Chân trời sáng tạo

HOÀNG VĂN KIẾM (Tổng Chủ biên)
NGÔ QUỐC VIỆT – HOÀNG NGỌC THANH (đồng Chủ biên)
TRẦN QUANG VĨNH CHÁNH – PHAN MAI CHI – TRẦN HÀM DƯƠNG
LỤC VĂN HÀO – NGUYỄN TRUNG TIẾN – HUỲNH NGỌC TÍN

TIN HỌC

ĐỊNH HƯỚNG KHOA HỌC MÁY TÍNH

10

NHÀ XUẤT BẢN GIÁO DỤC VIỆT NAM

BẢN MẪU

HỘI ĐỒNG QUỐC GIA THẨM ĐỊNH SÁCH GIÁO KHOA

Môn: Tin học – Lớp 10

(Theo Quyết định số 2038/QĐ-BGDĐT ngày 31 tháng 7 năm 2024 của Bộ trưởng Bộ Giáo dục và Đào tạo)

Chủ tịch: LÊ HOÀI BẮC
Phó Chủ tịch: TRẦN ĐĂNG HƯNG
Uỷ viên, Thư kí: HỒ VĨNH THẮNG

Các uỷ viên: NGUYỄN TRUNG TRỰC – TRẦN CAO ĐỆ
QUÁCH XUÂN TRƯỜNG – ĐỖ TRUNG KIÊN
NGUYỄN THỊ VÂN KHÁNH – PHAN THỊ MAY
HOÀNG VĂN QUYẾN – HOÀNG XUÂN THẮNG

Chân trời sáng tạo

HOÀNG VĂN KIẾM (Tổng Chủ biên)
NGÔ QUỐC VIỆT – HOÀNG NGỌC THANH (đồng Chủ biên)
TRẦN QUANG VĨNH CHÁNH – PHAN MAI CHI – TRẦN HÀM DƯƠNG
LỤC VĂN HÀO – NGUYỄN TRUNG TIẾN – HUỲNH NGỌC TÍN

TIN HỌC

ĐỊNH HƯỚNG KHOA HỌC MÁY TÍNH

10

Chân trời sáng tạo

NHÀ XUẤT BẢN GIÁO DỤC VIỆT NAM

HƯỚNG DẪN SỬ DỤNG SÁCH

Mỗi bài học đều được thiết kế bao gồm mục tiêu và các hoạt động dạy và học. Các hoạt động trọng tâm được gắn thêm hình ảnh nhận diện là các "biểu tượng" hay "icon".

MỤC TIÊU là những gì em sẽ đạt được sau bài học. Bắt đầu vào bài học, em cần đọc mục tiêu để biết các yêu cầu của bài học. Trước khi kết thúc bài học, em cần so sánh những gì đã học được với mục tiêu của bài.

KHỞI ĐỘNG là hoạt động để gợi mở, tạo hứng thú học tập và định hướng cho các em suy nghĩ, khám phá nội dung bài học. Em sẽ giải quyết được vấn đề đặt ra ở phần này khi tìm hiểu nội dung ở phần **Khám phá**.

KHÁM PHÁ là nội dung chính của bài học. Trong đó, **Đọc và quan sát**, **Làm** và **Ghi nhớ** là ba hoạt động cần thực hiện để hoàn thành cơ bản các nhiệm vụ học tập.

- Đọc và quan sát - gặp biểu tượng này, em cần đọc, quan sát để tìm hiểu kiến thức, kĩ năng mới của bài học.
- Làm - thực hiện các yêu cầu để hoàn thành nhiệm vụ học tập này giúp em khám phá, lĩnh hội kiến thức, kĩ năng mới của bài học.
- Ghi nhớ - tóm tắt ngắn gọn kiến thức, kĩ năng trọng tâm của mỗi phần nội dung bài học mà em cần ghi nhớ.

LUYỆN TẬP là nội dung gồm các câu hỏi, bài tập để củng cố kiến thức, kĩ năng trong bài học.

THỰC HÀNH là hoạt động rèn luyện thao tác sử dụng thiết bị máy tính, phần mềm tin học.

VẬN DỤNG là nội dung gồm các câu hỏi, bài tập, tình huống, vấn đề thực tiễn mà em cần vận dụng kiến thức, kĩ năng vừa học để giải quyết.

Em có biết là mục cung cấp cho học sinh một số thông tin bổ sung, mở rộng, nâng cao liên quan đến nội dung bài học.

Chân trời sáng tạo

Ngoài ra:

Các hình ảnh trong sách không chỉ là minh hoạ mà còn là một phần quan trọng của nội dung học tập. Các em cần "đọc" được nội dung của hình ảnh (quan sát, tìm hiểu, so sánh,...) để hoàn thành nhiệm vụ học tập. Kĩ năng có được của các em thông qua quá trình làm việc với kênh hình (kênh thông tin về hình ảnh) là yếu tố quan trọng để phát triển năng lực tự tìm hiểu, khám phá phần mềm máy tính trong môn Tin học.

Các chữ số đặt trong vòng tròn (1, 2, 3,...) được dành riêng để đánh số thứ tự các thao tác, công việc cần được thực hiện theo trình tự. Điều này giúp các em dễ dàng nhận biết các bước thực hiện nhiệm vụ và thuận tiện để đối chiếu, tra cứu khi thực hành trên máy tính.

Hãy bảo quản, giữ gìn Sách giáo khoa để dành tặng các em học sinh lớp sau!

LỜI NÓI ĐẦU

Các em học sinh thân mến!

Chuyển đổi số (digital transformation) là xu hướng tất yếu trong thời kì cách mạng công nghiệp 4.0, yêu cầu sự kết hợp mạnh mẽ giữa công nghệ thông tin với các ngành kinh doanh và công nghiệp truyền thống. Định hướng Tin học ứng dụng (ICT) trở thành một nhân tố quan trọng trong việc xây dựng hệ thống thông tin, quản lí dữ liệu, giao dịch điện tử và truyền thông của các công ty và tổ chức. Định hướng Khoa học máy tính (CS) đóng vai trò không thể thiếu trong việc phát triển phần mềm, xử lí dữ liệu, nghiên cứu ứng dụng trí tuệ nhân tạo để tạo ra các giải pháp công nghiệp thông minh và tự động. Tuỳ theo sở thích và định hướng nghề nghiệp tương lai, học sinh sẽ lựa chọn học theo một trong hai định hướng trên.

Bộ sách giáo khoa Chân trời sáng tạo, môn Tin học lớp 10 gồm bốn quyển: Tin học 10 – Định hướng Tin học ứng dụng, Tin học 10 – Định hướng Khoa học máy tính, Chuyên đề học tập Tin học 10 – Định hướng Tin học ứng dụng, Chuyên đề học tập Tin học 10 – Định hướng Khoa học máy tính.

Bộ sách được biên soạn bám sát Chương trình Giáo dục phổ thông 2018 của Bộ Giáo dục và Đào tạo, được cập nhật những thông tin khoa học kĩ thuật mới trên thế giới trong các hoạt động học tập, giúp các em thích ứng, hoà nhập với xã hội hiện đại, hình thành và phát triển năng lực tự học, là công cụ hiệu quả hỗ trợ biến việc học thành tự học suốt đời.

Sách giáo khoa **Tin học 10 – Định hướng Khoa học máy tính** là nội dung cốt lõi của định hướng CS với 5 chủ đề A, B, D, F, G và 32 bài học. Trong đó, toàn bộ 03 bài cuối của chủ đề A được dành riêng cho định hướng Khoa học máy tính với thời lượng 07 tiết.

Quyển sách được tích hợp các hoạt động dạy học phát triển năng lực, khuyến khích làm việc theo nhóm, giúp phát triển bộ kĩ năng mềm, kích thích sự tò mò, sáng tạo, tạo động lực, niềm đam mê để các em tiếp tục tìm hiểu, khám phá và mở rộng hiểu biết.

Chúc các em học tốt và vận dụng hiệu quả kiến thức, kĩ năng vào học tập và thực tiễn cuộc sống!

CÁC TÁC GIẢ

MỤC LỤC

Hướng dẫn sử dụng sách 2
Lời nói đầu 3
Mục lục 4

CHỦ ĐỀ A. MÁY TÍNH VÀ XÃ HỘI TRI THỨC 6

Bài A1. Dữ liệu và thông tin 6
Bài A2. Tin học và xã hội 11
Bài A3. Vai trò của thiết bị thông minh trong xã hội và cuộc cách mạng công nghiệp 4.0 18
Bài A4. Thực hành sử dụng một số thiết số thông dụng 23
Bài A5. Hệ nhị phân trong tin học 29
Bài A6. Số hoá hình ảnh, âm thanh 35
Bài A7. Số hoá văn bản, bảng mã chuẩn quốc tế 41

CHỦ ĐỀ B. MẠNG MÁY TÍNH VÀ INTERNET 46

Bài B1. Mạng máy tính, Internet và tác động đến cuộc sống 46
Bài B2. Internet vạn vật và điện toán đám mây 50
Bài B3. Sử dụng một số dịch vụ tiện ích trên máy tính và Internet 53
Bài B4. Sử dụng Internet an toàn 56

CHỦ ĐỀ D. ĐẠO ĐỨC, PHÁP LUẬT VÀ VĂN HOÁ TRONG MÔI TRƯỜNG SỐ 64

Bài D1. Vấn đề bản quyền và giao tiếp trong môi trường số 64
Bài D2. Luật công nghệ thông tin và luật an ninh mạng 71

Chân trời sáng tạo

CHỦ ĐỀ F. GIẢI QUYẾT VẤN ĐỀ VỚI SỰ TRỢ GIÚP CỦA MÁY TÍNH.....77

Bài F1. Ngôn ngữ lập trình bậc cao và môi trường lập trình Python 77

Bài F2. Biến, hằng và biểu thức số học 82

Bài F3. Các lệnh vào/ra và chuyển đổi kiểu dữ liệu..... 87

Bài F4. Thực hành biến, hằng, biểu thức và các lệnh vào/ra đơn giản..... 92

Bài F5. Lệnh rẽ nhánh 95

Bài F6. Thực hành lệnh rẽ nhánh..... 102

Bài F7. Lệnh lặp 106

Bài F8. Thực hành lệnh lặp 112

Bài F9. Xâu kí tự 116

Bài F10. Thực hành xử lí xâu 121

Bài F11. Danh sách..... 124

Bài F12. Một số hàm trên danh sách 131

Bài F13. Hàm trong Python 136

Bài F14. Tham số của hàm và phạm vi của biến 143

Bài F15. Thực hành hàm trong Python..... 150

Bài F16. Kiểm thử và gỡ lỗi chương trình..... 155

Bài F17. Thực hành tổng hợp 163

CHỦ ĐỀ G. HƯỚNG NGHIỆP VỚI TIN HỌC.....168

Bài G1. Nhóm nghề thiết kế đồ hoạ 168

Bài G2. Nhóm nghề lập trình 173

Phụ lục 179

Bảng giải thích thuật ngữ 180

Chân trời sáng tạo

Chủ đề A

MÁY TÍNH VÀ XÃ HỘI TRI THỨC

BÀI A1 DỮ LIỆU VÀ THÔNG TIN

MỤC TIÊU **Sau bài học này, em sẽ:**

- Phân biệt được dữ liệu và thông tin.
- Chuyển đổi được giữa các đơn vị lưu trữ dữ liệu.
- Nêu được sự ưu việt của việc lưu trữ, xử lí và truyền thông tin bằng thiết bị số.

