



AntMe!
Mentoring WiSe 2019/20

Diane Hanke, Hannes Kneiding, Alexander Korzec,
Anton Kriese, Sönke Schmidt

Freie Universität Berlin

4. Oktober 2019

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme
„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

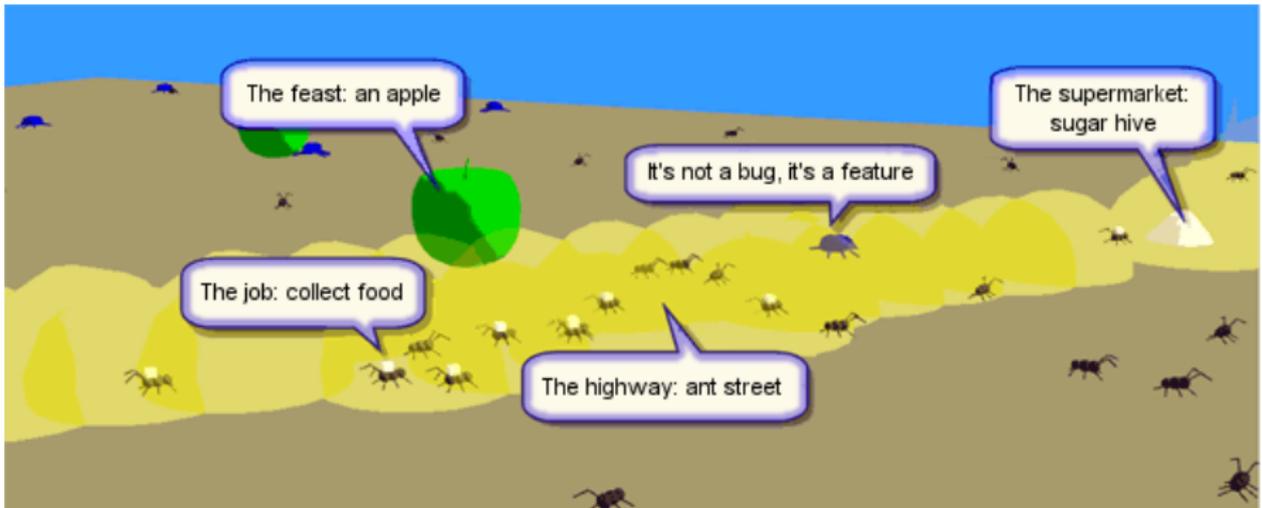
AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

AntMe! - Die sehr kurze Kurzfassung (1/3)



Ziel von AntMe!: **Punkte sammeln!**

- ▶ Zucker sammeln (*1 Punkt pro Einheit Zucker*)
- ▶ Äpfel sammeln (*250 Punkte pro Apfel*)
- ▶ Wanzen töten (*150 Punkte pro Wanze*)
- ▶ Gegnerische Ameisen töten (*5 Punkte pro Ameise*)
- ▶ Eigene Ameisen nicht von gegnerischen Ameisen töten lassen (*-5 Punkte pro Ameise*)

AntMe! - Die sehr kurze Kurzfassung (3/3)

Wie wird AntMe! gespielt?

AntMe! - Die sehr kurze Kurzfassung (3/3)

Wie wird AntMe! gespielt?



```

112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
100%

```

```

/*
 * Interne Methode zur Wiederherstellung der Maxheap-Eigenschaft.
 * Maxheap-Eigenschaft: Schlüssel vom Elternknoten => Schlüssel vom linken Kindknoten und
 * Schlüssel vom Elternknoten => Schlüssel vom rechten Kindknoten.
 */
private void maxHeapify(int i) {
    int l = left(i);
    int r = right(i);
    int maximum = i;
    if(l <= this.heapsize && l == data[l].CompareTo(data[i])) {
        maximum = l;
    }
    if(r <= this.heapsize && r == data[r].CompareTo(data[maximum])) {
        maximum = r;
    }
    if(maximum != i) {
        swap(maximum, i);
        maxHeapify(maximum); //Rekursion
    }
}

/*
 * Gibt das Element mit maximalem Schlüssel aus dem Heap zurück und löscht es.
 */
internal T extractMax() {
    T max = data[0].item;
    data[0] = data[this.heapsize - 1];
    this.heapsize--;
    maxHeapify(0);
    return max;
}

```

► Ihr werdet ein *Ameisenvolk* für AntMe! in C# programmieren!

