$$\frac{7}{10} \times 2 + \frac{3}{10} \times \left(\frac{1}{3} \times 2 + \frac{1}{3} \times 4 + \frac{1}{3} \times 6 \right) = 1.4 + 1.2 = 2.6$$

الگوریتم Expectimax

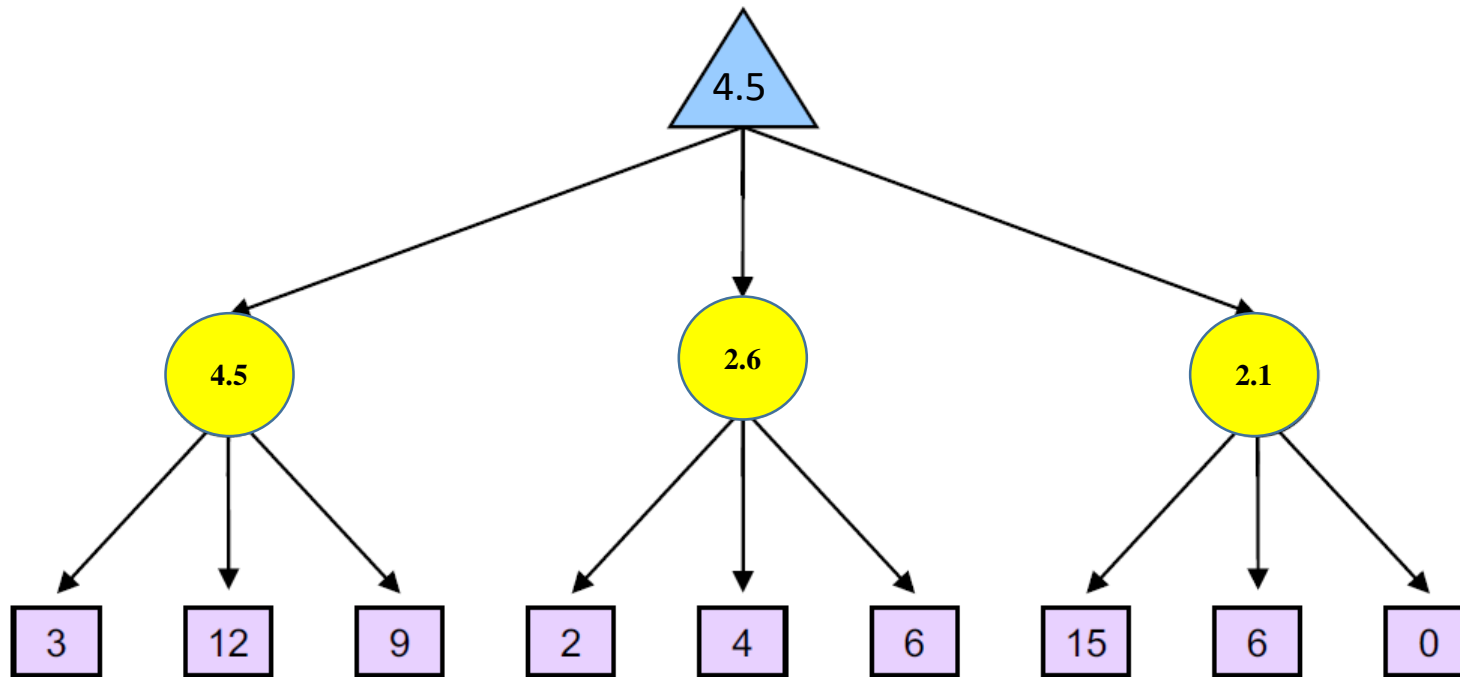
مثال: فرض کنید که در حال بازی با رقیبی هستید که در ۷۰ درصد تصمیمات خود، از الگوریتم MiniMax و در ۳۰ درصد دیگر، از تصمیمات کاملاً تصادفی استفاده می کند. مقادیر Expectimax را برای درخت زیر مشخص کنید.