KHỞI ĐỘNG

Trong chương trình Tin học 6, em đã được tìm hiểu về thông tin và dữ liệu. Từ các dữ liệu ghi nhận từ thứ Hai đến thứ Sáu, mỗi ngày một lần về nhiệt độ (28°C, 30°C, 29°C, 31°C, 32°C), độ ẩm (70%, 65%, 75%, 60%, 68%) và lượng mưa (0 mm, 5 mm, 0 mm, 10 mm, 0 mm), em có được thông tin về tình hình thời tiết của tuần vừa qua là nóng ẩm và có một ít mưa nhỏ. Em hãy tìm hiểu và giới thiệu với các bạn thêm một số ví dụ về thông tin và dữ liệu.

KHÁM PHÁ

1. Mối quan hệ giữa thông tin và dữ liệu

Thông tin giúp chúng ta tăng thêm kiến thức, hiểu biết về thế giới xung quanh và chính bản thân mình. Thông tin được thể hiện ở nhiều dạng khác nhau. Ví dụ, thông tin về dự báo thời tiết tại địa phương có thể được thể hiện ở các dạng như văn bản (tin dự báo thời tiết trên báo địa phương), hình ảnh và âm thanh (chương trình truyền thanh, truyền hình về dự báo thời tiết trên các đài phát thanh và truyền hình địa phương).

Thông tin có được từ việc xử lí, tổng hợp, phân tích dữ liệu. Máy tính là thiết bị hỗ trợ xử lí dữ liệu để tạo ra thông tin, hỗ trợ con người trong quá trình nhận thức. Theo các cách xử lí khác nhau, từ một nguồn dữ liệu có thể cho ra nhiều thông tin khác nhau. Ví dụ, từ nguồn dữ liệu quan trắc khí tượng thuỷ văn tại địa phương trong nhiều năm (nhiệt độ, lượng mưa, độ ẩm,…), sau khi chọn lọc và xử lí, máy tính và phần mềm chuyên dụng về thời tiết có thể cung cấp cho người dùng thông tin về xu hướng nhiệt độ theo mùa trong năm, thông tin dự báo thời tiết phục vụ sản xuất nông nghiệp, thông tin hỗ trợ hoạch định xây dựng cơ sở hạ tầng như hệ thống thuỷ lợi,…

Nếu dữ liệu không đầy đủ, thông tin nhận được có thể sẽ không chính xác. Ví dụ, với dữ liệu 30°C, nếu được đặt cùng các dữ liệu của bệnh án thì cho thông tin "bệnh nhân đang bị hạ thân nhiệt nghiêm trọng", nếu được đặt cùng với các dữ liệu quan trắc khí tượng thuỷ văn thì lại cho thông tin "nhiệt độ ngoài trời tương đối nóng",... Như vậy, thông tin chỉ được hiểu đúng khi có đủ dữ liệu.

Thông tin là ý nghĩa của dữ liệu và dữ liệu là yếu tố thể hiện, xác định thông tin. Dữ liệu không mang ý nghĩa cụ thể khi đứng đơn lẻ. Thông tin là các dữ liệu đã được xử lí, tổng hợp, mang ý nghĩa và hữu ích đối với người sử dụng, hỗ trợ việc ra quyết định, giúp hiểu rõ hơn về các sự kiện, tình huống cụ thể. Quá trình chuyển dữ liệu thành thông tin bằng máy tính bao gồm các bước như *Hình 1*.

Hình 1. Quá trình chuyển dữ liệu thành thông tin bằng máy tính

1 **Nhập dữ liệu:** Dữ liệu được đưa vào máy tính thông qua các thiết bị (máy quét, máy chụp ảnh, máy ghi âm,...) hoặc do con người nhập từ bàn phím.

2 **Xử lí dữ liệu:** Máy tính và các phần mềm ứng dụng xử lí và phân tích dữ liệu để tạo ra thông tin hay dữ liệu mới. Ví dụ, từ bảng dữ liệu lưu trữ điểm kiểm tra định kì môn Tin học khối lớp 10 trong 5 năm gần nhất, phần mềm phân tích dữ liệu đánh giá được xu hướng học tập của học sinh đối với môn Tin học tại nhà trường.

3 **Xuất kết quả:** Máy tính có thể xuất kết quả thành các văn bản, hình ảnh, âm thanh mà con người có thể hiểu được (dữ liệu được chuyển thành thông tin). Ngoài ra máy tính cũng có thể xuất kết quả là dữ liệu đầu vào cho các hoạt động xử lí khác để đưa ra thông tin có mức độ phân tích tổng hợp cao hơn.

Chân trời sáng tạo

1. Để thiết kế chương trình cho hoạt động tham quan các danh lam thắng cảnh, trong các dữ liệu quan trắc khí tượng thuỷ văn tại địa phương, đơn vị du lịch cần quan tâm đến những dữ liệu nào để rút ra được những thông tin hữu ích?

2. Đầu năm học mới, để nắm được thông tin đánh giá về năng lực học tập của các học sinh trong lớp 10A1, giáo viên chủ nhiệm cần tìm hiểu và phân tích những dữ liệu gì?

- Thông tin là ý nghĩa của dữ liệu. Dữ liệu là yếu tố thể hiện, xác định thông tin. Cùng một nguồn dữ liệu có thể xử lí ra được nhiều thông tin khác nhau. Thông tin được thể hiện bởi nhiều dạng khác nhau (văn bản, âm thanh, hình ảnh).
- Quá trình chuyển dữ liệu thành thông tin trong máy tính được xử lí qua các bước: nhập dữ liệu, xử lí dữ liệu, xuất kết quả.

2. Đơn vị lưu trữ dữ liệu

Dữ liệu được lưu trữ trong bộ nhớ máy tính theo từng nhóm bit (các tín hiệu đơn lẻ nhận giá trị 0 hoặc 1). Máy tính ngày nay đều tổ chức lưu trữ, truy cập và xử lí dữ liệu theo đơn vị lưu trữ có độ dài là bội của byte (8 bit) như 2 byte (16 bit), 4 byte (32 bit) hay 8 byte (64 bit). Byte được xem là đơn vị lưu trữ dữ liệu trong máy tính (*Bảng 1*).

Bảng 1. Một số đơn vị lưu trữ dữ liệu

Đơn vị	Kí hiệu	Lượng dữ liệu
Bit	b	1 b
Byte	B	8 b
Kilobyte	KB	2^{10} B
Megabyte	MB	2^{20} B
Gigabyte	GB	2^{30} B
Terabyte	TB	2^{40} B
Petabyte	PB	2^{50} B
Exabyte	EB	2^{60} B
Zettabyte	ZB	2^{70} B
Yottabyte	YB	2^{80} B

Bắt đầu từ đơn vị KB, các đơn vị lưu trữ dữ liệu hơn kém nhau $2^{10} = 1024$ lần.

Bạn Huy vừa được trang bị một ổ cứng có dung lượng 1 TB.

a) Hỏi ổ cứng của bạn Huy có dung lượng là bao nhiêu MB?

b) Biết rằng mỗi tệp ảnh chất lượng cao có dung lượng khoảng 5 MB. Hỏi bạn Huy có thể lưu trữ tối đa bao nhiêu tệp ảnh chất lượng cao trong ổ cứng trên?

Đơn vị lưu trữ dữ liệu trong máy tính là byte (8 bit). Bắt đầu từ KB, các đơn vị lưu trữ dữ liệu hơn kém nhau 2^{10} lần.

3. Lưu trữ, xử lí và truyền thông tin bằng thiết bị số

Theo Luật Công nghệ thông tin của Việt Nam, thiết bị số bao gồm các loại thiết bị như máy tính (máy tính để bàn, máy tính xách tay, máy tính bảng); thiết bị viễn thông (thiết bị di động có khả năng kết nối Internet và chạy các ứng dụng); thiết bị truyền dẫn, thu phát sóng vô tuyến điện và các thiết bị tích hợp khác. Với sự phát triển của công nghệ thông tin, việc lưu trữ, xử lí và truyền thông tin bằng thiết bị số đã đem lại sự thay đổi trong nhiều lĩnh vực của đời sống.

Về lưu trữ: Thiết bị số cho phép người dùng lưu trữ dữ liệu một cách hiệu quả, tiết kiệm chi phí và không gian. Ví dụ, nếu ảnh chụp từng trang sách của một cuốn sách có tổng dung lượng là 50 MB thì một đĩa cứng (*Hình 2*) có dung lượng lưu trữ 8 TB có thể lưu trữ dữ liệu của hơn 160 000 cuốn sách, tương đương với số lượng sách trong thư viện St. Florian Monastery (Áo) (*Hình 3*). Sử dụng thiết bị số để lưu trữ dữ liệu là một giải pháp gọn nhẹ, chi phí thấp, dễ dàng chia sẻ, di chuyển và sao lưu. Với khả năng kết nối các thiết bị số với nhau thông qua Internet, nhiều kho lưu trữ dữ liệu khổng lồ đã được hình thành, giúp bảo tồn và phát triển những thành tựu nghiên cứu của nhân loại.

Hình 2. Đĩa cứng dung lượng 8 TB

Hình 3. Thư viện St. Florian Monastery (Áo)

Về xử lí: Máy tính và các thiết bị số giúp xử lí dữ liệu một cách nhanh chóng và cung cấp thông tin đáng tin cậy. Với sự phát triển của các yếu tố công nghệ như bộ vi xử lí, công nghệ lưu trữ, công nghệ mạng,… tốc độ xử lí dữ liệu ngày càng được nâng cao. Ngày nay, máy tính sử dụng cho văn phòng đạt 50 đến 100 tỉ phép tính mỗi giây, siêu máy tính Fugaku (Nhật Bản) đạt 422 triệu tỉ phép tính mỗi giây, siêu máy tính Frontier (Hoa Kỳ), một kì tích của khoa học công nghệ, tại thời điểm năm 2022, đạt 1,102 tỉ tỉ phép tính mỗi giây (*Hình 4*). Các siêu máy tính đang hỗ trợ nhân loại giải quyết nhiều bài toán quan trọng trong nghiên cứu khoa học, mô phỏng và dự đoán với khối lượng dữ liệu cần tính toán cực kì lớn.

Hình 4. Siêu máy tính Frontier (Hoa Kỳ, 2022), đạt tốc độ trên 1,102 tỉ tỉ phép tính mỗi giây[1]

Về truyền thông tin: Thiết bị số cho phép truyền thông tin qua mạng một cách nhanh chóng và tiện lợi. Từ việc chỉ có thể truyền được 3-5 từ mỗi phút (sử dụng điện báo trong thời kì đầu), phát triển đến tốc độ 64 kbps (truyền 64 000 bit dữ liệu mỗi giây, sử dụng mạng điện thoại) và hiện nay là khoảng 1 Gpbs (truyền 1 tỉ bit dữ liệu mỗi giây, sử dụng mạng 4G). Người dùng đã có thể sử dụng các thiết bị di động thông minh để truy cập và tìm kiếm dữ liệu từ các kho lưu trữ khổng lồ trên mạng Internet, xem phim qua Internet trên mạng di động, thực hiện các cuộc gọi truyền hình giúp kết nối các thành viên trong gia đình, trong tập thể lại với nhau,…

[1] Nguồn ảnh: OLCF at ORNL.