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

Ihr braucht...

- ▶ Zugriff auf das KVV ⇒ **Fachbereichsaccount!**
 - ▶ Aktivierung des Fachbereichsaccount: <https://portal.mi.fu-berlin.de/login>

Wollt ihr auf eurem PC/Laptop arbeiten, dann benötigt ihr...

- ▶ *AntMe!*
- ▶ eine *Entwicklungsumgebung*
 - ▶ *Visual Studio*
 - ▶ *SharpDevelop*
- ▶ *Microsoft .NET Framework 4*
- ▶ *Microsoft XNA Framework Redistributable 4.0 Refresh*

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

Typische Sorge: „*Ich habe noch nie programmiert. Wahrscheinlich baue ich nur eine Schrott-KI :(Lohnt es sich überhaupt dann mitzumachen?*“

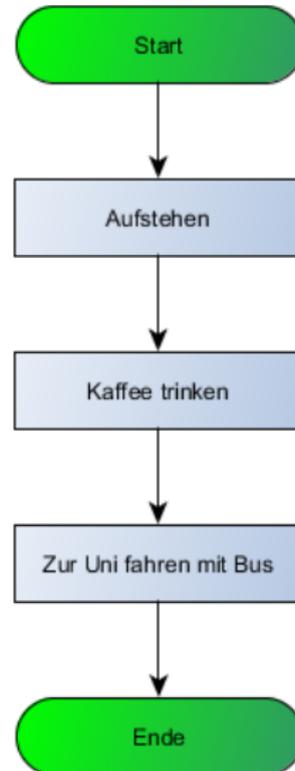
Typische Sorge: „*Ich habe noch nie programmiert. Wahrscheinlich baue ich nur eine Schrott-KI :(Lohnt es sich überhaupt dann mitzumachen?*“



- ▶ AntMe! abstrahiert stark von der eigentlichen Programmieretechnik. Ameisen lassen sich mit vorgefertigten „Befehlen“ recht intuitiv programmieren.

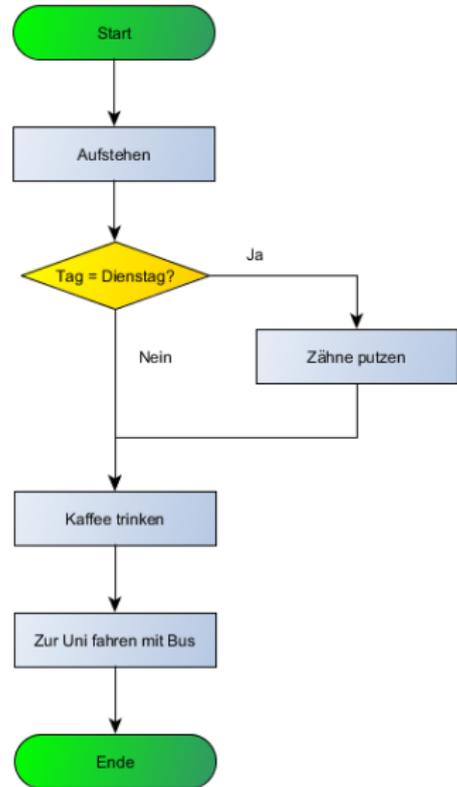
„Programmierskills“ - Programm (2/5)

- 1 Stehe auf
- 3 Trinke einen Kaffee
- 5 Fahre zur Uni mit dem Bus



„Programmierskills“ - Bedingte Anweisung (3/5)

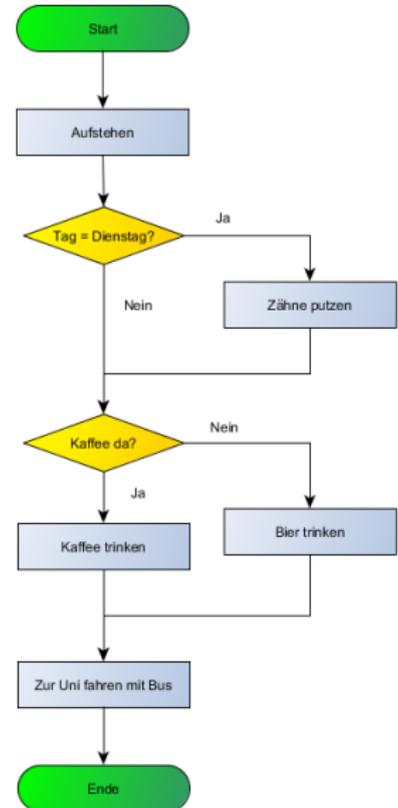
- 1 Stehe auf
- 3 Falls Tag = Dienstag dann
- 4 Putze dir die Zähne
- 6 Trinke einen Kaffee
- 8 Fahre zur Uni mit dem Bus