$$\frac{7}{10} \times 0 + \frac{3}{10} \times \left(\frac{1}{3} \times 15 + \frac{1}{3} \times 6 + \frac{1}{3} \times 0 \right) = 0 + 2.1 = 2.1$$

الگوریتم Expectimax

مثال: فرض کنید که در حال بازی با رقیبی هستید که در ۷۰ درصد تصمیمات خود، از الگوریتم MiniMax و در ۳۰ درصد دیگر، از تصمیمات کاملاً تصادفی استفاده می کند. مقادیر Expectimax را برای درخت زیر مشخص کنید.



الغوريتم Expectimax

```
def value(state):
```

```
    if the state is a terminal state: return the state's utility  
    if the next agent is MAX: return max-value(state)  
    if the next agent is EXP: return exp-value(state)
```

```
def max-value(state):
```

```
    initialize v =  $-\infty$   
    for each successor of state:  
        v = max(v, value(successor))  
    return v
```

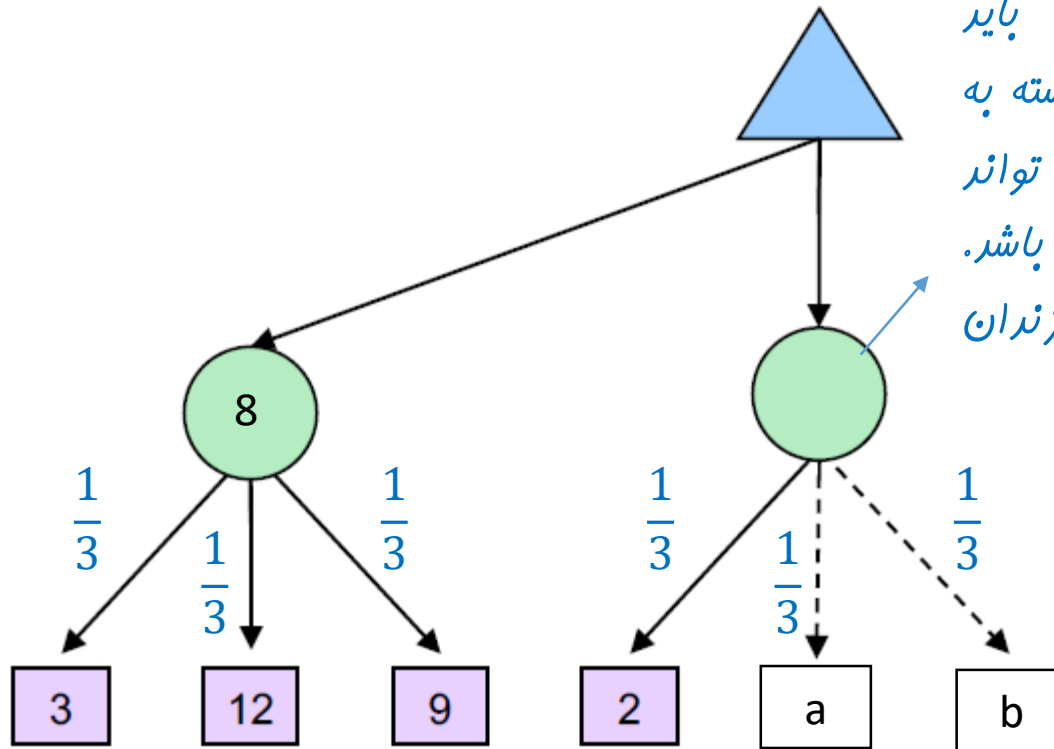
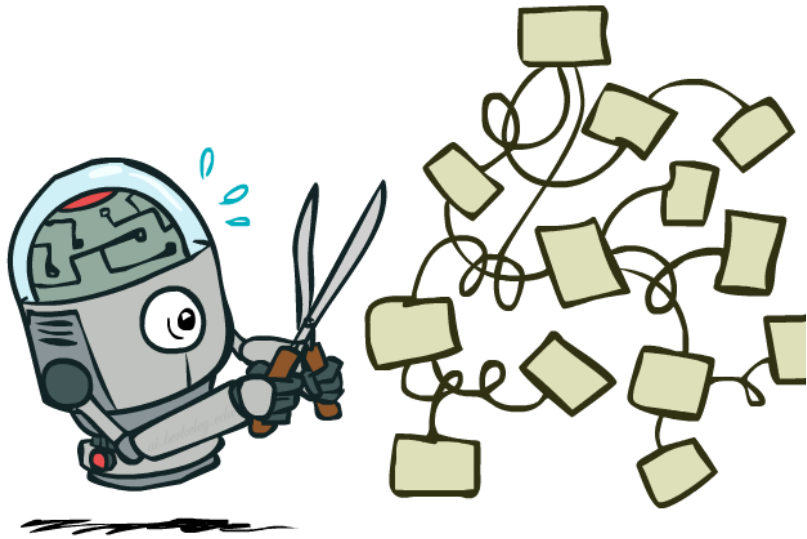
```
def exp-value(state):
```

```
    initialize v = 0  
    for each successor of state:  
        p = probability(successor)  
        v += p * value(successor)  
    return v
```