1. Theo em, đồng hồ điện tử (không phải đồng hồ thông minh) có phải là thiết bị số hay không? Hãy tìm hiểu và giới thiệu về một thiết bị số mà em biết.

2. Để lưu trữ toàn bộ tệp ảnh, video các sự kiện của người thân trong gia đình, em cần sử dụng các thiết bị số nào? Em hãy tìm hiểu và trao đổi với các bạn trong lớp về giải pháp lưu trữ trên mạng và các dịch vụ lưu trữ trên mạng phù hợp với khả năng tài chính và nhu cầu của mình.

Với dung lượng lưu trữ lớn, tốc độ xử lí cao và khả năng kết nối trực tuyến, thiết bị số đã cùng công nghệ thông tin góp phần mang lại nhiều thay đổi tích cực trong việc lưu trữ, xử lí và truyền thông tin trong cuộc sống ngày nay.

LUYỆN TẬP

1. Khi đề cập đến dữ liệu, phát biểu nào dưới đây là đúng?

A. Là số liệu hoặc tài liệu chưa qua xử lí.

B. Dữ liệu được thu thập từ Internet.

C. Là thông tin được lưu trữ trong máy tính.

D. Dữ liệu và thông tin là hai khái niệm tương đồng nhau.

2. Lựa chọn nào sau đây KHÔNG là ưu điểm của việc sử dụng thiết bị số trong lưu trữ dữ liệu?

A. Bị giới hạn về dung lượng.

B. Có thể truy cập mọi lúc mọi nơi.

C. Bảo vệ dữ liệu an toàn, bảo mật cao.

D. Dễ dàng tổ chức, sao lưu dữ liệu.

3. Cô Vân cần thông tin về mức độ sử dụng mạng xã hội của học sinh các lớp do cô phụ trách giảng dạy. Theo em, cô Vân cần phân tích những dữ liệu gì và nhận được các dữ liệu đó ở dạng gì (văn bản, âm thanh, hình ảnh)?

VẬN DỤNG

1. Hiện nay, một số đơn vị giáo dục tại Việt Nam đã trang bị các máy tính hiệu năng cao (máy tính có khả năng thực hiện hàng triệu tỉ phép tính/giây) để phục vụ cho công tác nghiên cứu khoa học và xây dựng các ứng dụng thực tiễn. Em hãy tìm hiểu và giới thiệu về một máy tính hiệu năng cao đã được trang bị tại Việt Nam.

2. Em hãy tìm hiểu về những đóng góp quan trọng của các thiết bị số trong việc ghi hình và truyền hình trực tuyến.

Chân trời sáng tạo

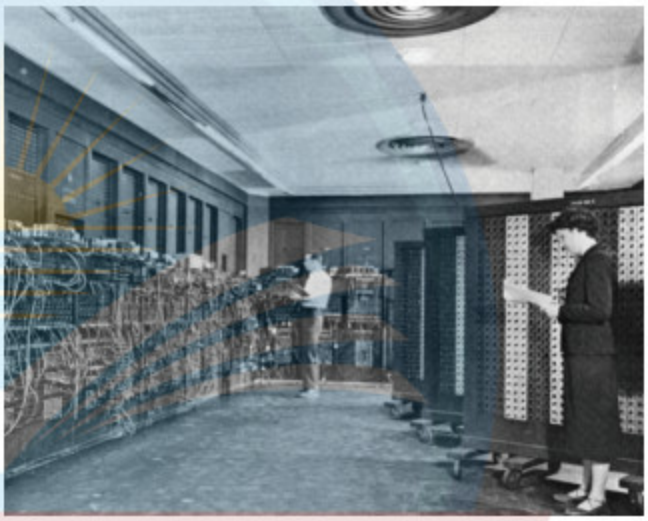
BÀI A2 TIN HỌC VÀ XÃ HỘI

MỤC TIÊU **Sau bài học này, em sẽ:**

- Giới thiệu được các thành tựu nổi bật ở một số mốc thời gian để minh hoạ sự phát triển của ngành Tin học.
- Trình bày được những đóng góp cơ bản của Tin học đối với xã hội, nêu được ví dụ minh hoạ.

KHỞI ĐỘNG

Vào thập niên 1950, những chiếc máy tính đầu tiên (*Hình 1*) chiếm diện tích bằng cả một căn phòng, nhưng khả năng xử lí thông tin bị hạn chế. Đến nay, Tin học đã có những bước tiến vượt bậc khi có các thiết bị nhỏ gọn như máy tính xách tay, điện thoại thông minh,... có thể xử lí và thực hiện hàng triệu phép tính mỗi giây. Em hãy thảo luận với bạn và nêu một số thành tựu trong lịch sử phát triển của Tin học có ảnh hưởng đến việc học tập và giao tiếp hằng ngày mà em biết.



Hình 1. ENIAC có thiết kế to lớn chiếm cả một căn phòng[1]

Chân trời sáng tạo

KHÁM PHÁ

1. Quá trình phát triển của ngành Tin học

a) Giai đoạn 1940 - 1980: Thời kì bắt đầu của máy tính

Từ nhiều thế kỉ trước, con người đã biết cách sử dụng các công cụ cơ học như bàn tính và máy tính cơ học để tính toán. Tuy nhiên phải đến thập kỉ 1940, máy tính điện tử mới bắt đầu xuất hiện. Dấu mốc đầu tiên đáng nhớ cho sự kiện này là vào năm 1936, nhà toán học người Anh Alan Turing đã đề xuất ý tưởng về một cỗ máy vạn năng, được cộng đồng khoa học gọi là máy Turing. Ý tưởng của Turing đã hình thành nền tảng cho máy tính hiện đại mà chúng ta sử dụng ngày nay.

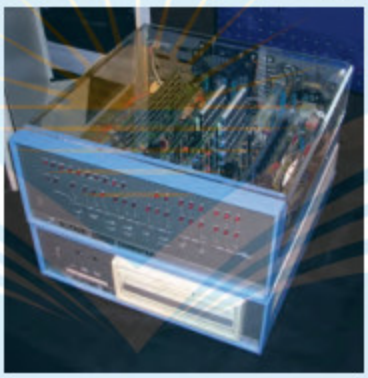
[1] Nguồn ảnh: U.S.Army Photo

Năm 1943 – 1944, hai giáo sư đến từ đại học Pennsylvania là John Mauchly và J. Presper Eckert, đã hợp tác tạo ra thiết bị tích phân và tính toán số học điện tử hay còn gọi là ENIAC (Electronic Numerical Integrator and Calculator – *Hình 1*). ENIAC được coi là máy tính điện tử đầu tiên trên thế giới và được sử dụng trong các dự án quân sự, khoa học, đánh dấu sự bắt đầu của kỉ nguyên máy tính điện tử.

Hình 2. Máy tính DEC PDP–8[1]

Năm 1965, công ty Thiết bị kĩ thuật số (DEC) đã thương mại hoá thành công sản phẩm máy tính với tên gọi là DEC PDP–8 (*Hình 2*). Máy tính này có kích thước nhỏ gọn nhưng có thể thực hiện mọi công việc của các máy tính có kích thước to lớn trước đó.

Từ năm 1974 đến 1977, bắt đầu có nhiều máy tính cá nhân xuất hiện trên thị trường. Bao gồm IBM 5100, TRS–80 và Scelbi & Mark–8 Altair. Trong đó, Altair 8080 (*Hình 3*) được mô tả là bộ máy tính mini đầu tiên trên thế giới.



Hình 3. Máy tính Altair 8080[2]

Hình 4. Máy tính IBM Model 5150[3]

Năm 1981, IBM giới thiệu máy tính cá nhân đầu tiên của họ, IBM Model 5150 (*Hình 4*). Máy sử dụng hệ điều hành PC-DOS, đồng thời được trang bị chip CPU Intel 8088 với hai ổ đĩa mềm và màn hình màu tuỳ chọn. Sự ra đời của hệ điều hành PC-DOS đánh dấu bước ngoặt quan trọng, giúp đơn giản hoá việc quản lí và sử dụng máy tính. Đặc biệt, vào năm 1985, sự xuất hiện của hệ điều hành Windows với giao diện đồ hoạ đã mở ra kỉ nguyên mới, cho phép người dùng tương tác với máy tính một cách dễ dàng thông qua các biểu tượng và cửa sổ thay vì phải nhập lệnh thủ công. Cùng với đó, sự phát triển của các ngôn ngữ lập trình bậc cao như BASIC và C đã giúp lập trình viên tạo ra các ứng dụng phức tạp hơn một cách dễ dàng. Khi phần cứng máy tính ngày càng mạnh mẽ và giá thành rẻ hơn, kết hợp với hệ điều hành thân thiện và ngôn ngữ lập trình dễ tiếp cận, máy tính đã dần trở nên phổ biến với tất cả mọi người.

[1] Nguồn ảnh: Alkivar.

[2] Nguồn ảnh: Michael Holley.

[3] Nguồn ảnh: Hidalgo944.

b) Giai đoạn 1980 - 2000: Thời kì của mạng Internet toàn cầu

Tiền thân của mạng Internet ngày nay là mạng ARPANET, do cơ quan quản lí dự án nghiên cứu phát triển ARPA thuộc bộ quốc phòng Hoa Kỳ thiết lập để liên kết bốn địa điểm đầu tiên vào tháng 7 năm 1969 bao gồm: Viện nghiên cứu Stanford, Đại học California (Los Angeles), Đại học Utah và Đại học California (Santa Barbara). Đó chính là mạng liên khu vực (Wide Area Network – WAN) đầu tiên được xây dựng.

Năm 1983, ARPANET chính thức chuyển sang sử dụng giao thức TCP/IP. Giao thức TCP/IP ngày càng thể hiện rõ các điểm mạnh, quan trọng nhất là khả năng liên kết với các mạng khác một cách dễ dàng. Chính điều này cùng với các chính sách mở đã cho phép các mạng khác kết nối được với mạng ARPANET. Từ đó, ARPANET được đánh giá là mạng trụ cột của Internet.

Năm 1991, Tim Berners Lee ở Trung tâm nghiên cứu nguyên tử châu Âu (CERN) phát minh ra World Wide Web (WWW) dựa theo một ý tưởng về siêu văn bản (hypertext) được Ted Nelson đưa ra từ năm 1985. Có thể nói đây là một cuộc cách mạng trên Internet vì người dùng có thể truy cập, trao đổi thông tin một cách dễ dàng, nhanh chóng. Nhờ đó, Internet đã phát triển một cách nhanh chóng và trở thành một mạng lớn nhất trên thế giới, xuất hiện trong mọi lĩnh vực thương mại, chính trị, quân sự, nghiên cứu, giáo dục, văn hoá, xã hội,…

c) Giai đoạn 2000 - đến nay: Thời kì của thiết bị di động, mạng xã hội và trí tuệ nhân tạo

Ngày 03/04/1973, tiến sĩ Martin Cooper của Motorola đã thực hiện cuộc gọi đầu tiên từ chiếc điện thoại di động của mình trong khi đi dạo trên đại lộ Sixth Anvenue của thành phố New York (Hoa Kỳ). Vào thời điểm đó, không một ai có thể nhận ra cái khối thô kệch, có hình dáng giống như một viên gạch với trọng lượng gần một kilôgam trong tay Martin là tiền thân của những điện thoại di động ngày nay.