„Programmierskills“ - Verzweigung (4/5)

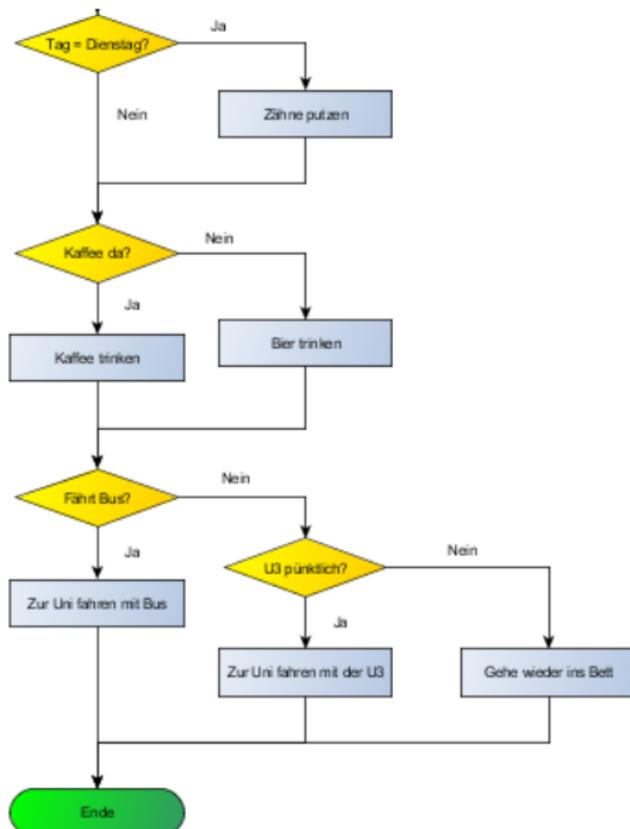
```

1 Stehe auf
3 Falls Tag = Dienstag dann
4   Putze dir die Zähne
6 Falls ausreichend Kaffee da dann
7   Trinke einen Kaffee
8 sonst
9   Trinke Bier
11 Fahre zur Uni mit dem Bus
  
```



„Programmierskills“ - Mehrfachverzweigung (5/5)

- 1 Stehe auf
- 3 Falls Tag = Dienstag dann
- 4 Putze dir die Zähne
- 6 Falls ausreichend Kaffee da dann
- 7 Trinke einen Kaffee
- 8 sonst
- 9 Trinke Bier
- 11 Falls Bus fährt dann
- 12 Fahre zur Uni mit dem Bus
- 13 sonst Falls U3 rechtzeitig dann
- 14 Fahre zur Uni mit der U3
- 15 sonst
- 16 Gehe wieder ins Bett



AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

Arten von Eigenschaften

- ▶ Grundeigenschaften
 - ▶ Besitzt eure Ameise von Anfang an:
 - ▶ Maximale Energie
 - ▶ Maximal mögliche Last
 - ▶ Angriff
 - ▶ ...
- ▶ Zustandseigenschaften
 - ▶ Wird zur Spielzeit ermittelt:
 - ▶ Aktuelle Energie
 - ▶ Aktuelle Last
 - ▶ ...

Modifikation von Grundeigenschaften (1/2)

Eigenschaften eurer Ameisen und deren Modifikatoren:

Modifikator	-1	0	1	2
<i>Geschwindigkeit</i>	3 Schritte/Runde	4 Schritte/Runde	5 Schritte/Runde	6 Schritte/Runde
<i>Drehgeschwindigkeit</i>	6 Grad/Runde	8 Grad/Runde	12 Grad/Runde	16 Grad/Runde
<i>Last</i>	4 Einheiten Zucker	5 Einheiten Zucker	7 Einheiten Zucker	10 Einheiten Zucker
<i>Sichtweite</i>	45 Schritte	60 Schritte	75 Schritte	90 Schritte
<i>Reichweite</i>	0.75 · Standard	1 · Standard	1.5 · Standard	2 · Standard
<i>Energie</i>	50 LP	100 LP	175 LP	250 LP
<i>Angriff</i>	kein Angriff	10 LP/Runde	20 LP/Runde	30 LP/Runde