الگوریتم Expectimax

سوال: آیا در الگوریتم Expectimax می توان از هرس آلفا-بتا استفاده کرد؟

جواب: خیر



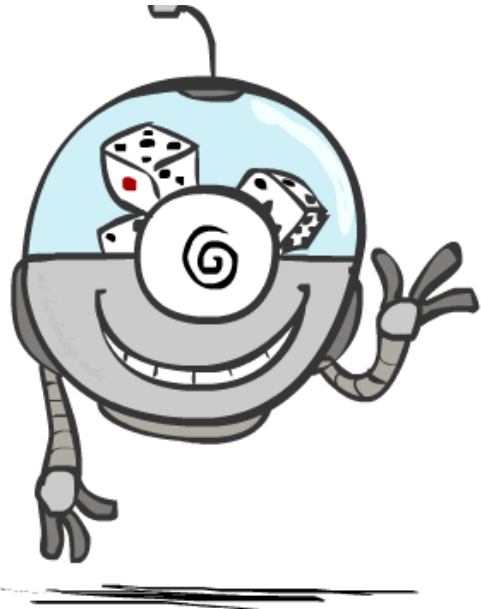
برای مناسبه مقدار این گره، باید میانگین فرزندان مناسبه شود. بسته به مقادیر a و b ، مقدار این گره می تواند کوچکتر از ۸ و یا بزرگتر از ۸ باشد. بنابراین، نیاز به تمامی فرزندان داریم.