Năm 2007, hãng Apple đã cho ra mắt điện thoại Iphone, đánh dấu sự sáng tạo đột phá về kiểu dáng với nét đặc trưng là màn hình cảm ứng siêu nhạy giúp người sử dụng có thể thao tác dễ dàng chỉ bằng việc "lướt" nhẹ các ngón tay. Iphone đã tạo nên một cơn sốt chưa từng có và chính thức khởi đầu cho cuộc cạnh tranh khốc liệt trong phân khúc điện thoại thông minh.

Trải qua hơn 50 năm phát triển, điện thoại di động ngày càng trở nên phổ biến và không thể thiếu được trong cuộc sống con người (*Hình 5*). Chúng không ngừng được thay đổi, cải tiến trong công nghệ cũng như kiểu dáng với các thương hiệu sản xuất di động hàng đầu như: Nokia, Blackberry, Apple, Samsung, LG, Sony Ericsson, Motorola,…

Hình 5. Trải qua hơn 50 năm phát triển, điện thoại di động ngày càng trở nên phổ biến và không thể thiếu được trong cuộc sống con người

Cùng với sự phát triển của điện thoại di động và Internet, các mạng xã hội đã nhanh chóng hình thành và phát triển. Ví dụ, cách đây khoảng 20 năm, tỉ lệ sử dụng mạng xã hội ở Hoa Kỳ chỉ ở mức thấp. Thế nhưng từ năm 2005 đến nay, các mạng xã hội như: Facebook, Instagram,… đã mở ra một kỉ nguyên hoàn toàn mới cho người sử dụng Internet.

Thuật ngữ "Trí tuệ nhân tạo" được Giáo sư John McCarthy (Đại học Stanford) đưa ra với định nghĩa là "khoa học và kĩ thuật chế tạo máy móc thông minh". Hội nghị Dartmouth năm 1956 được đánh dấu như sự ra đời của ngành Trí tuệ nhân tạo. Ở giai đoạn đầu, Trí tuệ nhân tạo tập trung vào việc tạo ra các thuật toán giải quyết các vấn đề toán học và trò chơi đơn giản. Đến đầu thế kỉ 21, sự bùng nổ của dữ liệu lớn, cùng với sự tiến bộ vượt bậc về năng lực xử lí của máy tính cũng như các phương pháp học máy, học sâu, đã mở ra kỉ nguyên mới của Trí tuệ nhân tạo. Ngày nay, Trí tuệ nhân tạo đã trở thành yếu tố thiết yếu trong nhiều lĩnh vực như giao thông, y tế, tài chính, thương mại điện tử,...

Chọn câu trả lời đúng cho câu hỏi dưới đây.

Ý tưởng của ai về máy tính số đã giúp hình thành nền tảng cho máy tính hiện đại mà chúng ta sử dụng ngày nay?

A. Steve Jobs. B. Bill Gates.

C. Alan Turing. D. Tim Berners Lee.

Những điểm đáng nhớ về sự phát triển của Tin học:

- **Thời kì bắt đầu:** Ý tưởng về cỗ máy vạn năng của Alan Turing năm 1936 đã hình thành nền tảng cho các máy tính ngày nay. Sự phát triển của máy tính bán dẫn đã giúp máy tính trở nên nhỏ gọn và tiếp cận rộng rãi hơn với tất cả mọi người.
- **Mạng Internet:** Phát minh World Wide Web của Tim Berners Lee năm 1991 là một cuộc cách mạng, vì qua đó mọi người có thể truy cập và trao đổi thông tin một cách dễ dàng, nhanh chóng. Đưa Internet trở thành một mạng lớn nhất trên thế giới.
- **Thời kì của điện thoại di động, mạng xã hội và trí tuệ nhân tạo:** Sự xuất hiện của điện thoại di động và mạng xã hội đã cách mạng hoá cách chúng ta giao tiếp, làm việc và kết nối toàn cầu. Sự bùng nổ của dữ liệu lớn, cùng với sự tiến bộ vượt bậc về năng lực xử lí của máy tính cũng như học máy, học sâu, đã mở ra kỉ nguyên mới để trí tuệ nhân tạo ngày càng đóng vai trò thiết yếu trong nhiều ngành, lĩnh vực.

2. Đóng góp của Tin học đối với xã hội

Sự phát triển của Tin học mang lại nhiều đóng góp to lớn cho sự phát triển của xã hội. Một số lĩnh vực và cách mà Tin học đã tạo ra sự ảnh hưởng tích cực đối với cuộc sống và xã hội có thể kể đến như sau.

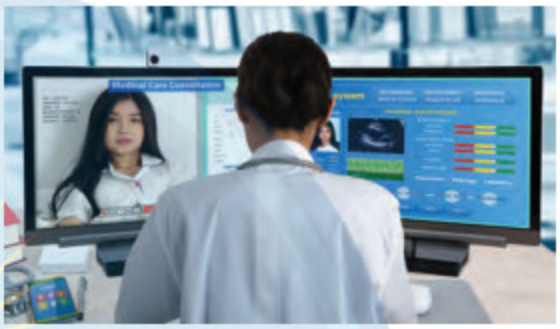
a) Giao thông và vận tải

Có rất nhiều ứng dụng của Tin học cho lĩnh vực giao thông vận tải được phát triển trong nhiều năm qua, giúp cho việc đi lại nhanh chóng và tiện lợi hơn so với trước đây. Một vài thành tựu có thể kể đến như: Hệ thống định vị GPS và các ứng dụng tìm đường; hệ thống thu phí không dừng; camera 360 trên xe ô tô; camera hành trình;... giúp cải thiện việc đi lại và giảm thiểu thời gian di chuyển. Bên cạnh các thành tựu trên, công nghệ xe tự hành đang ngày càng phát triển, hứa hẹn mang lại sự an toàn và hiệu quả hơn trong giao thông đường bộ (*Hình 6*).

Hình 6. Xe tự hành

b) Y tế

Ở lĩnh vực y tế, hồ sơ y tế điện tử giúp cải thiện việc quản lí thông tin bệnh án, đồng thời tăng cường chia sẻ dữ liệu. Tư vấn sức khoẻ từ xa và các ứng dụng sức khoẻ giúp người dân tự theo dõi và quản lí sức khoẻ cá nhân một cách hiệu quả (*Hình 7*).



Hình 7. Bác sĩ tư vấn bệnh nhân từ xa

c) Giáo dục

Tin học ngày càng phát triển đã mở ra cánh cửa cho giáo dục phát triển với ngày càng nhiều công nghệ được ứng dụng vào việc giảng dạy và học tập. Có thể kể đến như: các bài giảng điện tử, bảng tương tác, các hệ thống mô phỏng (*Hình 8*), kho học liệu mở, các khoá học trực tuyến,... Điều đó giúp cho việc học tập của học sinh trở nên thuận lợi hơn bao giờ hết. Các em học sinh ở vùng sâu, vùng xa cũng có thể dễ dàng truy cập được sách, bài giảng và nhiều tài liệu bổ ích từ khắp nơi trên thế giới.

Hình 8. Ứng dụng công nghệ mô phỏng trong giáo dục

d) Kinh doanh và thương mại

Trong lĩnh vực kinh doanh và thương mại, Tin học cung cấp sự thuận tiện, tin cậy, dễ dàng tương tác trong giao dịch tài chính và thương mại điện tử. Trí tuệ nhân tạo và phân tích dữ liệu giúp dự đoán xu hướng thị trường và tối ưu hoá chiến lược kinh doanh,...

e) Truyền thông và mạng xã hội

Các nền tảng truyền thông xã hội như Facebook, Instagram tạo ra cơ hội giao tiếp và chia sẻ thông tin trên toàn cầu. Các công nghệ phát trực tiếp giúp mọi người tiếp cận thông tin, nội dung giải trí một cách dễ dàng và linh hoạt.

g) Nông nghiệp và bảo vệ môi trường

Trong nông nghiệp, Tin học hỗ trợ người nông dân thông qua ứng dụng nông nghiệp thông minh và chính xác; thiết bị cảm biến thông minh thu thập dữ liệu về đất đai, thời tiết và chất dinh dưỡng để cung cấp thông tin cho quản lí nông trại hiệu quả hơn; các hệ thống trí tuệ nhân tạo phân tích dữ liệu để dự đoán mùa màng và tình trạng cây trồng (*Hình 9*).

Hình 9. Ứng dụng Tin học trong nông nghiệp

Trong quản lí tài nguyên nước thông minh, các hệ thống giám sát từ xa được sử dụng để theo dõi lượng nước tiêu thụ, đánh giá tình trạng nguồn nước và dự đoán các hiện tượng thiên tai như hạn hán, lũ lụt để có biện pháp ứng phó. Từ đó tối ưu hoá việc sử dụng nước trong nông nghiệp, sinh hoạt,…

Đối với quản lí rừng thông minh, các hệ thống giám sát từ xa giúp theo dõi tình trạng của rừng, phát hiện sớm các dấu hiệu nguy cơ cháy rừng, đồng thời giúp hỗ trợ quản lí và phát triển sinh thái rừng,…

Chọn câu trả lời đúng cho các câu hỏi dưới đây.

1. Giải pháp Tin học nào trong giao thông và vận tải đã góp phần giảm thời gian di chuyển và giúp việc đi lại trở nên thuận tiện hơn?

A. Hệ thống định vị GPS.

B. Tư vấn sức khoẻ từ xa.

C. Kho học liệu mở.

D. Công nghệ Streaming video.

2. Hồ sơ y tế điện tử giúp cải thiện điều gì trong hệ thống y tế?

A. Quản lí thông tin bệnh án và tăng cường chia sẻ dữ liệu.

B. Cải thiện giao dịch tài chính.

C. Giúp người dân tự theo dõi và quản lí sức khoẻ cá nhân.

D. Tăng cường khả năng mô phỏng trong giáo dục.

Tin học đã tạo ra sự ảnh hưởng tích cực đối với cuộc sống và xã hội trong các lĩnh vực như: giao thông và vận tải; y tế; giáo dục; kinh doanh và thương mại; truyền thông và mạng xã hội; nông nghiệp và bảo vệ môi trường;...

Chân trời sáng tạo

LUYỆN TẬP

Chọn câu trả lời đúng cho các câu hỏi dưới đây.