- ▶ Standard = 2250 Schritte
- ▶ Initial sind alle Modifikatoren auf 0 eingestellt. Diese können auf einen Wert $\in \{-1, 0, 1, 2\}$ eingestellt werden.
- ▶ Die Summe der Modifikatoren muss 0 ergeben!
- ▶ Die Werte in der Tabelle sind die *Defaultwerte* in der *1.6 beta* Version. Diese gelten beim Turnier.

Modifikation von Grundeigenschaften (2/2)

```

1 [Kaste(
2     Name = "Turbosammler",           // Name der Berufsgruppe
3     AngriffModifikator = -1,        // Angriffsstärke einer Ameise
4     DrehgeschwindigkeitModifikator = 0, // Drehgeschwindigkeit einer Ameise
5     EnergieModifikator = -1,       // Lebensenergie einer Ameise
6     GeschwindigkeitModifikator = 2, // Laufgeschwindigkeit einer Ameise
7     LastModifikator = 2,           // Tragkraft einer Ameise
8     ReichweiteModifikator = -1,    // Ausdauer einer Ameise
9     SichtweiteModifikator = -1     // Sichtweite einer Ameise
10 )]
```

- ▶ Angriff
- ▶ Drehgeschwindigkeit
- ▶ Kaste
- ▶ MaximaleEnergie
- ▶ MaximaleGeschwindigkeit
- ▶ MaximaleLast
- ▶ Reichweite (*Wie weit die Ameise laufen kann, ohne den Bau zu besuchen*)
- ▶ Sichtweite

Auflistung - Zustandseigenschaften (2/2)

- ▶ AktuelleEnergie
- ▶ AktuelleLast
- ▶ AnzahlAmeisenDerSelbenKasteInSichtweite
- ▶ AnzahlAmeisenDesTeamsInSichtweite
- ▶ AnzahlAmeisenInSichtweite
- ▶ AnzahlFremderAmeisenInSichtweite
- ▶ EntfernungZuBau
- ▶ GetragenesObst (*Referenz* auf getragenes **Obst**)
- ▶ IstMüde
- ▶ WanzenInSichtweite
- ▶ Ziel (*Referenz* auf ein **Spielobjekt**)
- ▶ ZurückgelegteStrecke (*wie viele Schritte die Ameise seit ihrem letzten Besuch im Bau zurückgelegt hat*)

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

```
1 public class LangweiligeDemoKlasse : Basisameise
2 {
4     Kasten
6     Fortbewegung
8     Nahrung
10    Kommunikation
12    Kampf
14 }
```

Ereignisse - Kasten (2/6)

```
1 #region Kasten
3 public override string BestimmeKaste(Dictionary<string, int> anzahl)
4 {
5     // Gibt den Namen der betroffenen Kaste zurück.
6     return "Turbosammler";
7 }
9 #endregion
```

- ▶ Wird immer dann aufgerufen, wenn neue Ameisen erzeugt werden.
- ▶ Ist nur wirklich interessant, wenn ihr mehrere Kasten habt.

Ereignisse - Fortbewegung (3/6)

```

1 #region Fortbewegung
3 public override void Wartet() {...}
5 public override void WirdMüde() {...}
7 public override void IstGestorben(Todesart todesart) {...}
9 public override void Tick() {...}
11 #endregion
    
```

- ▶ `Wartet()` wird immer dann aufgerufen, wenn die Ameise kein Ziel hat.
- ▶ Eine Ameise ruft einmal `WirdMüde()` auf, wenn

$$\text{ZurückgelegteStrecke} = \frac{\text{Reichweite}}{3}$$

gilt.