الگوریتم ExpectiMiniMax

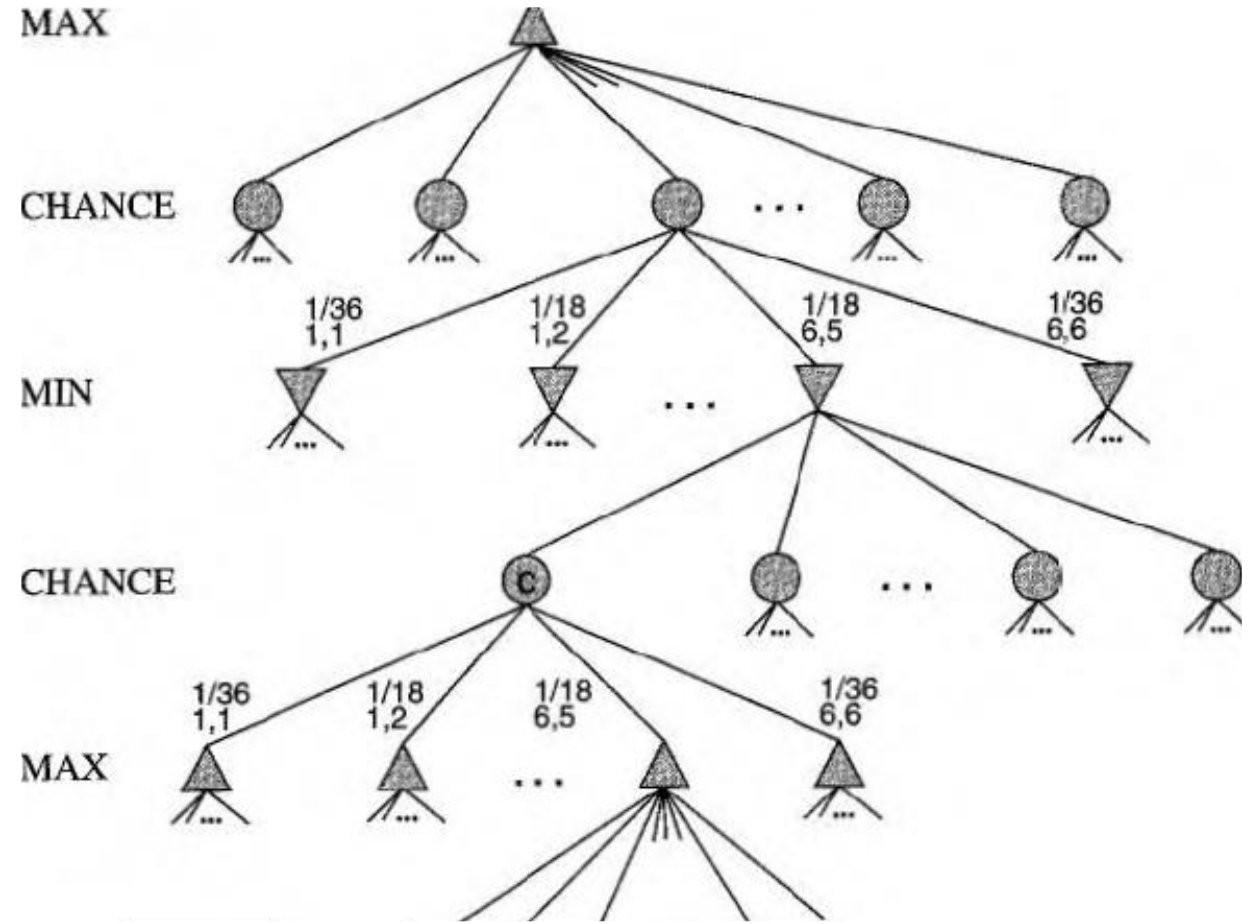
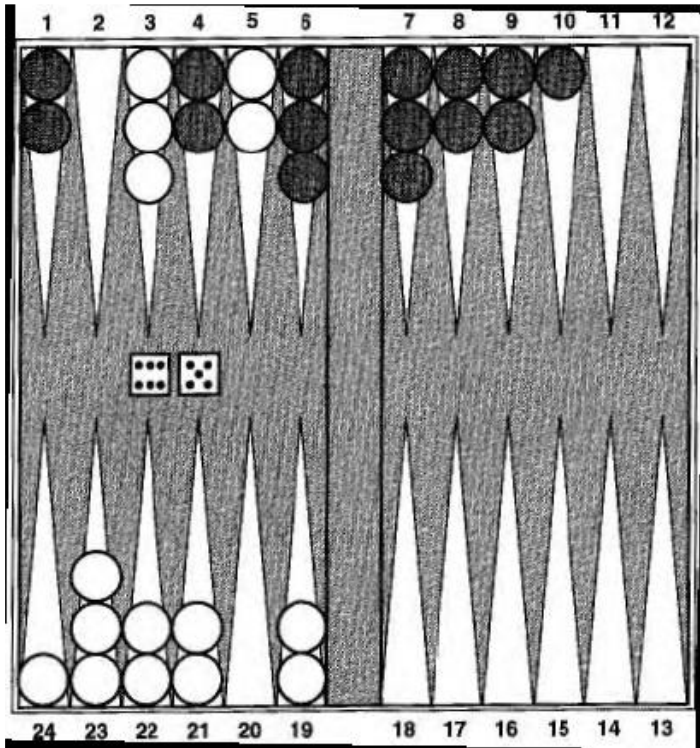
○ در بازی هایی که ماهیتاً دارای عنصر شانس هستند (مانند تخته نرد)، علاوه بر MAX و MIN ، محیط نیز به عنوان یک بازیکن تصادفی بعد از هر یک از دو بازیکن عمل می کند.

