Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет информационных технологий и управления

Кафедра вычислительных методов и программирования

ОТЧЕТ

по лабораторной работе

**«Преследование и уклонение»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 924403 |  | К.Д. САНДАРОВИЧ |
| Проверил |  | С.О. ЛУКАШОВ |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск 2022

# Общая постановка задачи

## 1. Краткие теоретические сведения

Модели преследования и уклонения отлично подходят для реализации в первую очередь, потому что опираются только на самые основные модели поведения и расширяют их возможностью предсказания следующего шага к цели. Нам потребуются две базовые модели поведения: Seek и Flee.

2. Системные подходы к решению задачи

Нам потребуются две базовые модели поведения: Seek и Flee. Для алгоритмов Arrive и Leave необходимо создать отдельные файлы и не забыть настроить порядок их выполнения.

3. Разбор решения поставленной задачи

Эти модели поведения базируются на моделях Seek и Flee, но вдобавок учитывают скорость цели, чтобы предсказать, куда двигаться дальше. Координаты целевой точки при этом сохраняются в дополнительном внутреннем объекте.

1. Реализуем две базовые модели поведения: Seek и Flee.

2. Поместим их в порядке выполнения сценариев сразу после класса Agent.

3. Создадим класс Pursue, наследующий класс Seek, и добавим атрибуты для прогнозирования.

4. Реализуем в классе Pursue функцию Awake для настройки всего, что связано с реальной целью.

5. Реализуем внутри класса Pursue функцию OnDestroy для корректной обработки внутреннего объекта.

6. Далее напишем реализацию функции GetSteering.

7. Реализация модели поведения Evade выглядит точно так же, с той лишь разницей, что она наследует класс Flee.

Эти модели поведения, Pursue и Evade, базируются на моделях Seek и Flee, но вдобавок учитывают скорость цели, чтобы предсказать, куда двигаться дальше. Координаты целевой точки при этом сохраняются в дополнительном внутреннем объекте.

# Блок-схема алгоритма

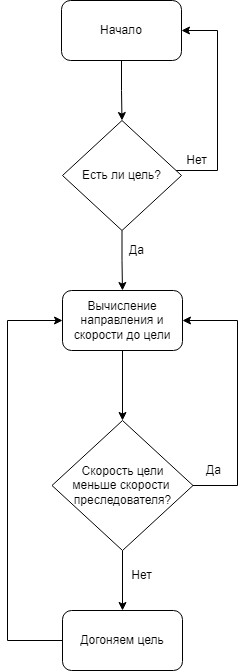


Рисунок 1 – блок-схема алгоритма преследования

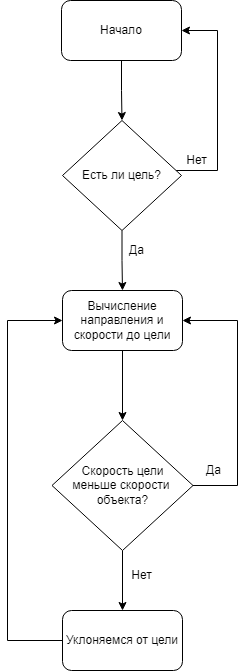


Рисунок 2 – блок-схема алгоритма уклонения

# Листинг

using UnityEngine;

using System.Collections;

public class Pursue: Seek

{

public float maxPrediction;

private GameObject targetAux;

private Agent targetAgent;

public override void Awake()

{

base.Awake();

targetAgent = target.GetComponent<Agent>();

targetAux = target;

target = new GameObject();

}

public override Steering GetSteering()

{

Vector3 direction = targetAux.transform.position - transform.position;

float distance = direction.magnitude;

float speed = agent.velocity.magnitude;

float prediction;

if (speed <= distance / maxPrediction)

prediction = maxPrediction;

else

prediction = distance / speed;

target.transform.position = targetAux.transform.position;

target.transform.position += targetAgent.velocity \* prediction;

return base.GetSteering();

}

void OnDestroy ()

{

Destroy(targetAux);

}

}

using UnityEngine;

using System.Collections;

public class Evade: Flee

{

public float maxPrediction;

private GameObject targetAux;

private Agent targetAgent;

public override void Awake()

{

base.Awake();

targetAgent = target.GetComponent<Agent>();

targetAux = target;

target = new GameObject();

}

public override Steering GetSteering()

{

Vector3 direction = targetAux.transform.position - transform.position;

float distance = direction.magnitude;

float speed = agent.velocity.magnitude;

float prediction;

if (speed <= distance / maxPrediction)

prediction = maxPrediction;

else

prediction = distance / speed;

target.transform.position = targetAux.transform.position;

target.transform.position += targetAgent.velocity \* prediction;

return base.GetSteering();

}

void OnDestroy ()

{

Destroy(targetAux);

}

}

# Вывод

В процессе выполнения лабораторной работы были реализованы модели поведения для исскуственного интеллекта на Unity такие как «преследование» и «уклонение».