



UIT

TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

KỸ NĂNG NGHỀ NGHIỆP

Đề tài:

SS004 - Đồ Án Cuối Kỳ - Tetris Game

Lớp học phần: CN1.K2025.1.CNTT

Môn học: Kỹ năng nghề nghiệp

Giảng viên hướng dẫn: ThS. Nguyễn Văn Toàn

MỤC LỤC

I	Hợp đồng nhóm	1
II	Công cụ hỗ trợ và quản lý dự án	8
III	Phần giới thiệu và hướng dẫn chơi game	9
1	Giới thiệu về Tetris	9
2	Câu chuyện đằng sau dự án	9
3	Luật chơi cơ bản	10
4	Hướng dẫn điều khiển	12
5	Hệ thống tính điểm	13
6	Cấp độ và độ khó	13
7	Chiến thuật và mẹo chơi hay	14
8	Tính năng đặc biệt	15
9	Cài đặt và khởi động game	16
10	Câu hỏi thường gặp (FAQ)	18
11	Liên hệ và hỗ trợ	20
12	Tài liệu tham khảo	20
13	Lời kết	21

I Hợp đồng nhóm

HỢP ĐỒNG LÀM VIỆC NHÓM

1. Thời gian thành lập

Thứ 4, Ngày 3 tháng 12 năm 2025.

2. Thời hạn hợp đồng

Đến khi hoàn thành ĐỒ ÁN CUỐI KỲ – Môn Kỹ Năng Nghề Nghiệp.

3. Tên nhóm

5 ducks

4. Thành viên

STT	Họ và Tên	MSSV
1	Dương Hoà Long	25730040
2	Lê Quang Nhật	25730047
3	Lê Hữu Nhị	25730048
4	Nguyễn Duy Thanh	25730068
5	Kiều Quang Việt	25730093

5. Mục đích thành lập

- Nâng cao kỹ năng làm việc nhóm cũng như các kỹ năng mềm mà môn *Kỹ năng Nghề nghiệp* yêu cầu.
- Tạo môi trường để các thành viên rèn luyện kỹ năng giao tiếp, phân chia công việc, quản lý thời gian và giải quyết xung đột.
- Hoàn thành bài tập nhóm đúng tiến độ, đúng yêu cầu của giảng viên và đảm bảo chất lượng tốt nhất có thể.
- Toàn nhóm thống nhất phấn đấu đạt **điểm cao (9–10)** cho bài tập cuối kỳ, xem đây như một cơ hội để cải thiện tư duy làm việc nhóm chuyên nghiệp.
- Cùng nhau xây dựng tinh thần học hỏi, chia sẻ kiến thức và nâng cao kỳ vọng bản thân trong môn học.

6. Vai trò của các thành viên trong nhóm

	Lãnh đạo giữ tiến độ	Xử lý đầu vào và di chuyển	Thao tác khối Tetris	Điểm và cấp độ	Hiệu ứng âm thanh	Kiểm thử game	Tổng hợp và báo cáo
Dương Hoà Long		X	X			X	
Lê Quang Nhật	X				X	X	X
Lê Hữu Nhị		X	X			X	
Nguyễn Duy Thanh				X	X	X	
Kiều Quang Việt			X	X		X	

7. Mô tả dự án Tetris

Nhóm sẽ phát triển game Tetris cổ điển với các tính năng chính sau:

- **Xử lý đầu vào và di chuyển:** Điều khiển các khối Tetris bằng bàn phím (mũi

tên, phím quay, phím thả nhanh).

- **Thao tác khối Tetris:** Tạo và quản lý các loại khối (I, O, T, S, Z, J, L), xoay khối, xử lý va chạm.
- **Điểm và cấp độ:** Hệ thống tính điểm khi xóa hàng, tăng độ khó theo cấp độ.
- **Hiệu ứng âm thanh:** Nhạc nền, âm thanh khi xóa hàng, game over.
- **Giao diện người dùng:** Hiện thị bảng chơi, khối tiếp theo, điểm số, cấp độ.
- **Kiểm thử:** Đảm bảo game hoạt động mượt mà, không có lỗi logic hay hiển thị.

Công nghệ sử dụng: Ngôn ngữ lập trình C++ với thư viện hệ thống POSIX (termios, fcntl) để xử lý terminal và đầu vào.