○ در این الگوریتم، از سه نوع گره استفاده می شود: MAX ، MIN و $Chance$.

- گره های MIN و MAX همانند الگوریتم $MiniMax$ عمل می کنند.
- مابین هر دو لایه MIN و MAX یک لایه $Chance$ قرار می گیرد.
- گره های $Chance$ در الگوریتم $ExpectiMiniMax$ همانند این گره ها در الگوریتم $Expectimax$ عمل می کنند.



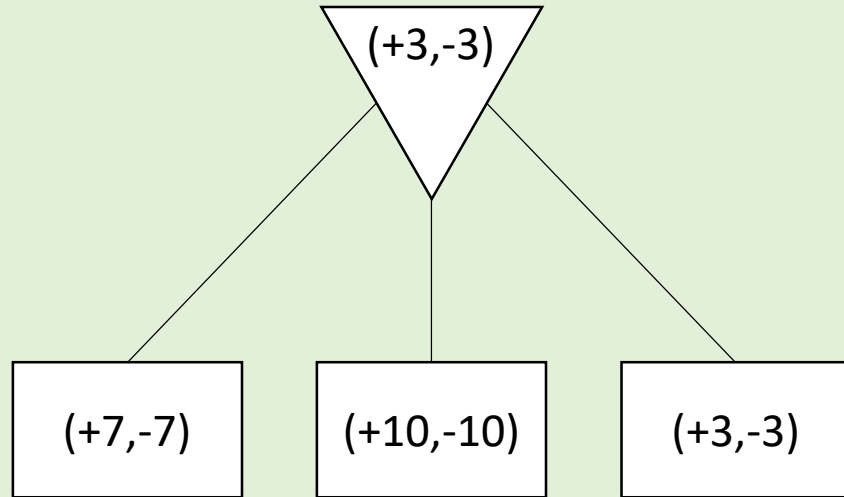
ExpectiMiniMax الگوریتم



انواع دیگر بازیها: بازیهایی که Zero-Sum نیستند

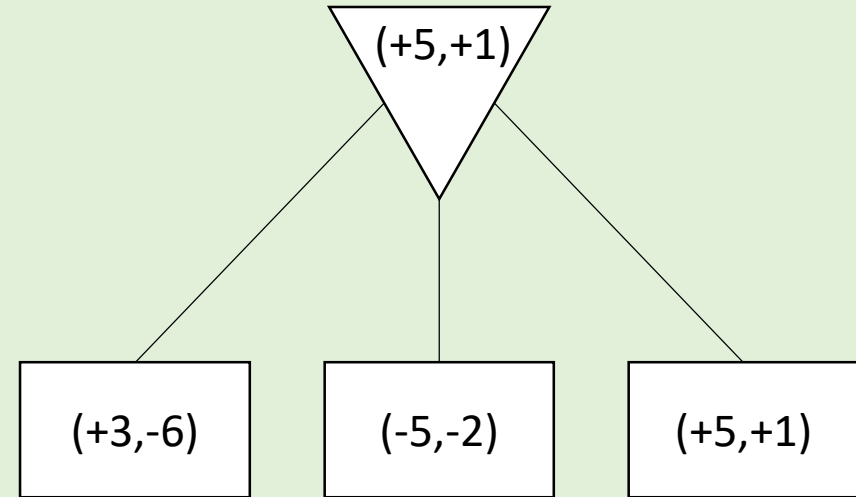
می توان الگوریتم MiniMax را به بازی های دونفره ای که Zero-Sum نیستند نیز بسط داد.

