

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 nine disciplines to win an internship:

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

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 2024;
- Have less than 2 years of relevant professional Programming experience;
- 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 - **Simulation of large number of entities** (e.g. Civilization Series, Sims, Advanced Wars, Cities Skylines, Vampire Survivors, Starcraft etc.). You can choose if you want to create a single player **or** multi-player game. Using the [latest API provided](#), create a game that showcases a well-crafted and compelling game experience. While your game can be inspired by the game theme provided above, you must innovate and add your own flavour.

Your submission must be an application written entirely in C++. Regardless of your operating system, you will need to download [CMake](#). This is a build system generator which allows us to build the project for multiple operating systems. **Full instructions on how you can use the API are included in the downloadable API folder.**

For Windows users:

You will need to download [Microsoft Visual Studio 2022](#) to successfully complete this challenge. We recommend [Visual Studio Community 2022](#).

For Mac users:

You will need to download [Homebrew](#) which is a package manager for the Mac operating system. Homebrew will be used to download the [freeglut](#) and [SDL3](#) libraries which are needed to help make the API run.

This challenge must be done individually, and **all code must be original**. Team based submissions will not be reviewed. You are welcome to use anything in C++ standard library but **DO NOT** use any other 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.**

⚠ IMPORTANT Please note the following updates to the API (December 10, 2025):

Make sure you re-download the API as there is a new version of it available.

[\[DOWNLOAD the latest API\]](#)

In this version there are the following changes:

1. Some stylistic mistakes in the code were cleaned up.

2. A bug with the `IsMousePressed` function was fixed.
3. The `DrawTriangle` function was updated to draw 3D triangles instead of 2D ones. For this update make sure you pay attention to the comment above the `DrawTriangle` function in `app.h` to ensure that the proper values are being used for the `x`, `y`, `z`, `w` components of the triangle's vertex data.

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 or the latest MacOS
- **Do not change / modify the API.** Use only the provided API for all input & output.
- The API Supports
 - Sprites, lines, triangles 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 **simulates a large number of entities**. As an example, *Cities Skylines* is a game where the player builds a city from a small town to a sprawling metropolis, and each individual citizen's daily life is simulated. Whereas with *Vampire Survivors*, you are fighting to stay alive for as long as possible against a continuous wave of enemy monsters. Focus your submission on the performance and the concept of having many entities in the game simultaneously. 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.
- We highly recommend using Visual Studio for your submission. You are welcome use any IDE for your submission, as long as the submission runs smoothly in Visual Studio.

Submission Requirements & Documentation

Please follow the three-part instructions below carefully to ensure your submission is complete and eligible for review.

Step 1: Submit Your Resume Separately

- Up-to-date Resume in PDF format
Your resume must be submitted separately from your main submission package.
 - ▶ Use the designated link provided [\[Resume Submission HERE\]](#) to upload your resume. Once successfully submitted, you will receive a welcome message confirming your resume upload.
-

Step 2: Prepare and Submit Your Submission Package

Your submission package must be named using the following format:

NEXT_2025-2026_Programming_FirstnameLastname.zip

Use the designated link to submit your submission package >> [\[Submission Form HERE\]](#)

⚠ Important:

To maintain anonymity and confidentiality of Ubisoft NEXT:

- Do not include your name anywhere inside the submission content (e.g., documents, images, videos, folder names).
 - Your name should only appear in the .zip file name and your resume.
 - Please do not share your submission on any online platform or social media until the competition concludes (April 2026)
-

Step 3: Include the Following Two (2) Components in Your Submission Package

All items are **mandatory**, unless explicitly stated otherwise below.
Additional scene files may be requested for considered submissions.

1. Your complete code and documentation saved in a .zip file:
 - a. The submission package must use the naming convention indicated above.
NEXT_2025-2026_Programming_FirstnameLastname.zip
 - b. Documentation must be in PDF format.

2. Please submit a YouTube link to a video showcasing a play-through of your game, recorded using screen capture software (e.g., OBS). The video should highlight both gameplay and technical features, and must meet the following criteria:
 - a. The video must be no longer than 5 minutes.
 - b. Ensure the video is of good visual (720p or higher) and clear audio quality.
 - c. To maintain anonymity, you may choose to:
Your choice to remain anonymous will not affect the evaluation of your submission.
 - i. Use text overlays or voice narration to describe your play-through
 - ii. Kindly ensure that your name or image are not included in the video title or anywhere within the video content itself.

Judging Criteria

A panel of expert judges will individually rank the overall entry package, out of 50, based on the following criteria. This is a technical and gameplay challenge. Artistic merit will not be judged.

Code Structure (1-20):

- 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 (1-15):

- Advanced and well-applied techniques
- Ambitious design

Innovation (1-10):

- Unique solutions to technical problems
- Interesting and effective gameplay

Written Documentation (1-5):

- Overview of systems and related classes

Key Dates & Deadlines

Submission Deadline: *Monday, January 19th, 2026, 11:59 pm EST*

Complete this [\[Submission Form\]](#) where you will be asked to provide a link to your submission package. **Late submissions will not be accepted.**

Ubisoft Toronto judges will review all submissions and select up to five (5) successful participants to proceed to the interviews. All participants will be notified of their results via email by *Friday, February 6th, 2026.*

Interviews: Interviews will be held virtually in February 2026. Selected participants will receive all information about the interview process via email.

Ubisoft Toronto judges will select up to three (3) finalists after the interviews. Finalists will be invited to the Awards Ceremony where the results of the competition will be announced. 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 2026. Dates 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.