8. Hiệp định nhóm

- Quyết định được đưa ra dựa trên sự đồng thuận của đa số thành viên, mọi thành viên đều được lắng nghe và có quyền nêu ý kiến.
* **Trường hợp không đạt được sự thống nhất, quyết định của trưởng nhóm sẽ là quyết định cuối cùng.**
- Tôn trọng lẫn nhau, kể cả trong trường hợp có sự khác biệt về quan điểm hoặc cách tiếp cận.
- Hỗ trợ kịp thời cho các thành viên khi gặp khó khăn về kỹ thuật, tiến độ hoặc ý tưởng.
- Đảm bảo mỗi thành viên đều có đóng góp rõ ràng, phù hợp với năng lực và công việc được phân công.

- Cam kết hoàn thành nhiệm vụ đúng thời hạn, chủ động báo cáo nếu có rủi ro hoặc chậm trễ.
- Giữ liên lạc thường xuyên, sử dụng các kênh trao đổi chính thức để cập nhật tiến độ và đưa ra quyết định.
- Sử dụng Git/GitHub để quản lý mã nguồn, commit code thường xuyên với commit message rõ ràng.
- Áp dụng quy trình phân nhánh (branching strategy):
 - Nhánh **main**: Nhánh production, chỉ team lead được phép push trực tiếp
 - Nhánh **develop**: Nhánh phát triển chính, tất cả thành viên thực hiện implementation tại đây
 - Các nhánh feature: Tạo từ develop cho từng tính năng cụ thể, merge về develop sau khi hoàn thành
- Tổ chức buổi demo tính năng (feature demo) ít nhất 1 lần/tuần để theo dõi tiến độ, kiểm tra chất lượng và điều chỉnh kế hoạch kịp thời.
- Viết code sạch, có comment giải thích khi cần, tuân thủ chuẩn coding style mà nhóm thống nhất.
- Code review lẫn nhau trước khi merge vào nhánh chính để đảm bảo chất lượng.

9. Không gian sinh hoạt và trao đổi của nhóm

Để đảm bảo quá trình làm việc diễn ra hiệu quả và có tính hệ thống, nhóm đã thiết lập kênh Slack riêng mang tên **5ducks** làm nền tảng trao đổi chính thức. Kênh Slack được sử dụng thường xuyên với các mục đích:

- Phân công nhiệm vụ và theo dõi tiến độ công việc một cách trực quan.
- Thảo luận và phản hồi ý tưởng nhanh chóng, hỗ trợ đính kèm file, hình ảnh, hoặc đoạn mã nguồn.
- Lưu trữ tài liệu, liên kết tham khảo và toàn bộ lịch sử trao đổi, giúp tra cứu dễ dàng.
- Gửi thông báo nhắc nhở về deadline và lịch họp nhóm.
- Tiến hành biểu quyết đối với các quyết định quan trọng của nhóm.

10. Tiêu chí đánh giá các thành viên

Đặc điểm	Nổi bật	Tốt	Bình thường	Kém
Thái độ làm việc	Sẵn sàng nhận nhiệm vụ và hoàn thành tốt.	Hoàn thành nhiệm vụ được giao.	Hoàn thành nhiệm vụ với sự nhắc nhở.	Không hoàn thành nhiệm vụ được giao.
Quản lý thời gian	Hoàn thành trước hạn, đúng giờ họp.	Đúng hạn, trễ dưới 5 phút.	Đúng hạn khi nhắc nhở, trễ 5–10 phút.	Trễ quá 10 phút, không hoàn thành.
Giải quyết vấn đề phát sinh	Tích cực tìm kiếm giải pháp để giải quyết các vấn đề phát sinh.	Nhờ người khác giải quyết vấn đề phát sinh.	Không giải quyết vấn đề nhưng có đưa ra ý kiến đóng góp.	Không tham gia vào vấn đề cần giải quyết.
Nêu ý kiến	Sẵn sàng nêu ý kiến.	Chỉ nêu ý kiến khi có việc cần.	Đưa ra ý kiến khi có sự nhắc nhở.	Không nêu ý kiến gì cho nhóm.

Giữ liên lạc	Luôn phản hồi tin nhắn nhóm trong vòng 30 phút.	Phản hồi tin nhắn nhóm trong vòng 1–2 giờ.	Phản hồi tin nhắn nhóm trong vòng 2–5 giờ.	Phản hồi tin nhắn nhóm sau hơn 5 giờ hoặc không phản hồi.
Chất lượng code	Code sạch, có comment, tuân thủ chuẩn, không bug.	Code hoạt động tốt, có một vài chỗ cần cải thiện nhỏ.	Code hoạt động nhưng khó đọc hoặc có bug nhỏ.	Code có nhiều bug hoặc không hoạt động đúng.