1. Tư vấn sức khoẻ từ xa và các ứng dụng sức khoẻ giúp người dân làm gì?

A. Cải thiện giao dịch tài chính.

B. Tự theo dõi và quản lí sức khoẻ cá nhân một cách hiệu quả.

C. Tăng cường khả năng tự mô phỏng trong giáo dục.

D. Giảm thời gian di chuyển.

2. Những giải pháp Tin học nào được áp dụng trong giáo dục để hỗ trợ việc giảng dạy và học tập?

A. Bảng tương tác, hệ thống mô phỏng, kho học liệu mở.

B. Hệ thống định vị GPS.

C. Các công cụ streaming video và podcasting.

D. Hồ sơ y tế điện tử.

3. Công nghệ nào được sử dụng để dự đoán xu hướng thị trường và tối ưu hoá chiến lược kinh doanh?

A. Trí tuệ nhân tạo và phân tích dữ liệu.

B. Hệ thống thanh toán trực tuyến.

C. Máy quét mã vạch.

D. Mang lại sự an toàn và hiệu quả hơn trong giao thông đường bộ.

4. Trong Tin học đã có ứng dụng nào góp phần vào việc cải thiện tránh va chạm trong giao thông vận tải?

A. Tư vấn sức khoẻ từ xa.

B. Camera 360 trên xe ô tô.

C. Hồ sơ y tế điện tử.

D. Tất cả các phương án trên.

Chân trời sáng tạo

VẬN DỤNG

Em hãy thảo luận với bạn và thực hiện bài báo cáo về ứng dụng công nghệ thông tin trong một lĩnh vực mà em quan tâm (Giao thông vận tải, Y tế, Giáo dục, Kinh doanh và thương mại, Truyền thông và mạng xã hội, Nông nghiệp và bảo vệ môi trường,...). Về hình thức, báo cáo từ 10 – 15 trang, các phiếu đăng kí, bảng phân công công việc và bảng tổng kết, đánh giá hoạt động của nhóm trên tệp .pdf.

BÀI A3 VAI TRÒ CỦA THIẾT BỊ THÔNG MINH TRONG XÃ HỘI VÀ CUỘC CÁCH MẠNG CÔNG NGHIỆP 4.0

MỤC TIÊU **Sau bài học này, em sẽ:**

- Nêu được ví dụ cụ thể về thiết bị thông minh. Giải thích được vai trò của thiết bị thông minh đối với sự phát triển của xã hội và cuộc cách mạng công nghiệp 4.0.

KHỞI ĐỘNG

Theo em, các thiết bị thông minh đã và đang thay đổi cách chúng ta sống, làm việc và tương tác trong bối cảnh của cuộc cách mạng công nghiệp 4.0 như thế nào?

KHÁM PHÁ

1. Thiết bị thông minh

Thiết bị thông minh là các thiết bị điện tử có khả năng kết nối với mạng, xử lí dữ liệu và thực hiện các chức năng tự động hoặc điều khiển từ xa. Các thiết bị thường tích hợp cảm biến, phần mềm và kết nối Internet, cho phép người dùng điều khiển hoặc theo dõi thiết bị qua các ứng dụng trên điện thoại hoặc máy tính.

Hình 1. Điện thoại thông minh

Điện thoại thông minh: Điện thoại thông minh không chỉ là công cụ liên lạc, mà còn là một trung tâm thông tin cá nhân có khả năng xử lí thông tin. Chúng có nhiều tính năng mạnh mẽ như: truy cập Internet, định vị GPS, chụp ảnh, quay phim, quản lí email và lịch trình cá nhân,... Ngoài ra, điện thoại thông minh còn cho phép người dùng trải nghiệm sử dụng các phần mềm giải trí, mạng xã hội,… (*Hình 1*).

Hình 2. Máy tính bảng

Máy tính bảng: Máy tính bảng là một công cụ có các chức năng kết hợp giữa điện thoại thông minh và máy tính cá nhân, cho phép người dùng truy cập Internet, xem và chỉnh sửa tài liệu và thậm chí làm việc từ xa một cách tiện lợi (*Hình 2*).

Đồng hồ thông minh: Đồng hồ thông minh không chỉ để xem giờ mà còn cung cấp nhiều tính năng như nhận thông báo từ điện thoại, theo dõi sức khoẻ, hoạt động thể chất và thậm chí làm việc như một trợ lí cá nhân (*Hình 3*).

Hình 3. Đồng hồ thông minh không chỉ để xem giờ mà còn cung cấp nhiều tính năng khác

Loa thông minh: Loa thông minh được trang bị trợ lí ảo như Amazon Alexa, Google Assistant hoặc Apple Siri giúp người dùng tương tác với các dịch vụ trực tuyến, điều khiển các thiết bị trong nhà thông minh và trả lời câu hỏi một cách nhanh chóng (*Hình 4*).

Hình 4. Loa thông minh được trang bị trợ lí ảo

1. Theo em, trong cuộc sống hằng ngày, máy tính bảng thường được sử dụng cho các công việc gì?

2. Hãy liệt kê một vài tính năng trên đồng hồ thông minh mà em biết.

Một số thiết bị thông minh thông dụng: điện thoại thông minh, máy tính bảng, đồng hồ thông minh, loa thông minh,…

2. Vai trò của thiết bị thông minh trong cách mạng công nghiệp 4.0

Các cuộc cách mạng công nghiệp là những giai đoạn phát triển vượt bậc trong lịch sử nhân loại, đặc trưng bởi những bước tiến đáng kể trong công nghệ và cách mạng hoá sản xuất. Dưới đây là tóm tắt các cuộc cách mạng công nghiệp:

Cách mạng công nghiệp lần thứ nhất: Bắt đầu vào khoảng năm 1784 với điểm nổi bật là việc sử dụng năng lượng nước, hơi nước và cơ giới hoá sản xuất. Cuộc cách mạng công nghiệp này bắt đầu từ việc James Watt phát minh ra động cơ hơi nước vào năm 1784, phát minh vĩ đại này đã châm ngòi cho sự bùng nổ của công nghiệp thế kỉ 19 lan rộng từ Anh đến khắp Châu Âu và Hoa Kỳ.

Cách mạng công nghiệp lần thứ hai: Bắt đầu vào khoảng năm 1870, đặc trưng của cuộc cách mạng công nghiệp này là việc sử dụng năng lượng điện và sự ra đời của các dây chuyền sản xuất hàng loạt trên quy mô lớn. Cách mạng công nghiệp lần thứ hai đã tạo nên tiền đề cơ sở vững chắc để phát triển nền công nghiệp ở mức cao hơn nữa.

Cách mạng công nghiệp lần thứ ba: Xuất hiện vào khoảng năm 1969 với sự ra đời và lan toả của công nghệ thông tin, máy tính hỗ trợ con người trong hoạt động trí tuệ. Cuộc cách mạng này còn được gọi là cuộc cách mạng số vì nó được xúc tác nhờ sự phát triển của chất bán dẫn, siêu máy tính, máy tính cá nhân và Internet.

Cách mạng công nghiệp lần thứ tư (cách mạng công nghiệp 4.0): Đây là cuộc cách mạng công nghiệp đang diễn ra dựa trên nền tảng công nghệ số và tích hợp với các công nghệ thông minh để tạo ra quy trình và phương thức sản xuất mới. Việc áp dụng các thành tựu của công nghệ thông tin như trí tuệ nhân tạo, Internet của vạn vật, điện toán đám mây, dữ liệu lớn cùng các tiến bộ mang tính đột phá của công nghệ sinh học, công nghệ vật liệu, tự động hoá và các công nghệ khác vào hệ thống sản xuất và sản phẩm đã làm mờ ranh giới của các hệ thống thực (thế giới vật lí, thế giới các thực thể) và các hệ thống ảo (thế giới số, không gian số).

Điều đó tạo ra một nền sản xuất thông minh mà ở đó sản phẩm được sản xuất trong thế giới vật lí nhưng quá trình tính toán, thiết kế, tạo mẫu,… được thực hiện trên không gian số. Trong bối cảnh cách mạng công nghiệp 4.0, các thiết bị thông minh đóng vai trò ngày càng quan trọng, làm thay đổi cách vận hành của nhiều ngành nghề và dịch vụ. Cụ thể là:

Tạo ra một nền sản xuất thông minh: Các nhà máy và dây chuyền sản xuất hiện nay đang tích hợp các thiết bị thông minh như cảm biến, robot và các hệ thống điều khiển tự động. Những thiết bị này giúp tự động hoá và tối ưu hoá quá trình sản xuất, giảm thiểu lãng phí và tăng cường hiệu suất. Chẳng hạn, cảm biến thông minh được lắp đặt trên máy móc giúp giám sát và thu thập dữ liệu về trạng thái của thiết bị, từ đó giúp phát hiện sớm các sự cố và ngăn ngừa hư hỏng. Hệ thống tự động hoá trong sản xuất cũng giúp giảm bớt sự phụ thuộc vào lao động thủ công và tăng cường sự ổn định và chính xác của quy trình sản xuất.

Tăng cường vai trò của tri thức – yếu tố trí tuệ: Trong cách mạng công nghiệp 4.0, tri thức - yếu tố trí tuệ của con người ngày càng chiếm tỉ trọng lớn hơn trong cơ cấu giá trị của sản phẩm và dịch vụ, tỉ trọng này đặc biệt được tăng cường khi có sự kết hợp với các thiết bị thông minh. Ví dụ, điện thoại thông minh ngày nay không chỉ là công cụ liên lạc mà còn là một kho tri thức, trí tuệ với hàng loạt ứng dụng thông minh hỗ trợ học tập, làm việc và giải trí.

Thiết bị thông minh đóng vai trò chủ chốt trong các hệ thống IoT – một nội dung cơ bản của cuộc cách mạng công nghiệp 4.0: Thiết bị thông minh kết hợp với việc kết nối và trao đổi thông tin sẽ tạo ra một hệ thống các thiết bị thông minh có khả năng tự động hoá và phối hợp để cung cấp những dịch vụ tiện ích cho con người. Ví dụ, trong hệ thống nhà thông minh, các thiết bị như đèn, điều hòa, camera giám sát,… đều được kết nối với nhau và có thể điều khiển từ xa qua ứng dụng trên điện thoại. Điều này không chỉ mang lại sự tiện nghi mà còn giúp tiết kiệm năng lượng và tăng cường an ninh cho người sử dụng.

Vai trò của các thiết bị thông minh đối với cuộc cách mạng công nghiệp 4.0: Tạo ra một nền sản xuất thông minh; Tăng cường vai trò của tri thức và yếu tố trí tuệ trong cơ cấu giá trị của sản phẩm và dịch vụ; Đóng vai trò chủ chốt trong các hệ thống IoT.

3. Những đóng góp của thiết bị thông minh đối với sự phát triển của xã hội

Hiện nay, các thiết bị thông minh đã trở thành những người bạn đồng hành không thể thiếu trong mọi lĩnh vực của sự phát triển của xã hội. Dưới đây là các đóng góp quan trọng:

a) Tự động hoá và tăng hiệu suất công việc

Máy tính và hệ thống tự động: Máy tính điều khiển các hệ thống tự động hoá trong sản xuất và quản lí doanh nghiệp, giúp tăng hiệu suất, giảm sai sót và tiết kiệm chi phí. Chẳng hạn, các dây chuyền sản xuất tự động và hệ thống quản lí tài nguyên doanh nghiệp (ERP) được điều khiển và giám sát qua máy tính.

