Python Software Foundation

Panda3D: Recast and Detour Integration

BASIC INFORMATION

Personal Info
- **Name:** Ashwini Jha
- **Discord:** imaj_ashwini
- **Time Zone:** India (UTC+5:30)
- **GitHub:** [ash1215](ash1215)

University Info
- **University:** Indian Institute of Technology, Indore
- **Major:** Computer Science and Engineering
- **Current Year:** 3rd year (2021 expected graduation)
- **Degree:** Bachelor of Technology (B.Tech)

CODING SKILLS

Programming Languages and Frameworks
- Fluent in C/C++, Python and Javascript.
- Worked on Godot in GdScript.
- Developed small web games using HTML/JavaScript.
- Also worked with Flask, Qt, PyQt5, ReactJS.
- Familiar with Blender, Unity3D.
- Have done analysis of various pathfinding algorithms (A*, Dijkstra, BFS) as part of my course project. The implemented comparisons are live here:
    - A* vs Dijkstra vs BFS
    - Comparing Heuristics

Development Environment
- Ubuntu 18.04 LTS or Windows 10 (I work on former most of the time)
- Sublime Text as primary editor. Fully customised for efficient use.

PRE-GSOC INVOLVEMENTS

Contribution to Panda3D

I've made a pull request to Panda3D. I tried to solve the issue #888 on Panda3D github repository.

- **#893:** Solving the issue : PNMImage.add_sub_image() samples source image out of bounds #888

PROJECT INFORMATION

Panda3D: Recast and Detour Integration

Panda3D is an open source framework for 3D rendering and games. It combines the speed of C++ with the ease of use of Python in order to provide users a fast rate of development without sacrificing on performance.
The project I wish to work upon is Integration of Navigation-mesh Toolset for Games in Panda3D. Recast and Detour are tools/libraries for building navigation meshes and performing pathfinding. Once these tools/libraries are successfully integrated with Panda3D, it would be possible for a Panda3D *actor* to move from the source to the destination avoiding collisions with the walls and inappropriate paths.

DETAILED DESCRIPTION

Recast and Detour are tools/libraries for building navigation meshes and performing pathfinding
- **Recast:** Recast is state of the art navigation mesh construction toolset for games.
- **Detour:** Detour offers simple static navigation mesh which is suitable for many simple cases, as well as tiled navigation mesh which allows you to plug in and out pieces of the mesh. The tiled mesh allows you to create systems where you stream new navigation data in and out as the player progresses the level, or you may regenerate tiles as the world changes.

Panda3D, being a 3D engine is also used for game development where navmesh play a very important role. Path-finding and avoiding hurdles or blocks is important for a game character/actor to walk/run properly. *Recast* provides a proper navmesh over which *Detour* applies path-finding and spatial reasoning.

Classes to be integrated into Panda3D include:

- **NavMeshBuilder** - requires recast and returns a NavMesh, a separate builder allows not needing recast at runtime if loading from BAM
- **NavMeshNode** - anchor the navmesh in the scene graph (ie. make the position of other things relative to it meaningful), and also to allow us to display a visual representation, the way one can visualize collision nodes
- **NavMesh** - stores the navmesh
- **NavMeshQuery** - queries a navmesh and follows Detour's API (dtNavMeshQuery)

This will require working on the design of the framework/API to be integrated so that it functions well with other libraries of Panda3D. Getting continuous feedback from the mentors and community will be an important part of this project.
Later, more NavMesh building options might also be added along with more query options. Proper documentation is also an important part of any project. Usability of each library/tool/function will be added in the official documentation along with sample program illustrating the use of API.
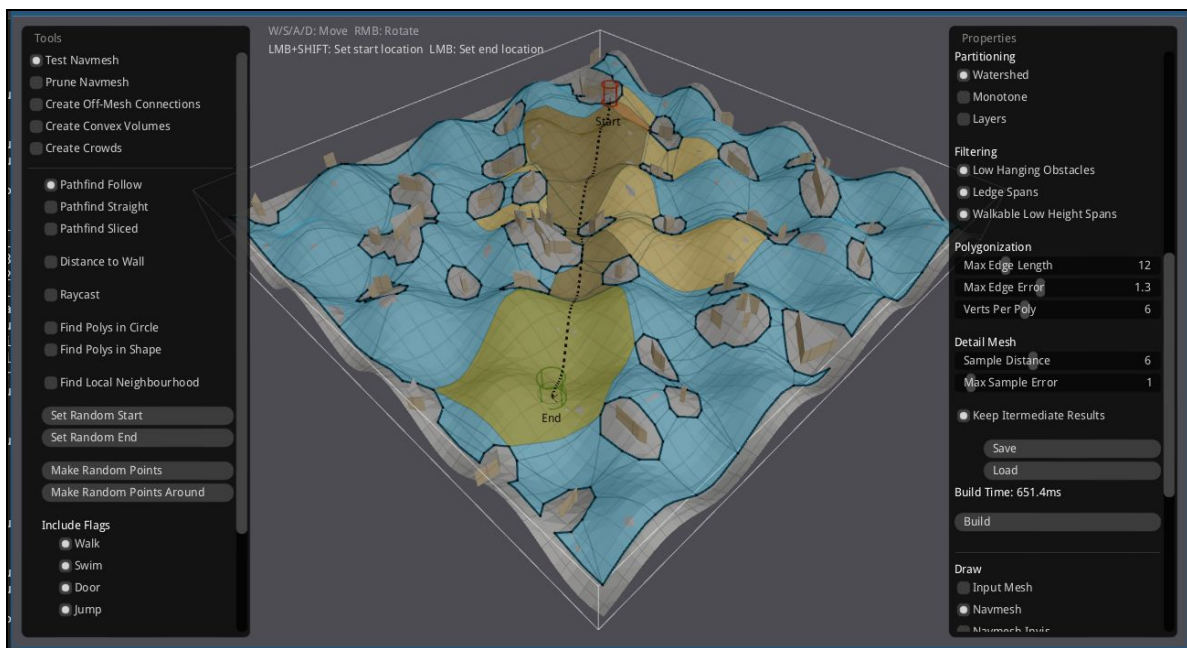
Recast Demo showing Pathfinding over NavMesh