11. Cam kết

Sau khi đọc kỹ các nội dung mà hợp đồng thành lập nhóm đã nêu ra, những thành viên trong nhóm cam kết sẽ thực hiện đúng những yêu cầu đã đặt ra.





Dương Hoà Long

Lê Quang Nhật

Lê Hữu Nhị




Nguyễn Duy Thanh

Kiều Quang Việt

II Công cụ hỗ trợ và quản lý dự án

Quản lý công việc và tiến độ

- **Trello:** Quản lý task, phân công công việc, theo dõi tiến độ theo phương pháp Kanban
 - Link: <https://trello.com/invite/b/693024cd112ba6767e45fd9a/ATTI060c7059b51bb4a69e34c070c2254ff40261BBCD/5ducks>

Quản lý mã nguồn

- **GitHub:** Lưu trữ và quản lý version control của source code, áp dụng quy trình Git Flow với nhánh main và develop
 - Repository: <https://github.com/lqnhat/5ducks-tetris>

Giao tiếp và trao đổi nhóm

- **Slack:** Kênh trao đổi chính thức của nhóm, thảo luận kỹ thuật, báo cáo tiến độ, chia sẻ tài liệu.
 - Workspace: <https://app.slack.com/client/T09M5KGA799/COA0AR9KJ4X>

Soạn thảo báo cáo

- **Overleaf:** Công cụ soạn thảo LaTeX để viết báo cáo, tài liệu dự án.

III Phần giới thiệu và hướng dẫn chơi game

1 Giới thiệu về Tetris

Chào mừng bạn đến với **Tetris** - một trong những trò chơi điện tử kinh điển và được yêu thích nhất mọi thời đại!

Tetris lần đầu được tạo ra vào năm 1985 bởi Alexey Pajitnov, một kỹ sư phần mềm người Nga. Từ đó đến nay, Tetris đã trở thành biểu tượng văn hóa đại chúng, xuất hiện trên hầu hết mọi nền tảng từ máy tính, điện thoại di động đến máy chơi game cầm tay. Sức hấp dẫn của Tetris nằm ở gameplay đơn giản nhưng cực kỳ gây nghiện - bất cứ ai cũng có thể chơi được sau vài phút làm quen, nhưng để trở thành cao thủ lại cần sự rèn luyện và chiến thuật.

2 Câu chuyện đằng sau dự án

Phiên bản Tetris này được phát triển bởi nhóm **5 Ducks** với mục tiêu tái hiện lại trải nghiệm chơi game cổ điển trên terminal. Chúng tôi sử dụng ngôn ngữ lập trình C++ thuần túy kết hợp với các thư viện hệ thống POSIX để mang đến trải nghiệm chơi game mượt mà ngay trên terminal của bạn.

Điểm đặc biệt của dự án là chúng tôi phát triển **hai phiên bản song song**:

- **Phiên bản Struct** (Procedural Programming): Sử dụng struct và các hàm để

tổ chức code theo phong cách lập trình thủ tục truyền thống

- **Phiên bản Class** (Object-Oriented Programming): Áp dụng các nguyên lý lập trình hướng đối tượng với class, encapsulation và inheritance

Việc phát triển hai phiên bản giúp chúng tôi:

- So sánh và hiểu sâu hơn về các paradigm lập trình khác nhau
- Rèn luyện kỹ năng tổ chức code và thiết kế kiến trúc phần mềm
- Cung cấp cho người dùng sự lựa chọn về phong cách code họ muốn nghiên cứu

Dự án này không chỉ là một trò chơi giải trí, mà còn là minh chứng cho kỹ năng lập trình, làm việc nhóm và quản lý dự án của chúng tôi trong môn học Kỹ Năng Nghề Nghiệp.

3 Luật chơi cơ bản

Mục tiêu

Mục tiêu của Tetris rất đơn giản: *xếp các khối rơi xuống sao cho tạo thành hàng ngang hoàn chỉnh*. Khi một hàng được lấp đầy, nó sẽ biến mất và bạn sẽ nhận được điểm. Trò chơi kết thúc khi các khối chồng lên nhau đạt tới đỉnh màn hình.