Ứng dụng quản lí công việc: Các ứng dụng quản lí dự án và lịch làm việc trên điện thoại thông minh, máy tính bảng,... giúp tổ chức công việc hiệu quả hơn, đảm bảo sự linh hoạt và tối ưu hoá thời gian làm việc.

b) Internet vạn vật (IoT)

Các hệ thống Internet vạn vật có thể liên kết các thiết bị và cảm biến thông qua Internet để thu thập và chia sẻ dữ liệu từ môi trường xung quanh. Ví dụ, trong giao thông thông minh, các cảm biến trên đường phố có thể kết nối với các phương tiện và đèn giao thông để điều chỉnh luồng xe, giảm ùn tắc và tối ưu hoá thời gian di chuyển. Dữ liệu thu thập được cũng có thể giúp người lái xe tránh các khu vực đông đúc hoặc tai nạn, đồng thời hỗ trợ quản lí giao thông hiệu quả hơn (*Hình 5*).

Hình 5. Hệ thống giao thông thông minh

c) Giao tiếp và kết nối

Mạng xã hội và truyền thông: Điện thoại di động và máy tính giúp mọi người kết nối qua mạng xã hội, email và các ứng dụng trò chuyện, tạo điều kiện chia sẻ thông tin và hợp tác dễ dàng hơn.

Họp trực tuyến và làm việc từ xa: Công cụ họp trực tuyến như Zoom, Microsoft Teams,... trên máy tính và điện thoại giúp duy trì hiệu quả làm việc từ xa.

d) Giáo dục và đào tạo

Học trực tuyến: Máy tính và điện thoại thông minh hỗ trợ học tập từ xa qua các nền tảng như Coursera, Khan Academy, giúp học sinh và sinh viên tiếp cận kiến thức mọi lúc, mọi nơi.

Ứng dụng học tập: Các ứng dụng học tập trên điện thoại cung cấp tài liệu học tập bổ ích, bài kiểm tra và hướng dẫn, giúp học sinh và sinh viên nâng cao trình độ học vấn.

e) Chăm sóc sức khoẻ

Thiết bị đeo thông minh: Đồng hồ thông minh và các thiết bị đeo tay theo dõi các chỉ số sức khoẻ như nhịp tim, giấc ngủ và mức độ hoạt động, cung cấp dữ liệu để cải thiện sức khoẻ cá nhân (*Hình 6*).

Tư vấn y tế từ xa: Điện thoại và máy tính cho phép thực hiện các buổi tư vấn y tế từ xa, giúp bệnh nhân nhận được sự chăm sóc y tế mà không cần phải đến bệnh viện.

Hình 6. Hệ thống chăm sóc sức khoẻ sử dụng các thiết bị đeo thông minh

g) Nông nghiệp thông minh

Quản lí từ xa: Máy tính và điện thoại thông minh cho phép quản lí trang trại từ xa, sử dụng cảm biến để theo dõi tình trạng của đất, thời tiết và cây trồng, giúp nông dân đưa ra các quyết định kịp thời và chính xác (*Hình 7*).

Drone và cảm biến: Máy tính điều khiển thiết bị bay không người lái (drone) và cảm biến trong nông nghiệp, giúp giám sát cây trồng, phun thuốc và thu hoạch, tăng hiệu suất và giảm lao động thủ công.

Hình 7. Hệ thống nhà vườn thông minh

h) Thành phố thông minh

Quản lí giao thông và năng lượng: Các hệ thống quản lí giao thông và năng lượng trong thành phố thông minh được điều khiển, giám sát qua máy tính và điện thoại, giúp giảm ùn tắc, tiết kiệm năng lượng và nâng cao chất lượng cuộc sống.

Giám sát an ninh: Hệ thống camera giám sát và các thiết bị an ninh thông minh giúp đảm bảo an toàn cho cư dân thành phố thông qua việc theo dõi và cảnh báo kịp thời.

Một số đóng góp chính của các thiết bị thông minh đối với sự phát triển của xã hội: Tự động hoá và tăng hiệu suất công việc; Internet vạn vật; Giao tiếp và kết nối; Giáo dục và đào tạo; Chăm sóc sức khoẻ; Nông nghiệp thông minh; Thành phố thông minh;...

Chân trời sáng tạo

LUYỆN TẬP

1. Kể tên một vài ứng dụng trên điện thoại hoặc máy tính bảng mà em biết.
2. Hãy kể tên một số ứng dụng hay công cụ hỗ trợ việc học trực tuyến mà em biết.

VẬN DỤNG

1. Ngoài các thiết bị thông minh kể trên, theo em còn biết những thiết bị thông minh nào khác, hãy kể tên và nêu một số chức năng của chúng.
2. Ngoài các đóng góp trên đối với xã hội, em hãy tìm hiểu và cho biết các đóng góp nào khác của thiết bị thông minh trong cuộc cách mạng công nghiệp 4.0?

BÀI A4 THỰC HÀNH SỬ DỤNG MỘT SỐ THIẾT BỊ SỐ THÔNG DỤNG

MỤC TIÊU **Sau bài học này, em sẽ:**

- Nhận biết được các thiết bị số thông dụng khác ngoài máy tính để bàn, laptop. Giải thích được các thiết bị đó cũng là những hệ thống xử lí thông tin.
- Biết cách sử dụng các thiết bị số thông dụng để phục vụ học tập, sinh hoạt và quản lí công việc hàng ngày.

KHỞI ĐỘNG

Ngoài các chức năng cơ bản là nghe, gọi và nhắn tin, em hãy nêu những chức năng khác trên điện thoại thông minh.

THỰC HÀNH

Nhiệm vụ 1. Nhận biết một số thiết bị số thông dụng

Thiết bị số là thiết bị điện tử, máy tính, viễn thông, truyền dẫn, thu phát sóng vô tuyến điện và thiết bị tích hợp khác được sử dụng để sản xuất, truyền đưa, thu thập, xử lí, lưu trữ và trao đổi thông tin số. Tương tự như một máy tính, các thiết bị số cũng được gắn bộ xử lí thông tin với đầy đủ các chức năng như tiếp nhận thông tin vào, lưu trữ, xử lí thông tin, đưa thông tin ra. Một số thiết bị số thông dụng ngoài máy tính có thể kể đến như: điện thoại thông minh, máy tính bảng và đồng hồ thông minh,...

Yêu cầu: Quan sát *Hình 1* và trả lời cho các câu sau:

a) Nêu tên của các thiết bị, vật dụng trong *Hình 1*.

b) Cho biết các thiết bị nào là thiết bị số.

Chân trời sáng tạo

Hình 1a.

Hình 1b.

Hình 1c.

Hình 1d.

Hình 1e.

Hình 1g.

Hình 1h.

Hình 1i.

Hình 1. Một số thiết bị

Nhiệm vụ 2. Chia sẻ tệp, thư mục trên thiết bị thông minh

Yêu cầu: Em hãy sử dụng thiết bị thông minh để gửi một tệp hoặc một thư mục bất kì qua email.

Hướng dẫn: Thực hiện các bước như ở *Hình 2* để chia sẻ tệp, thư mục trên thiết bị thông minh sử dụng hệ điều hành iOS.

1 Truy cập vào ứng dụng **Tệp** trên thiết bị sử dụng hệ điều hành iOS. Cửa sổ hiện ra như *Hình 2b*.

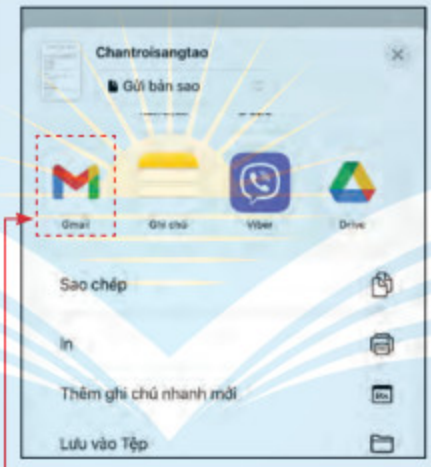
Hình 2a.

Hình 2b.

2 Chọn **Trên iPhone** để truy cập dữ liệu được lưu trữ trên thiết bị thông minh. Cửa sổ hiện ra như *Hình 2c*.

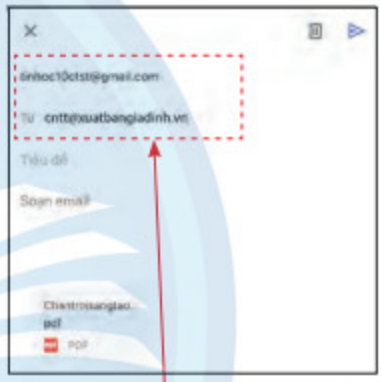
3 Truy cập đến thư mục có chứa dữ liệu cần chia sẻ trên thiết bị thông minh, nhấn và giữ vào dữ liệu cần chia sẻ cho đến khi xuất hiện bảng chọn, chọn Chia sẻ. Cửa sổ hiện ra như *Hình 2d*.

Hình 2c.



4 Chọn nơi cần chia sẻ (chẳng hạn, chọn Gmail). Cửa sổ hiện ra như *Hình 2e*.

Hình 2d.



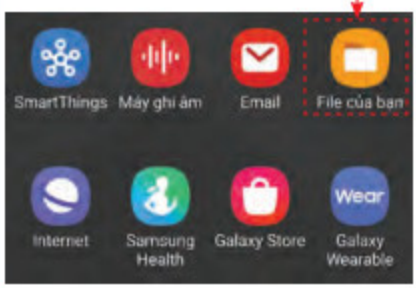
5 Nhập địa chỉ mail và gửi.

Hình 2e.

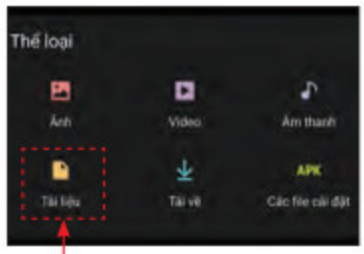
Hình 2. Thao tác chia sẻ tệp, thư mục trên thiết bị thông minh sử dụng hệ điều hành iOS

Thực hiện các bước như ở *Hình 3* để chia sẻ tệp, thư mục trên thiết bị thông minh sử dụng hệ điều hành Android.

1 Truy cập vào ứng dụng **File của bạn** trên thiết bị sử dụng hệ điều hành Android. Cửa sổ hiện ra như *Hình 3b*.



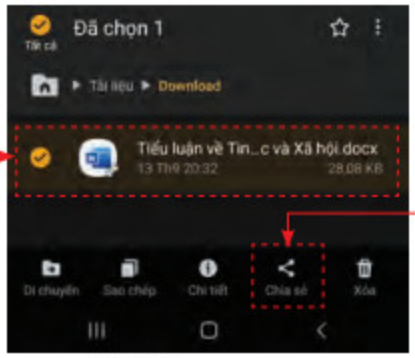
Hình 3a.



Hình 3b.

2 Chọn **Tài liệu** để truy cập tài liệu trên thiết bị thông minh. Cửa sổ hiện ra như *Hình 3c*.

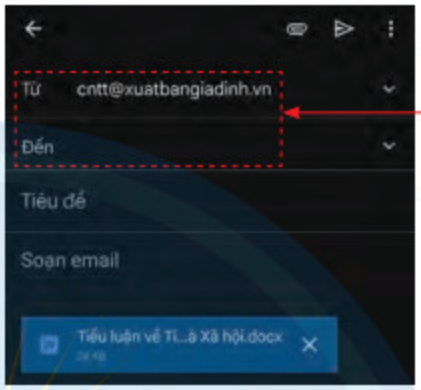
3 Truy cập đến thư mục có chứa dữ liệu cần chia sẻ trên thiết bị thông minh, nhấn và giữ vào dữ liệu cần chia sẻ cho đến khi xuất hiện bảng chọn.