MIN



Zero-Sum

MIN

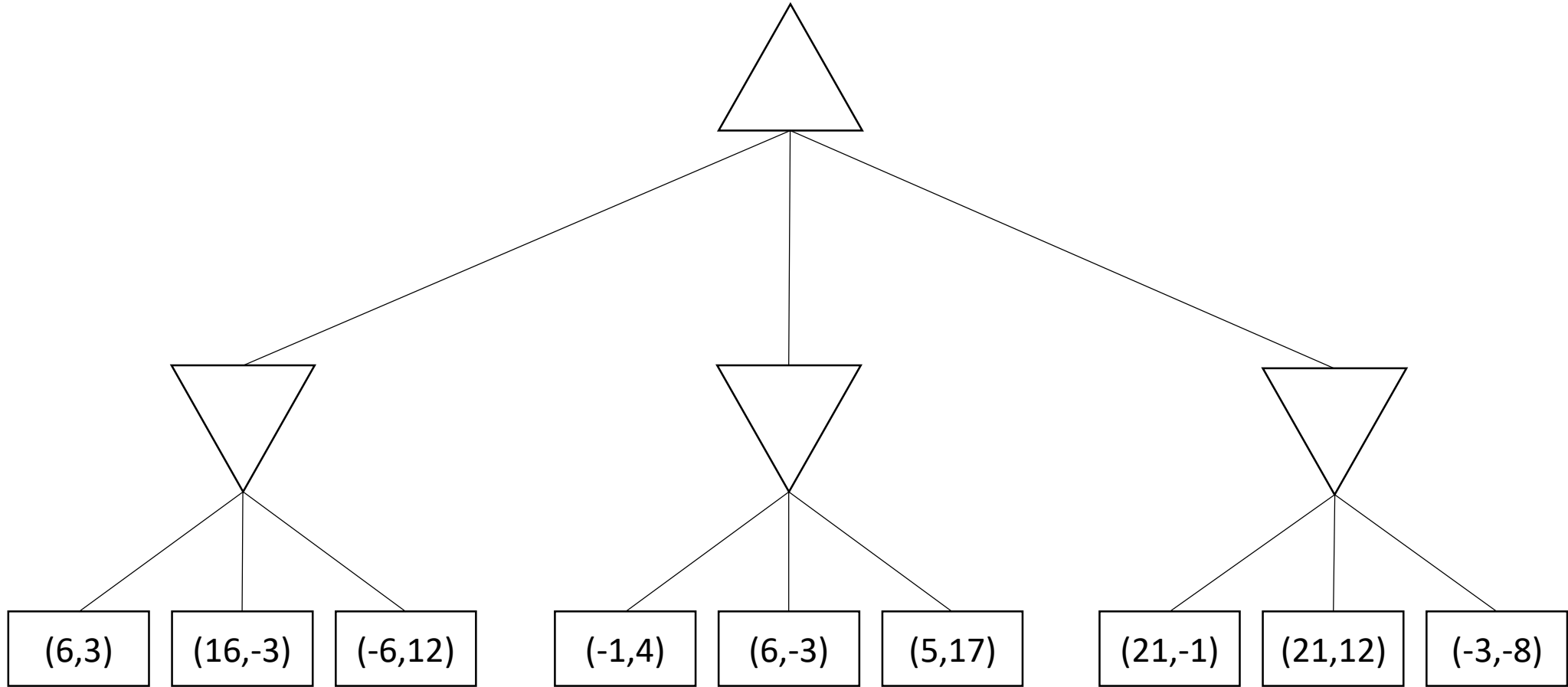


Non Zero-Sum

مقادیر سودمندی به شکل دوتایی نمایش داده می شوند و هر بازیکن (MAX و MIN) سعی می کنند تا دوتایی را انتخاب کنند که سودمندی متناظر در آن بیشترین مقدار باشد.