Các khối Tetris (Tetromino)

Có 7 loại khối cơ bản trong Tetris, mỗi loại có hình dạng và màu sắc riêng:

- **Khối I (I-Block):** Hình thanh dài 4 ô - Màu xanh dương (Cyan)
 - Khối này cực kỳ hữu ích để xóa 4 hàng cùng lúc (gọi là "Tetris")
 - Chiến thuật: Để dành một cột trống bên cạnh để chờ khối I
- **Khối O (O-Block):** Hình vuông 2×2 - Màu vàng (Yellow)
 - Khối duy nhất không thể xoay
 - Dễ sử dụng để lấp đầy các khoảng trống lớn
- **Khối T (T-Block):** Hình chữ T - Màu tím (Purple)
 - Linh hoạt và dễ sử dụng
 - Có thể tạo "T-Spin" - kỹ thuật nâng cao để ghi điểm cao
- **Khối S (S-Block):** Hình chữ S - Màu xanh lá (Green)
 - Tạo các đường ziczac
 - Cần sử dụng cẩn thận để tránh tạo khoảng trống khó lấp
- **Khối Z (Z-Block):** Hình chữ Z ngược - Màu đỏ (Red)
 - Giống khối S nhưng hướng ngược lại
 - Kết hợp với khối S để tạo bề mặt phẳng
- **Khối J (J-Block):** Hình chữ J - Màu xanh đậm (Blue)
 - Hữu ích để lấp đầy các góc

- Có thể tạo nhiều combo khi sử dụng khéo léo
- **Khối L (L-Block):** Hình chữ L - Màu cam (Orange)
 - Đối xứng với khối J
 - Dễ kết hợp với nhiều loại khối khác

4 Hướng dẫn điều khiển

Game được thiết kế với các phím điều khiển trực quan và dễ nhớ:

Phím	Chức năng
A hoặc ← (Mũi tên trái)	Di chuyển khối sang trái
D hoặc → (Mũi tên phải)	Di chuyển khối sang phải
S hoặc ↓ (Mũi tên xuống)	Tăng tốc độ rơi (Soft Drop)
W hoặc ↑ (Mũi tên lên)	Xoay khối theo chiều kim đồng hồ
Space (Phím cách)	Thả khối xuống ngay lập tức (Hard Drop)
P	Tạm dừng/Tiếp tục game
Q hoặc ESC	Thoát game

Mẹo nhỏ: Bạn có thể giữ phím di chuyển để khối tự động di chuyển liên tục theo hướng đó!

5 Hệ thống tính điểm

Điểm số trong Tetris được tính dựa trên số hàng bạn xóa được trong một lần:

Hành động	Số hàng xóa	Điểm cơ bản
Single (Đơn)	1 hàng	100 điểm
Double (Đôi)	2 hàng	300 điểm
Triple (Ba)	3 hàng	500 điểm
Tetris	4 hàng	800 điểm

Bonus theo cấp độ: Điểm số sẽ được nhân với level hiện tại của bạn. Ví dụ: Xóa 4 hàng ở level 5 sẽ cho bạn $800 \times 5 = 4000$ điểm!

Combo: Nếu bạn xóa nhiều hàng liên tiếp trong các lượt liên tiếp nhau, bạn sẽ nhận được điểm thưởng combo. Combo càng dài, điểm thưởng càng cao!

6 Cấp độ và độ khó

Tetris của chúng tôi có hệ thống cấp độ động:

- **Level 1-3:** Tốc độ rơi chậm, thích hợp cho người mới bắt đầu làm quen với game
- **Level 4-6:** Tốc độ trung bình, đòi hỏi phản xạ tốt và chiến thuật hợp lý

- **Level 7-9:** Tốc độ nhanh, chỉ dành cho người chơi có kinh nghiệm
- **Level 10+:** Tốc độ cực nhanh, thử thách giới hạn của bạn!

Mỗi khi bạn xóa được **10 hàng**, level sẽ tăng lên 1 và tốc độ rơi của các khối sẽ nhanh hơn. Điều này tạo nên sự thử thách không ngừng và khiến mỗi ván chơi đều gay cấn!

7 Chiến thuật và mẹo chơi hay

1. Giữ bề mặt phẳng

Luôn cố gắng giữ các khối ở cùng một độ cao. Tránh tạo ra các cột cao vút hoặc ”hố sâu” khó lấp đầy. Bề mặt phẳng giúp bạn linh hoạt hơn khi xếp các khối tiếp theo.