A sample output using Panda3D

PROPOSED TIMELINE

May 4 - June 1 (Community Bonding)

- Familiarising myself with the mentors and the codebase. I will dive deep into the code and learn all the ins and outs of the code with help of mentors.
- Learning from the mentors about how Panda3D is structured which would eventually help me to integrate the required tools/libraries.
- Familiarising myself with the various libraries of Recast and Detour.
- Knowing more about NavMesh and its implementation in other softwares.
- Going through the code for implementation of various NavMesh functions like query, random points and properties like height mesh, radius, jump implemented in Recast and Detour.
- Design basic structure of the Detour and Recast integration with Panda3D.
- Discuss about design of the API with other developers and get feedback.

June 1 - June 7 (Week 1)

- Review Recast demo for how to build a navigation mesh and Detour demo for how to query a navigation mesh.

June 8 - June 14 (Week 2)
- Pull source directories DebugUtils, Detour, DetourCrowd, DetourTileCache, and Recast from Recast and Detour repository as external libraries.
- Modify build systems to be able to build against Recast and Detour, both using CMake and makepanda.

June 15 - June 21 (Week 3)
- Start working on providing support for building a navmesh from a Geom

June 22 - June 28 (Week 4)
- Write unit tests for the functions.
- Implement NavMeshNode class.

June 29 - July 5 (Week 5,6)
- Visualizing results of navmesh building and debug.
- Testing the code for different navmesh.
- Discuss the results with other developers and get feedback.

July 13 - July 26 (Week 7,8)
- Implement NavMeshQuery with support for find_path() and find_straight_path()
- Add the usability of the API and its integration in the official documentation for Panda3D

July 27 - August 2 (Week 9)
- Support serialization of NavMeshes to/from BAM
- Make a sample program using these libraries

August 3 - August 16 (Week 10)
- Discuss with the mentors and add some extra NavMesh generation options, like:
    - Heightfield NavMesh
    - Collision Mesh
- Add more query options to the NavMesh.
- Clean up the code so that it remains useful and maintainable for future contributions.

August 17 - August 23 (Final Week)
- Buffer period for incomplete work or fix bugs reported during the GSoC period.
- Completion of tasks mentioned in the wishlist.
- Prepare for final evaluation.

PHASE DELIVERABLES

Phase I Evaluation
- Design for the API is ready
- Build systems can build against the *Recast and Detour* tool.
- Support for building NavMesh using Geom.

Phase II Evaluation
- NavMeshNode is implemented
- NavMeshQuery support added.
- Documentation is ready

Final Evaluation
- Support serialization of NavMeshes to/from BAM
- Support NavMesh building from Heightfield.
- Recast and Detour tool integrated with Panda3D with proper documentation, sample program and unit test files.

ABOUT ME

I am a third year student currently enrolled in Computer Science and Engineering (IV Year Course) at IIT Indore. I developed a passion for programming and game development in my junior years here at IIT Indore. I am an active member of The Programming Club. I have regularly participated in various hackathons. I won fourth prize at the Samsung Innovation Award 2019 for developing a biometric security software detecting liveness of a fingerprint.

MOTIVATION

My motivation for GSoC this year is getting started with the open source community. GSoC is a great program for introducing organizations with prospective contributors and when I saw this project, I thought this is something that I can do.
During my coursework, I have studied and analysed various pathfinding algorithms such as A* search, Dijkstra's, Greedy BFS and some modified versions of A* search which I found really interesting.
I chose this particular organisation because it is very well aligned with my interests. Other than that, this project provides a good opportunity to apply my learnings on a practical scale. This is my field of interest and therefore the natural inclination to this project.

OTHER COMMITMENTS

I have no other commitments during the GSoC period. Due to the pandemic, our lectures and exams have been postponed and might continue from May to beginning of June. Since, this is the community bonding period, so I would be able to manage, read the documentations and most importantly communicate with the mentors and the greatly helpful community. After the first week of June, I will be free and able to give in 40 hours of efforts per week.