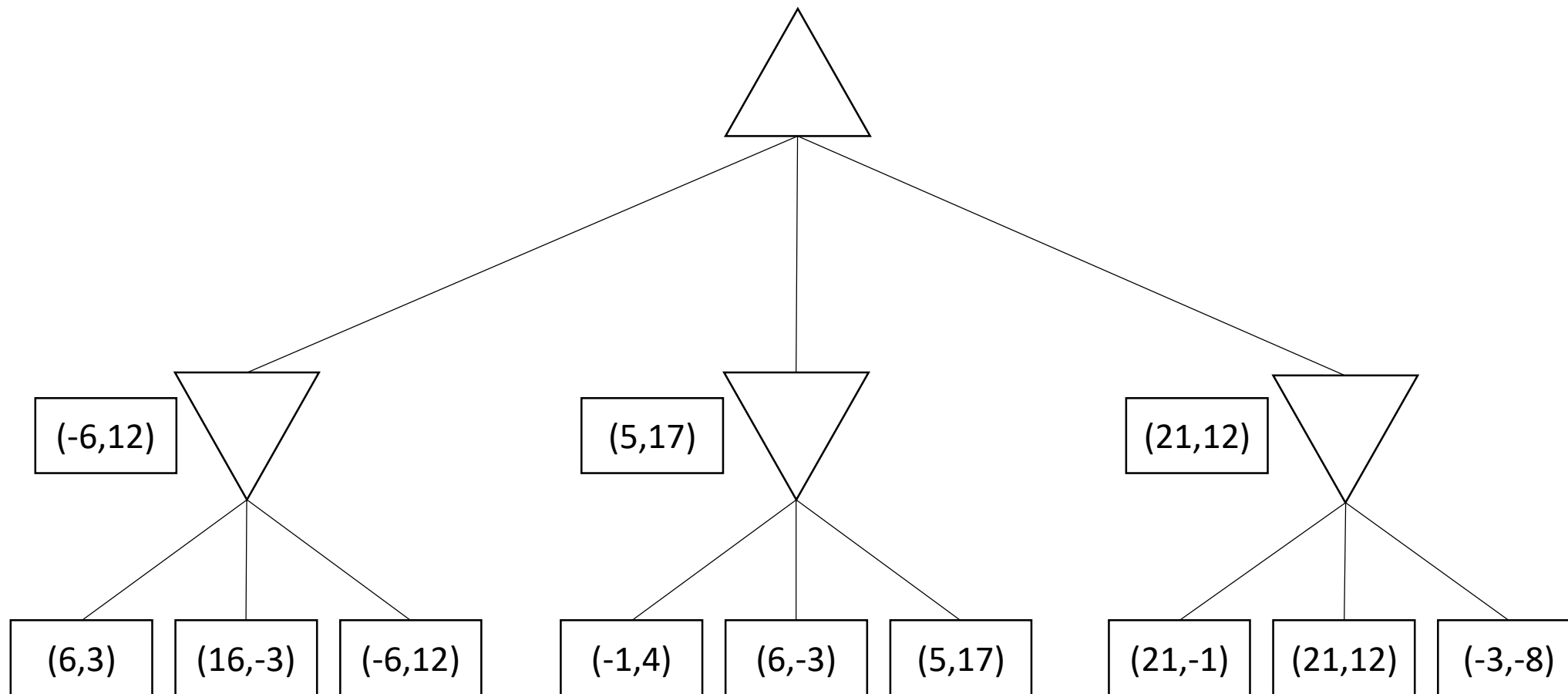
انواع دیگر بازیها: بازیهایی که Zero-Sum نیستند

مثال:



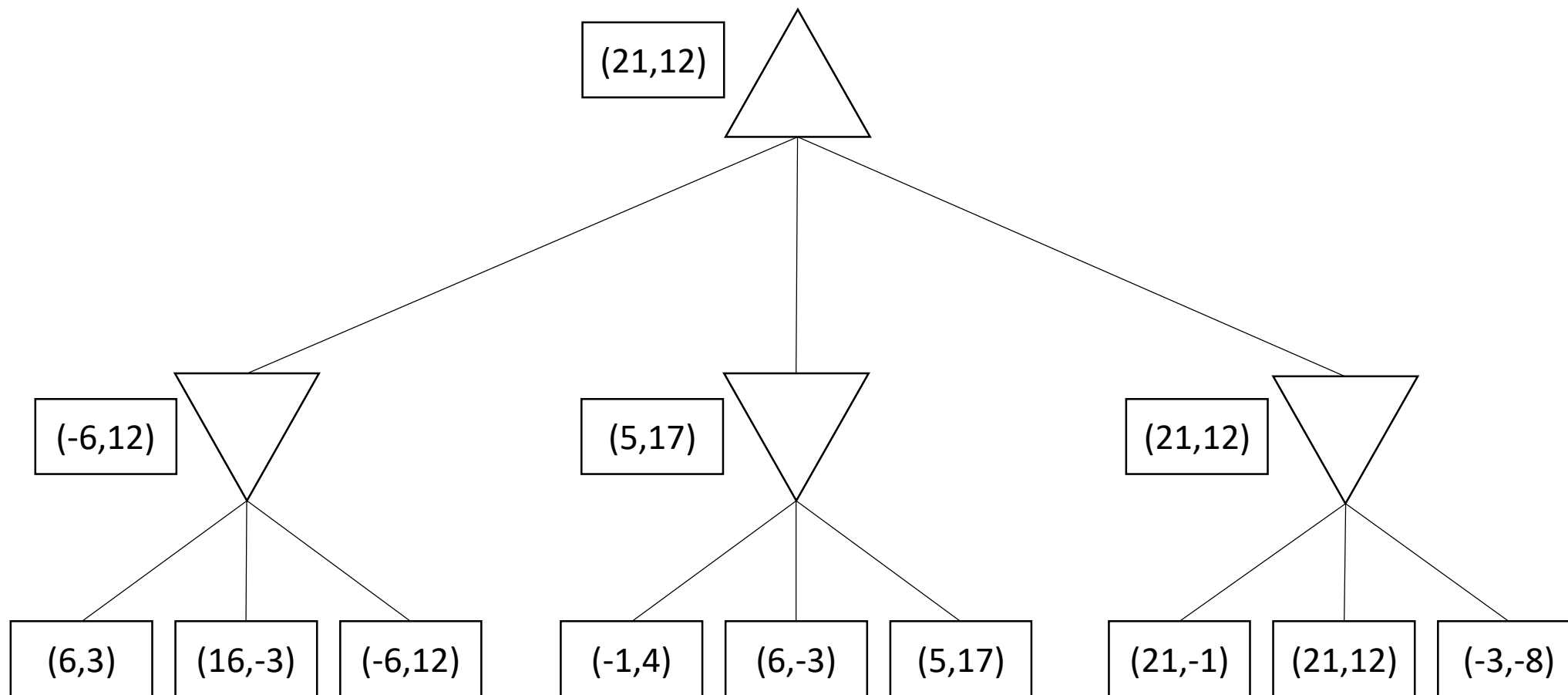
انواع دیگر بازیها: بازیهایی که Zero-Sum نیستند

مثال:



انواع دیگر بازیها: بازیهای که Zero-Sum نیستند

مثال:



انواع دیگر بازیها: بازیهایی که بیش از دو بازیکن دارند

به شیوه مشابهی می توان الگوریتم **MiniMax** را به بازیهای چند بازیکنی نیز تعمیم داد.