2. Để dành cột cho khối I

Một chiến thuật kinh điển là để dành một cột dọc bên cạnh (thường là cột ngoài cùng bên phải hoặc trái) để chờ khối I xuất hiện. Khi có khối I, bạn có thể xóa 4 hàng cùng lúc và ghi điểm cao (Tetris)!

3. Quan sát khối tiếp theo

Game luôn hiển thị khối tiếp theo sẽ xuất hiện. Hãy lợi dụng thông tin này để lên kế hoạch trước cho vị trí đặt khối hiện tại.

4. Không vội vàng

Ở level thấp, bạn có nhiều thời gian để suy nghĩ. Đừng vội thả khối xuống nếu chưa chắc chắn. Hãy tận dụng thời gian để tìm vị trí tối ưu nhất.

5. Tập trung vào việc tồn tại lâu hơn

Đôi khi, việc xóa nhiều hàng cùng lúc không phải là ưu tiên số 1. Nếu tình huống nguy cấp, hãy tập trung vào việc giảm độ cao của đồng khối xuống, ngay cả khi bạn chỉ xóa được 1-2 hàng.

6. Luyện tập xoay khối nhanh

Thành thạo việc xoay khối sẽ giúp bạn tiết kiệm thời gian quý báu, đặc biệt ở level cao. Hãy dành thời gian làm quen với cách các khối xoay.

8 Tính năng đặc biệt

- **Âm thanh sống động:** Nhạc nền Tetris kinh điển và hiệu ứng âm thanh khi xóa hàng, tạo không khí sôi động.
- **Bảng xếp hạng:** Theo dõi điểm số cao nhất của bạn và thử thách bản thân phá kỷ lục.
- **Tạm dừng game:** Cần nghỉ một chút? Nhấn P để tạm dừng bất cứ lúc nào.
- **Hiển thị thống kê:** Xem số hàng đã xóa, level hiện tại và thời gian chơi.

9 Cài đặt và khởi động game

Yêu cầu hệ thống

- **Hệ điều hành:** macOS 10.14+ hoặc Linux (Ubuntu 20.04+, Fedora 30+, Debian 10+)
- **CPU:** Intel Core i3 hoặc tương đương
- **RAM:** 2GB trở lên
- **Dung lượng ổ cứng:** 50MB không gian trống
- **Terminal:** Hỗ trợ ANSI escape codes
- **Compiler:** GCC 7.0+ hoặc Clang 5.0+ với hỗ trợ C++11

Lưu ý quan trọng: Phiên bản hiện tại chỉ hỗ trợ hệ điều hành Unix-based (macOS và Linux). Game chưa hỗ trợ Windows do sử dụng các thư viện hệ thống POSIX (termios, fcntl) cho xử lý terminal.

Hướng dẫn cài đặt

Game có **hai phiên bản** để bạn lựa chọn: Struct (Procedural) và Class (OOP). Cả hai đều có gameplay giống nhau, chỉ khác về cách tổ chức code.

Bước 1: Tải xuống mã nguồn từ GitHub repository của chúng tôi: <https://github.com/lqnhat/5ducks-tetris>

Bước 2: Mở Terminal và di chuyển đến thư mục chứa mã nguồn:

- Sử dụng lệnh: `cd /path/to/5ducks-tetris`

Bước 3: Chọn phiên bản bạn muốn chơi:

Option A - Phiên bản Struct (Procedural):

- Di chuyển vào thư mục: `cd tetris_struct`
- Biên dịch: `g++ -std=c++11 main.cpp -o tetris`
- Chạy game: `./tetris`

Option B - Phiên bản Class (OOP):

- Di chuyển vào thư mục: `cd tetris_class`
- Biên dịch: `g++ -std=c++11 main.cpp -o tetris`
- Chạy game: `./tetris`

Lưu ý: Nếu thư mục có Makefile, bạn có thể sử dụng lệnh `make` thay vì lệnh `g++` ở trên.

Bước 4: Đảm bảo terminal có kích thước đủ lớn (tối thiểu 80×24 characters)

Bước 5: Thưởng thức game!

10 Câu hỏi thường gặp (FAQ)

Q: Sự khác biệt giữa phiên bản Struct và Class là gì?

A: Cả hai phiên bản đều có gameplay và tính năng giống hệt nhau. Điểm khác biệt nằm ở cách tổ chức code:

- **Phiên bản Struct:** Sử dụng lập trình thủ tục (Procedural Programming) với struct và các hàm độc lập. Phù hợp với người mới học C++ hoặc muốn hiểu cách tiếp cận lập trình truyền thống.
- **Phiên bản Class:** Áp dụng lập trình hướng đối tượng (OOP) với class, encapsulation và các nguyên lý OOP. Phù hợp cho người muốn học về thiết kế phần mềm và OOP.