- ▶ `Tick()` ist nicht wirklich an ein Ereignis gebunden und wird von jeder Ameise in jeder Runde aufgerufen!

```
1 #region Nahrung
3 public override void Sieht(Obst obst) {...}
5 public override void Sieht(Zucker zucker) {...}
7 public override void ZielErreicht(Obst obst) {...}
9 public override void ZielErreicht(Zucker zucker) {...}
11 #endregion
```

```
1 #region Kommunikation
3 public override void RiechtFreund(Markierung markierung) {...}
5 public override void SiehtFreund(Ameise ameise) {...}
7 public override void SiehtVerbündeten(Ameise ameise) {...}
9 #endregion
```

- ▶ `SiehtVerbündeten()` könnt ihr komplett ignorieren für den Wettbewerb. Es finden keine Teamkämpfe statt.

```
1 #region Kampf
3 public override void SiehtFeind(Ameise ameise) {...}
5 public override void SiehtFeind(Wanze wanze) {...}
7 public override void WirdAngegriffen(Ameise ameise) {...}
9 public override void WirdAngegriffen(Wanze wanze) {...}
11 #endregion
```

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen

Befehle - Überblick (1/7)

- ▶ Gehen
- ▶ Drehen
- ▶ Nahrung
- ▶ Markierung
- ▶ Kampf
- ▶ Fehlersuche

Für eine Beschreibung siehe: <http://wiki.antme.net/de/API1:Befehlsliste>

Befehle - Gehen (2/7)

```
1 public void BleibStehen();  
3 public void GeheGeradeaus();  
5 public void GeheGeradeaus(int entfernung);  
7 public void GeheWegVon(Spielobjekt ziel);  
9 public void GeheWegVon(Spielobjekt ziel, int entfernung);  
11 public void GeheZuBau();  
13 public void GeheZuZiel(Spielobjekt ziel);
```

```
1 public void DreheInRichtung(int richtung);  
3 public void DreheUm();  
5 public void DreheUmWinkel(int winkel);  
7 public void DreheZuZiel(Spielobjekt ziel);
```

Befehle - Nahrung (4/7)

```
1 public void LasseNahrungFallen();  
3 public void Nimm(Nahrung nahrung);
```

- ▶ Ein Apfel wiegt deutlich mehr als ein Zuckerwürfel und lässt sich nur mit mehreren Trägern *effizient* transportieren.
- ▶ Mit BrauchtNochTräger (obst) könnt ihr prüfen, ob Ameisen beim tragen mithelfen können.



Befehle - Markierung (5/7)

- ```
1 public void SprüheMarkierung(int information);
3 public void SprüheMarkierung(int information, int gröÙe);
```

- ▶ Je größer ihr eure Markierungen wählt, desto schneller verschwindet diese auch.

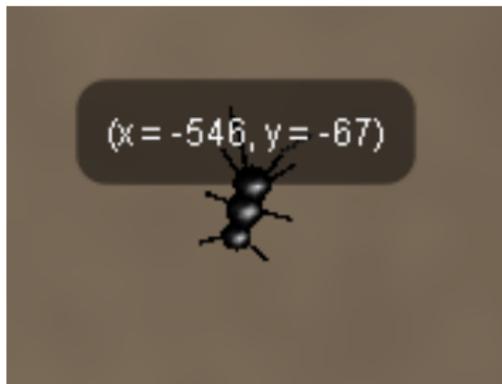


```
1 public void GreifeAn(Insekt ziel);
```

- ▶ Ist nur dann sinnvoll, wenn die Ameise *keine* Nahrung trägt und der Modifikator *AngriffModifikator* der Ameise mindestens 0 ist.

```
1 public void Denke(string nachricht);
```

**Tipp:** Damit die Ameisen anfangen zu „denken“, solltet ihr in der 2D/3D-Simulation die D-Taste drücken.



AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

AntMe!-Turnier

Quellen



<http://wiki.antme.net/de/Hauptseite>



<https://www.youtube.com/watch?v=wrsj4BzFrVE&list=PLFOBQ8ri3LGyLiarJcYC9E>

AntMe! - Die sehr kurze Kurzfassung

Was brauche ich für AntMe! ?

