UBISOFT TORONTO next

PROGRAMMING

Be Inspired. Be Creative. Be NEXT.

Ubisoft Toronto NEXT is an annual competition designed to showcase the talent of video game development students in Ontario. Kick-start your career in the industry with a paid Internship and learn from Ubisoft Toronto's top talent.

Compete in one of seven disciplines to win an internship:

- 3D Art (Modelling)
- Concept Art
- Technical Art
- Animation
- Game Design
- Level Design
- Programming

Eligibility

The competition is open to applicants who:

- Are legally entitled to work in Canada;
- Reside in Ontario;
- Are currently attending OR have graduated from an Ontario post-secondary school no earlier than Spring 2022;
- Have less than 2 years of professional experience in the game development industry;
- Are not a current or former employee of any Ubisoft studio;
- Complete the challenge individually. Team-based submissions will not be reviewed.

Questions?

Email us at NEXT@Ubisoft.com. Note: Email is not monitored 24/7 but we will do our best to respond within 48 hours.

Ubisoft Toronto NEXT: Programming Challenge

Challenge Brief

This year's challenge is focused on the theme - Firing Projectiles (e.g. Angry Birds, Worms etc.). You can choose if you want to create a single player **or** multi-player game. Using the API provided, create a game that showcases a well-crafted and compelling game experience. While your game can be inspired by the examples provided above, you must innovate and add your own flavour.

The provided framework supports drawing sprites, lines, and text. It also includes a simple controller and sound system.

Your entry must be a Windows application **written entirely in C++**. You will need to download Microsoft Visual Studio 2022 to successfully complete this challenge. If you are a Mac user, please use Boot Camp to install Windows.

This challenge must be done individually, and **all code must be original**. Team based submissions will not be reviewed. Do not use external libraries or write OpenGL code.

This is a **programming** challenge – we will **ignore** artistic merit in the judging. Using novel programmed effects to add visual flair is ok, but we are less interested with how the game looks, and more in **how well it plays and how well the underlying code is crafted**.

Click here to download the most up-to-date version of the API

Please note the following updates to the API (since the last release):

Issue	**Fix**
Sprite Animation accepts frame numbers outside of acceptable range	In *CSimpleSprite::SetFrame*, update the range check to use *>=* instead of just *>*
Sprite animation time calculations incorrect when delta time is large	Use *fmod* in *CSimpleSprite::Update* to correctly compute animation time

Texture loading using c strings in map keys	Update the *SimpleSprite* class to use C++ *std::string* as the loaded texture map's key type for flexibility/robustness
When starting a new animation on a sprite, it won't start from the beginning	Add an overload to *CSimpleSprie::SetAnimation* to allow starting the new animation from the beginning. This keeps the default behaviour & sample code working while allowing a fix for the issue
When pressing the Quit input key the Shutdown function is not called.	Update the *Idle* function inside *main.cpp* to not hard exit the application using *exit(0);*, but instead just leave the GLUT loop by using *glutLeaveMainLoop();*

Game Specifications

Go wild! Make a game that you enjoy playing. Keep to the theme but add your own spin on things. Look for interesting gameplay features you could add.

- Your game must successfully compile and run on a Windows 10 / 11 PC.
- **Do not change / modify the API.** Use only the provided API for all input & output.
- The API Supports
    - Sprites, lines, and text.
    - Keyboard, Mouse & Controller.
    - Simple Sound Support.
- You are not required to use sprites, but the API supports them.

Tips

- The central theme of the challenge is to create a game that is projectile based. As an example, Worms is a game where the player has team of worms that shoot projectiles at enemy worms. Whereas with Angry Birds, you launch projectiles to destroy objects. You could include these elements, but you have complete freedom to innovate.
- Feel free to be creative and push your game in an interesting direction. Think about bringing a new twist to this theme.
- Think of additional elements you can add to the game i.e., AI, multi-player, networking, physics elements etc. How would this affect gameplay?
- DO NOT spend too much time on the aesthetics. Remember, this is **not an art challenge.**
- **Use your time effectively**. You do not have to delete work in progress code for things you do not need. You can just comment it out.

- Do not limit yourself to the suggestions here. Your unique decisions could be the deciding factor.
- This is a learning experience and an opportunity to get feedback from industry professionals! There will be no prejudice for future submissions, **JUST GIVE IT A TRY!**
- **If** you choose to use sound / music, please be aware of copyright issues on YouTube.

Submission Requirements & Documentation

Provide your submission using this Submission Form and submit your resume online.

Your application package **must** contain the following three (**3**) parts. All items are mandatory.

1. Up-to-date resume in PDF format.
2. Your complete code and documentation saved in a .zip file:
    a. The submission package must be named as follows: **NEXT_2023-2024_Programming_FirstnameLastname.zip**
    b. Documentation must be in PDF format.
3. The link to a YouTube video of a play-through of your game with screen capture software (e.g., OBS) highlighting the gameplay and technical features. Please make sure to submit a good quality video. The video should not be longer than 5 minutes in duration. You can choose to maintain your anonymity in your YouTube video (use text instead of voice / not disclose your name or image etc. – you will not be judged for any of these choices).

Judging Criteria

A panel of expert judges will individually rank the overall entry package, out of 35, based on the following criteria. This is a technical and gameplay challenge. Artistic merit will not be judged.

Code Structure (1-15):

- Clear and simple structures with thought for future expansion
- Self-documenting code with clear interfaces
- Well-managed memory and data flow
- Understanding of game programming patterns

Technical Challenge (1-10):

- Advanced and well-applied techniques
- Ambitious design

Innovation (1-5):

- Unique solutions to technical problems
- Interesting and effective gameplay

Written Documentation (1-5):

- Overview of systems and related classes

Key Dates & Deadlines

Submission Deadline: Sunday January 21st, 2024, 11:59 pm EST
Complete this Submission Form where you will be asked to provide a link to your submission package. Late entries will not be accepted.

Ubisoft Toronto judges will review all submissions and select up to five successful participants to proceed to the interviews. All other participants will be notified of their results via email by **Friday, February 16th, 2024**.

Interviews: February 2024

Ubisoft Toronto judges will select up to three finalists after the interviews. Ubisoft Toronto is the sole judge of this competition and reserves the right to elect the panel as it sees fit; all judgement is final and non-negotiable.

Prizes

- 1st Prize:
  - One (1) paid Programming Internship at Ubisoft Toronto. The Internship shall be for a minimum duration of three (3) months at Ubisoft Toronto studio and is currently scheduled to begin May 2024. Dates are subject to change at the sole discretion of Ubisoft;
  - One Ubisoft prize pack valued at $300.
- Finalist Prizes:
  - Display of each finalist's submission in the Ubisoft Toronto NEXT Awards Ceremony;
  - One Ubisoft prize pack valued at $300.