Q: Tôi nên chọn phiên bản nào?

A: Nếu bạn mới bắt đầu học C++, phiên bản Struct dễ hiểu hơn. Nếu bạn muốn học OOP hoặc đang nghiên cứu về kiến trúc phần mềm, chọn phiên bản Class. Bạn cũng có thể chơi cả hai để so sánh!

Q: Game có chạy trên Windows không?

A: Hiện tại game chỉ hỗ trợ macOS và Linux. Phiên bản Windows đang được phát triển và sẽ ra mắt trong tương lai. Nếu bạn dùng Windows, bạn có thể sử dụng WSL (Windows Subsystem for Linux) để chạy game.

Q: Game có chạy trên điện thoại không?

A: Phiên bản hiện tại chỉ hỗ trợ máy tính (macOS, Linux). Chúng tôi có kế hoạch phát triển phiên bản mobile trong tương lai.

Q: Tôi gặp lỗi khi biên dịch, phải làm sao?

A: Đảm bảo bạn đã cài đặt compiler C++ (g++ hoặc clang) và hỗ trợ C++11. Trên Ubuntu/Debian, chạy: `sudo apt-get install build-essential`. Trên macOS, cài đặt Xcode Command Line Tools: `xcode-select --install`.

Q: Terminal của tôi không hiển thị đúng màu sắc?

A: Đảm bảo terminal của bạn hỗ trợ ANSI escape codes. Hầu hết các terminal hiện đại (Terminal.app trên macOS, GNOME Terminal, iTerm2) đều hỗ trợ. Nếu vẫn gặp vấn đề, thử terminal khác.

Q: Game bị giật hoặc phím bấm không phản hồi?

A: Thử các cách sau:

- Đóng các ứng dụng terminal khác đang chạy
- Tăng kích thước buffer của terminal
- Đảm bảo terminal không bị lag do quá nhiều process

- Khởi động lại terminal và chạy lại game

11 Liên hệ và hỗ trợ

Nếu bạn gặp bất kỳ vấn đề nào khi chơi game hoặc có góp ý, đề xuất, đừng ngại liên hệ với chúng tôi:

- **GitHub Issues:** <https://github.com/lqnhat/5ducks-tetris/issues>
- **Slack Community:** <https://app.slack.com/client/T09M5KGA799/C0A0AR9KJ4X>

Chúng tôi rất mong nhận được phản hồi từ bạn để cải thiện game ngày càng tốt hơn!

12 Tài liệu tham khảo

1. Pajitnov, A. (1985). *Tetris - Game Design and Implementation*. Soviet Academy of Sciences, Moscow.
2. The Linux Programming Interface Documentation. (n.d.). *termios - Terminal I/O*. Retrieved from <https://man7.org/linux/man-pages/man3/termios.3.html>
3. GitHub Documentation. (n.d.). *GitHub Flow - Understanding the GitHub workflow*. Retrieved from <https://docs.github.com/en/get-started/quickstart/github-flow>

4. Stevens, W. R., & Rago, S. A. (2013). *Advanced Programming in the UNIX Environment* (3rd ed.). Addison-Wesley Professional.
5. Stroustrup, B. (2013). *The C++ Programming Language* (4th ed.). Addison-Wesley Professional.
6. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional.

13 Lời kết

Cảm ơn bạn đã chọn chơi Tetris phiên bản của **5 Ducks**! Chúng tôi hy vọng bạn sẽ có những giây phút giải trí thú vị và đầy thử thách. Hãy nhớ rằng, Tetris không chỉ là một trò chơi - nó còn rèn luyện khả năng tư duy logic, phản xạ nhanh nhạy và kỹ năng quản lý không gian.

Trong Tetris cũng như trong cuộc sống, những thành tựu (achievements) biến mất, còn những sai lầm (mistakes) thì tích lũy lại.” - Một triết lý thú vị mà nhiều người chơi Tetris nhận ra.

Chúc bạn chơi game vui vẻ và đạt điểm cao!

— Đội ngũ phát triển 5 Ducks —

Dương Hoà Long • Lê Quang Nhật • Lê Hữu Nhị

Nguyễn Duy Thanh • Kiều Quang Việt