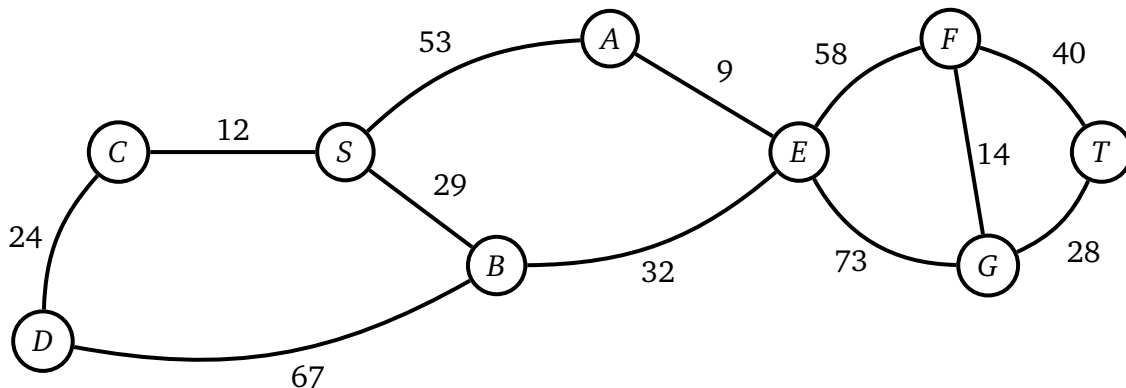


Aufgabe 2.3

(1,5 Punkte)

Die folgende vereinfachte Landkarte ist gegeben:

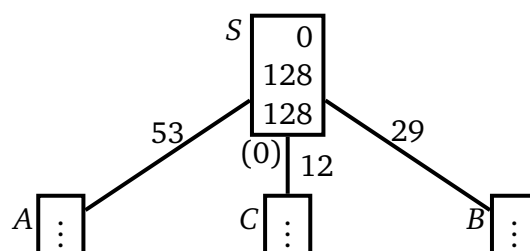


	S	A	B	C	D	E	F	G	T
$h(u, T)$	128	75	104	139	160	90	37	27	0

Lösen Sie die folgenden Aufgaben:

- Berechnen Sie mit Hilfe des A^* -Algorithmus den kürzesten Weg von S nach T . Sie dürfen, ohne dies explizit zu prüfen, annehmen, dass die Heuristik monoton ist.
- Nachdem einige Bauarbeiten zwischen den Orten G und T abgeschlossen wurden, reduzieren sich die Fahrtkosten auf dieser Strecke zu $c([G, T]) = 23$. Außerdem reduzieren sich die heuristischen Kosten von G nach T zu $h(G, T) = 3$. Wieso ist die so entstandene Heuristik nicht mehr monoton?
- Berechnen Sie mit Hilfe des A^* -Algorithmus und den Änderungen aus (b) den kürzesten Weg von S nach T . Es genügt, wenn sie mit dem Knoten E starten. Kopieren Sie hierzu die Werte für E aus der Teilaufgabe (a).

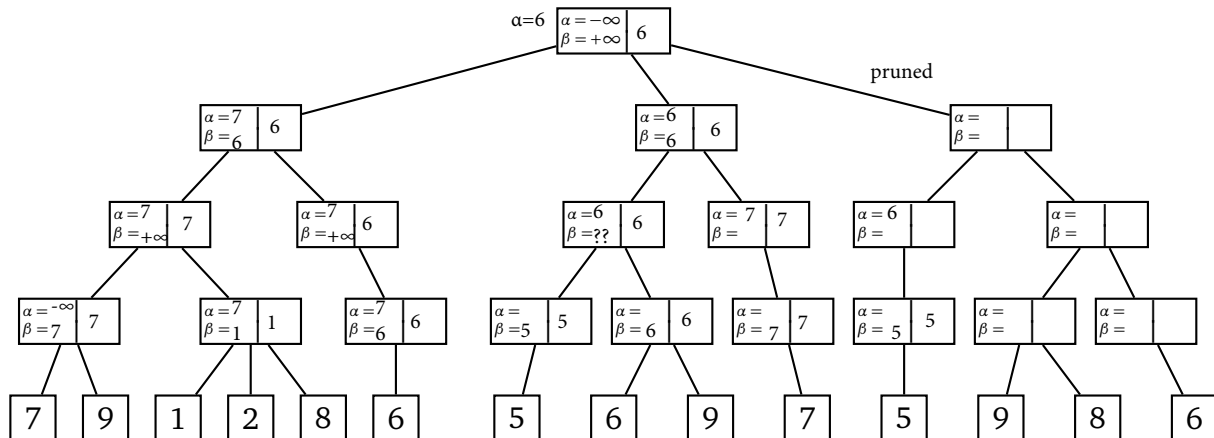
Verwenden Sie wie in den wöchentlichen Aufgaben, folgende Notation:



Aufgabe 2.4

(1,5 Punkte)

Verwenden Sie die MinMax Suche mit Alpha-Beta-Pruning auf dem folgenden Spielbaum. Gehen Sie davon aus, dass der MAX-Spieler beginnt. Tragen Sie nur die endgültigen α und β und Werte in die jeweiligen Knoten ein.



Hinweis: Sie müssen den Alpha-Beta Algorithmus nicht implementieren.

Aufgabe 2.5

(1 Punkt)

Fügen Sie Ihrer Klasse Node (siehe erstes Übungsblatt) eine Methode:

```
alpha_beta_search(self, alpha, beta, is_max_player)
```

hinzu, die auf dem Node-Objekt `self` eine α - β Suche durchführt. Der Parameter `is_max_player` gibt an ob es sich aktuell um eine MAX-Ebene oder MIN-Ebene handelt.