Accounts & Programme

„Programmierskills“

Über Ameisen

Eigenschaften

Ereignisse

Befehle

Weitere Hilfestellungen

AntMe!-Wiki

Let's Code AntMe! Videos

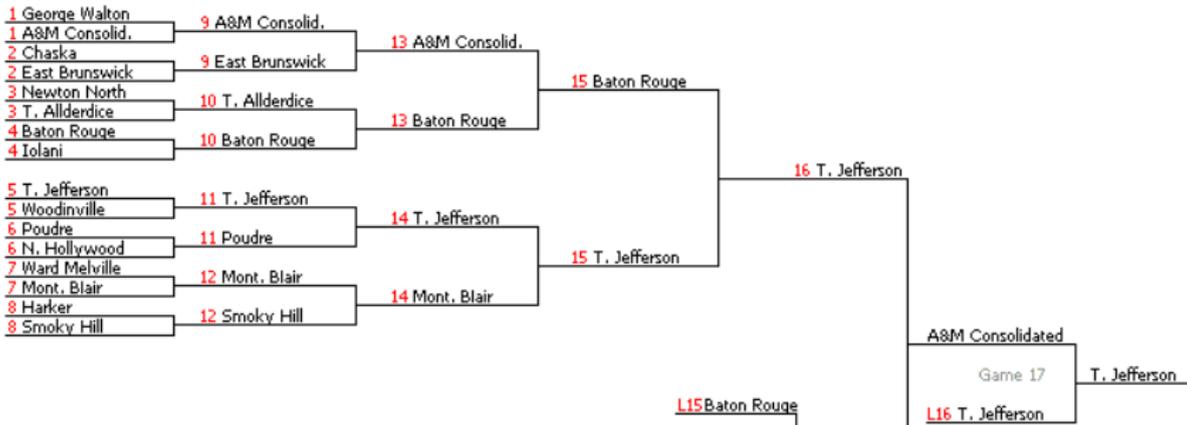
**AntMe!-Turnier**

Quellen

## AntMe!-Turnier:

- ▶ Zeit: Do., 10.10.2019 13:00-15:00 Uhr
- ▶ Ort: T9, großer Hörsaal
- ▶ Turnierform:
  - ▶ „Doppel K.O.-System“ (etwas abgewandelt)
  - ▶ In der ersten Runde spielen mehrere Teams gegeneinander
  - ▶ Danach 1 vs 1
  - ▶ **5000** Ticks
- ▶ Fertige DLL-Datei bis **Mi., 09.10.2019 18:00 Uhr** an [alexander.korzec@fu-berlin.de](mailto:alexander.korzec@fu-berlin.de) senden!
- ▶ *Statische Ameisenvölker sind erlaubt!*

# AntMe!-Turnier (2/4)



```
1 [Spieler(
2 Volkname = "Unberechenbare Ameisen",
3 Vorname = "Alan, Alonzo"
4 Nachname = "Turing, Church"
5)]
```

- ▶ Vergesst nicht euren Ameisen einen Namen zu geben!
- ▶ Vergesst eure Namen nicht!







# Viel Erfolg!

## Referenzen

- ▶ <http://www.antme.net/de/>
- ▶ <http://wiki.antme.net/de/Hauptseite>
- ▶ [http://userpage.fu-berlin.de/soenke/index.php?seite=mentoring\\_AntMe](http://userpage.fu-berlin.de/soenke/index.php?seite=mentoring_AntMe)
- ▶ <https://github.com/matthid/antme>
- ▶ <https://www.youtube.com/watch?v=wrsj4BzFrVE&list=PLFOBQ8ri3LGyLiarJcYC9E>

## Grafiken

- ▶ <https://www.antme.net/wp-content/themes/antme/img/logoen.png>
- ▶ [https://pixabay.com/static/uploads/photo/2013/07/12/19/15/smiley-154420\\_960\\_720.png](https://pixabay.com/static/uploads/photo/2013/07/12/19/15/smiley-154420_960_720.png)
- ▶ <https://upload.wikimedia.org/wikipedia/commons/6/68/NSB-doubleelim-draw-2004.png>
- ▶ <https://www.youtube.com/watch?v=wrsj4BzFrVE&list=PLFOBQ8ri3LGyLiarJcYC9E>
- ▶ [https://pixabay.com/static/uploads/photo/2013/07/12/18/58/keyboard-154116\\_960\\_720.png](https://pixabay.com/static/uploads/photo/2013/07/12/18/58/keyboard-154116_960_720.png)
- ▶ [https://pixabay.com/static/uploads/photo/2013/07/13/12/09/ant-159283\\_960\\_720.png](https://pixabay.com/static/uploads/photo/2013/07/13/12/09/ant-159283_960_720.png)
- ▶ <http://www.publicdomainpictures.net/pictures/10000/velka/1335-12439673009u9Z.jpg>