4 Chọn Chia sẻ. Cửa sổ hiện ra như *Hình 3d.*

Hình 3c.

5 Chọn nơi cần chia sẻ (chẳng hạn, chọn Gmail). Cửa sổ hiện ra như *Hình 3c.*



6 Nhập địa chỉ mail và gửi.

Hình 3d. *Hình 3e.*

Hình 3. Thao tác chia sẻ tệp, thư mục trên thiết bị thông minh sử dụng hệ điều hành Android

Nhiệm vụ 3. Thực hành họp trực tuyến trên điện thoại thông minh

Đôi khi, trong cuộc sống, em sẽ gặp một số cản trở về mặt địa lí dẫn đến không thể gặp nhau trực tiếp để trao đổi các thông tin quan trọng (họp nhóm, làm thuyết trình, thực hiện các tiểu luận,...). Khi gặp vấn đề này, họp trực tuyến sẽ là một công cụ hữu hiệu giúp cho các em giải quyết vấn đề trên. Nhiệm vụ này sẽ giới thiệu cho các em một nền tảng họp trực tuyến miễn phí nhưng cực kì tiện lợi là Google Meet.

Yêu cầu: Hoạt động theo nhóm, sử dụng ứng dụng Google Meet để tạo một buổi họp trực tuyến trên thiết bị thông minh thông qua các thao tác sau:

a) Tạo một buổi họp trực tuyến và gửi liên kết cho các bạn trong nhóm tham dự.

b) Sử dụng điện thoại để tham gia vào cuộc họp mà bạn em đã tạo.

c) Phê duyệt các bạn đã đăng kí tham gia trong nhóm vào buổi họp.

Hướng dẫn:

a) Thực hiện các thao tác dưới đây để tạo cuộc họp trực tuyến:

1 Tải và cài đặt ứng dụng Google Meet

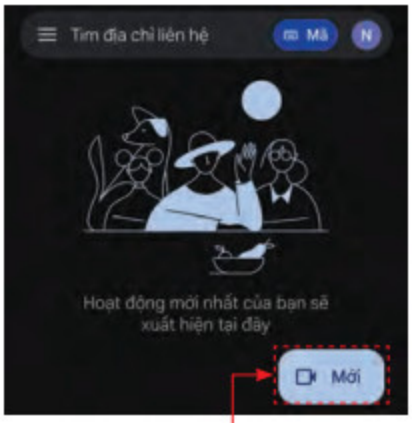
1a Mở cửa hàng ứng dụng trên điện thoại thông minh của em (cửa hàng App Store cho hệ điều hành iOS hoặc cửa hàng Google Play Store cho hệ điều hành Android, ở hướng dẫn này là cửa hàng Google Play Store).

1b Tìm kiếm "Google Meet" (*Hình 4*) trong cửa hàng ứng dụng và tải ứng dụng về máy.

Hình 4. Ứng dụng Google Meet

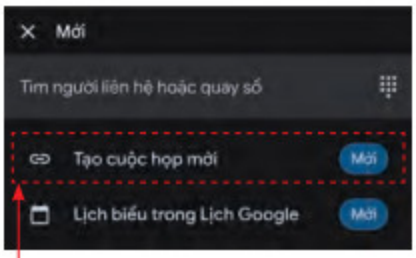
2 Đăng nhập bằng tài khoản Gmail của em. Cửa sổ hiện ra như *Hình 5b*.

Hình 5a.



3 Chọn **Mới** để tạo cuộc họp, cửa sổ hiện ra như *Hình 5c*.

Hình 5b.



4 Chọn **Tạo cuộc họp mới**, cửa sổ hiện ra như *Hình 5d*.

Hình 5c.

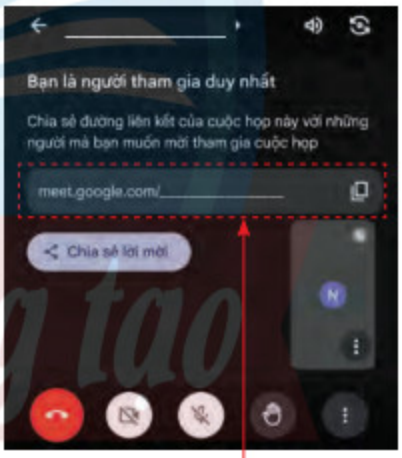
5 Chọn **Tham gia cuộc họp** để tạo cuộc họp, cửa sổ hiện ra như *Hình 5e*.

Lưu ý, ở cửa sổ này, em có thể sao chép liên kết để mời các bạn tham gia vào cuộc họp.

Hình 5d.

6 Tuỳ chỉnh các thiết bị phụ trợ phù hợp (bật/tắt camera, bật/tắt micro,...). Sau đó, chọn **Tham gia** để tham gia, cửa sổ hiện ra như *Hình 5g*.

Hình 5e.



7 Khi vào phòng họp, em là người đầu tiên trong cuộc họp, chia sẻ liên kết cho các bạn trong nhóm cùng tham gia.

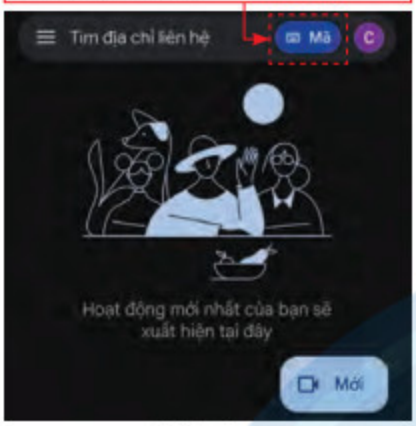
Hình 5g.

Hình 5. Thao tác tạo cuộc họp bằng Google Meet

Chân trời sáng tạo

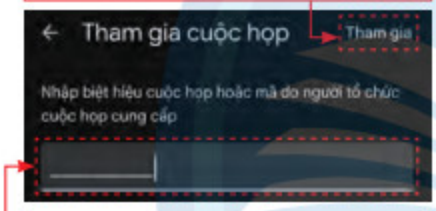
b) Thực hiện các thao tác dưới đây để tham gia cuộc họp trực tuyến của bạn em:

❶ Sau khi đăng nhập như ở Bước ❷ của Nhiệm vụ 3a, em chọn **Mã** để tham gia cuộc họp. Cửa sổ hiện ra như *Hình 6b*.



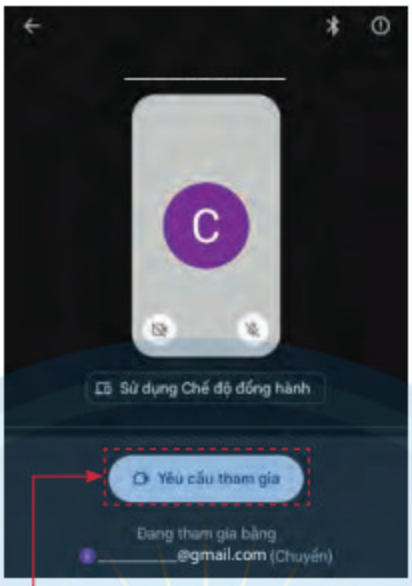
Hình 6a.

❸ Nhấn **Tham gia** để tham gia cuộc họp.



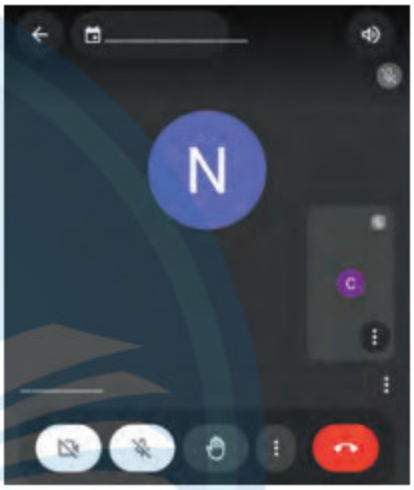
❷ Nhập mã cuộc họp mà bạn em đã chia sẻ.

Hình 6b.



❹ Tuỳ chỉnh các thiết bị phụ trợ phù hợp (bật/tắt camera, bật/tắt micro,...). Sau đó, chọn **Yêu cầu tham gia** để tham gia. Khi được cho phép tham gia cuộc họp, cửa sổ hiện ra như *Hình 6d*.

Hình 6c.



Sau khi được cho phép, bắt đầu cuộc họp.

Hình 6d.

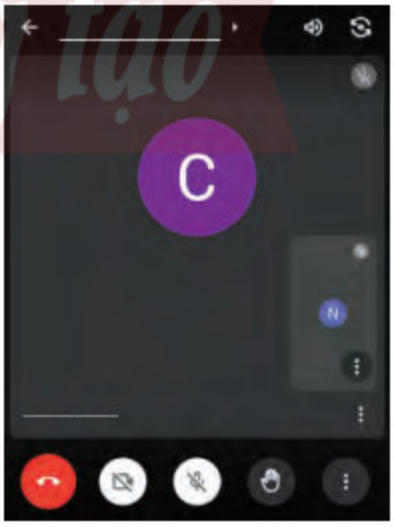
Hình 6. Thao tác tham gia cuộc họp trên Google Meet

c) Thực hiện các thao tác dưới đây để đồng ý cho các bạn em tham gia cuộc họp:

Từ bước ❼ ở Nhiệm vụ 3a, khi có tài khoản mới yêu cầu tham gia cuộc họp, sẽ xuất hiện cửa sổ như ở *Hình 7a*.

❶ Kiểm tra xem đây có phải là tài khoản của bạn em không. Nếu có thì chọn **Cho phép** để bạn em được tham gia cuộc họp. Nếu không là bạn em thì chọn **Từ chối** để ngăn chặn người dùng lạ truy cập. Khi chọn **Cho phép**, cửa sổ họp sẽ hiện ra như *Hình 7b*.

Hình 7a.

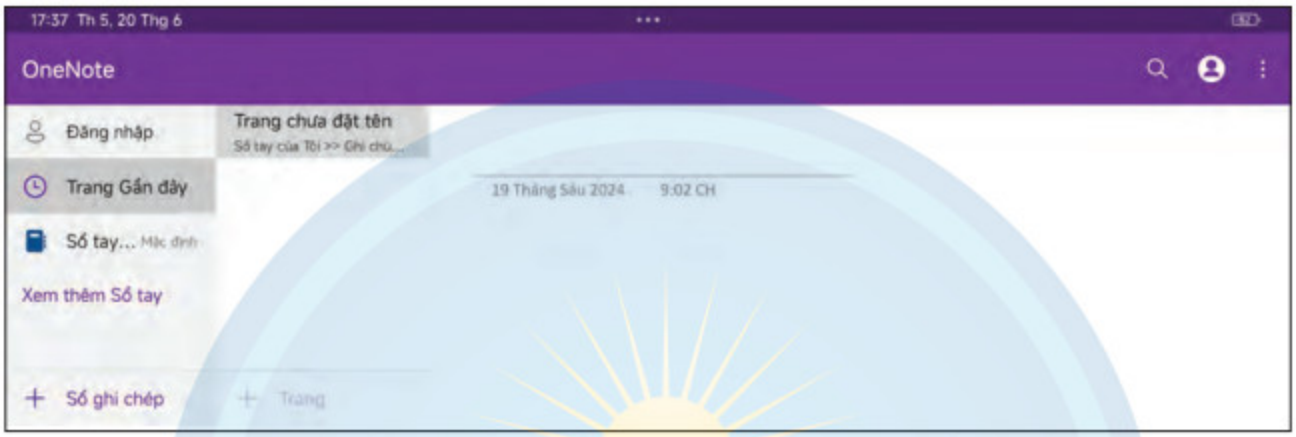


Hình 7b.

Hình 7. Thao tác phê duyệt cho thành viên tham gia họp trực tuyến

LUYỆN TẬP

1. Em hãy tìm hiểu và liệt kê các tính năng trong cuộc họp của Google Meet. Em thích tính năng nào nhất? Vì sao?
2. Em hãy tìm hiểu và trình bày các thao tác tạo cuộc họp, tham gia cuộc họp và phê duyệt cho các thành viên tham gia cuộc họp của thiết bị số chạy hệ điều hành iOS.
3. Em hãy tải xuống và sử dụng ứng dụng ghi chú OneNote (*Hình 8*) trên thiết bị số của mình để lưu lại ý tưởng hoặc lưu trữ lại các thông tin quan trọng.



Hình 8. Ứng dụng ghi chú OneNote trên máy tính bảng chạy hệ điều hành Android

VẬN DỤNG

1. Trên điện thoại thông minh, hãy tìm hiểu và sử dụng các ứng dụng Gmail và Maps để phục vụ cho các hoạt động cá nhân.
2. Trên thiết bị số của em, hãy tìm hiểu và sử dụng ứng dụng GeoGebra Geometry để phục vụ cho việc học tập bộ môn Toán học (*Hình 9*).

Hình 9a. Ứng dụng GeoGebra Geometry

Hình 9b. Các tính năng trên ứng dụng GeoGebra Geometry

Hình 9. Ứng dụng GeoGebra Geometry trên điện thoại thông minh chạy hệ điều hành iOS

BÀI A5 HỆ NHỊ PHÂN TRONG TIN HỌC

MỤC TIÊU **Sau bài học này, em sẽ:**

- Giải thích được ứng dụng của hệ nhị phân trong tin học.
- Thực hiện được các phép tính cơ bản AND, OR, NOT.

KHỞI ĐỘNG

Quy ước rằng bóng đèn ở trạng thái sáng mang tín hiệu là 1, bóng đèn ở trạng thái tắt mang tín hiệu là 0. Em hãy sử dụng các kí hiệu 0, 1 để mô tả trạng thái của 3 dãy bóng đèn ở *Hình 1*.

Hình 1. Dãy các bóng đèn

KHÁM PHÁ

1. Hệ nhị phân và ứng dụng trong tin học

a) Hệ đếm

Hệ đếm là một hệ thống các kí hiệu toán học và quy tắc sử dụng tập kí hiệu đó để biểu diễn và xác định giá trị các số. Số kí hiệu toán học được sử dụng trong hệ đếm được gọi là cơ số của hệ đếm.

Các số tự nhiên đang sử dụng ngày nay được biểu diễn theo hệ đếm thập phân (hệ đếm cơ số 10), sử dụng 10 kí hiệu (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) nhận giá trị phụ thuộc vào vị trí xuất hiện trong biểu diễn, quy tắc tính giá trị là mỗi đơn vị ở một hàng bất kì có giá trị bằng 10 đơn vị của hàng kế cận bên phải.

Các hệ đếm thường được sử dụng trong Tin học là: hệ đếm thập phân, hệ đếm nhị phân (hệ đếm cơ số 2; sử dụng 2 kí hiệu 0, 1) và hệ đếm thập lục phân (hệ đếm cơ số 16; sử dụng 16 kí hiệu 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) (*Bảng 1*).

Chân trời sáng tạo

Bảng 1. Biểu diễn của các số từ 0 tới 15 trong hệ đếm theo vị trí

Thập phân	Nhị phân	Thập lục phân	Thập phân	Nhị phân	Thập lục phân
0	0	0	8	1000	8
1	1	1	9	1001	9
2	10	2	10	1010	A
3	11	3	11	1011	B
4	100	4	12	1100	C
5	101	5	13	1101	D
6	110	6	14	1110	E
7	111	7	15	1111	F

Có thể biểu diễn cơ số của một số bằng cách ghi ở dạng chỉ số dưới, ví dụ: 101_2 nghĩa là số 101 ở hệ nhị phân, 5_{16} nghĩa là số 5 ở hệ thập lục phân.

b) Đổi biểu diễn số tự nhiên từ hệ nhị phân sang hệ thập phân và ngược lại

Số nhị phân N_2 với k + 1 chữ số (d_k, d_{k-1},..., d_1, d_0 nhận giá trị 1 hoặc 0) có biểu diễn $N_2 = d_k d_{k-1} \ldots d_1 d_0$. Chuyển đổi số N_2 sang hệ thập phân (nhận giá trị M_{10}) bằng cách tính tổng như sau:

$$N_2 = d_k \times 2^k + d_{k-1} \times 2^{k-1} + \ldots + d_1 \times 2^1 + d_0 \times 2^0 = M_{10}$$

Ví dụ 1: $1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 13_{10}$

Biểu diễn số ở hệ thập phân M_{10} được chuyển đổi sang hệ nhị phân $N_2 = d_k d_{k-1} \ldots d_1 d_0$ bằng cách thực hiện phép chia như sau: lấy M_{10} chia cho 2 (ghi lại số dư), lấy kết quả phần nguyên tiếp tục chia cho 2 (ghi lại số dư), quá trình này lặp lại đến khi kết quả phần nguyên nhận giá trị 0, dãy các số dư theo thứ tự từ dưới lên trên là số N_2 cần tìm.

Ví dụ 2: Chuyển đổi 20_{10} sang hệ nhị phân được 10100_2 (*Hình 2*).

Hình 2. Minh hoạ chuyển đổi số từ hệ thập phân (20_{10}) sang hệ nhị phân (10100_2)

c) Ứng dụng của hệ nhị phân trong tin học

Hệ nhị phân chỉ sử dụng 0 và 1 để biểu diễn các số, tương ứng với trạng thái "tắt" và "bật" của một công tắc trong mạch điện. Mỗi số nhị phân đơn lẻ (nhận giá trị 0 hoặc 1) được gọi là bit. Kết hợp 8 bit tạo thành một byte. Byte là đơn vị cơ bản để máy tính lưu trữ thông tin dưới dạng các chuỗi nhị phân như một lựa chọn đơn giản và hiệu quả để lưu trữ dữ liệu.

Việc lựa chọn hệ nhị phân trong tin học giúp đơn giản hoá việc sản xuất phần cứng máy tính với các linh kiện điện tử (bóng bán dẫn, cổng logic,...) được thiết kế để hoạt động dựa trên tín hiệu nhị phân (chỉ có 2 trạng thái 0 - tắt và 1 - bật).

Quy tắc tính toán theo hệ nhị phân cũng đơn giản và tiện lợi hơn so với các quy tắc tính toán trong hệ thập phân cũng như các hệ đếm khác, hệ nhị phân cũng tương thích hoàn toàn với các phép toán logic AND, OR, NOT (cùng sử dụng 2 trạng thái 0 - tắt và 1 - bật).

Với một dãy trạng thái của các bóng đèn như *Hình 1*, máy tính có thể biểu diễn được các số 84, 73, 78 trong hệ thập phân.

Trong lĩnh vực tin học, hệ nhị phân có những ứng dụng tiêu biểu trong các nội dung về biểu diễn dữ liệu và lưu trữ; số hoá hình ảnh và âm thanh; số hoá văn bản;...

1. Xác định tính đúng/sai của các nhận định sau đây.

a) Chỉ số dưới của một số cho biết cơ số của hệ đếm được sử dụng để biểu diễn số đó.

b) Hệ nhị phân chỉ sử dụng hai kí hiệu là 0 và 1 để biểu diễn các số.

c) Hệ thập phân chỉ biểu diễn được 10 số.

d) Hệ thập lục phân sử dụng tập kí hiệu gồm 16 số tự nhiên từ 0 đến 15 để biểu diễn các số.

2. Chuyển đổi các số 110011_2, 1000_2 sang hệ thập phân.

3. Chuyển đổi các số 256_{10}, 1023_{10} sang hệ nhị phân.

4. Tìm hiểu và chia sẻ với các bạn trong lớp về một ứng dụng của hệ nhị phân trong tin học.

Trạng thái của linh kiện điện tử trong máy tính dễ dàng biểu diễn hai giá trị tương ứng trong hệ nhị phân là 0, 1. Do đó, máy tính sử dụng hệ nhị phân để biểu diễn thông tin.

2. Các phép tính trong hệ nhị phân

a) Phép cộng và phép nhân hai số nhị phân

Phép cộng và phép nhân hai số trong hệ nhị phân được thực hiện tương tự phép cộng và phép nhân trong hệ thập phân, với lưu ý $1_2 + 1_2 = 10_2$ (*Bảng 2*). Phép cộng hai số nhị phân được thực hiện trên từng hàng từ phải sang trái (giống như cộng lần lượt hàng đơn vị, chục, trăm,... trong hệ 10). Nếu kết quả phép cộng là 10 thì ghi 0 ở hàng tương ứng và nhớ 1 sang hàng bên trái. Tương tự, nếu kết quả phép cộng là 11 thì ghi 1 ở hàng tương ứng và nhớ 1 sang hàng bên trái.

Bảng 2. Phép cộng và phép nhân trong hệ nhị phân

x	y	x + y	x × y
0	0	0	0
0	1	1	0
1	0	1	0
1	1	10	1

Ví dụ 3: $1100101_2 + 110_2 = 1101011_2$

$$\begin{array}{r} 1100101 \\ +\quad 110 \\ \hline 1101011 \end{array}$$

Ví dụ 4: $110010_2 \times 111_2 = 101011110_2$

$$\begin{array}{r} 110010 \\ \times\quad 111 \\ \hline 110010 \\ 110010 \\ 110010 \\ \hline 101011110 \end{array}$$

b) Phép tính NOT, AND, OR và XOR theo từng bit và cho các dãy bit

Dữ liệu trong máy tính đều được chuyển về dạng dãy các bit, thao tác xử lí dữ liệu cuối cùng đều dẫn đến xử lí các bit với bốn phép tính cơ bản là NOT, AND, OR và XOR (*Bảng 3*).

Bảng 3. Các phép tính cơ bản

Bảng 3a. Phép tính NOT

x	NOT x
0	1
1	0

Bảng 3b. Phép tính AND, OR và XOR

x	y	x AND y	x OR y	x XOR y
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

Quan sát các phép tính ở *Bảng 3*, em có những nhận xét như sau: phép NOT thực hiện việc phủ định trên từng bit, bit có giá trị 0 sẽ đổi sang giá trị 1 và ngược lại; phép tính AND cho kết quả là 1 khi và chỉ khi cả 2 bit đầu vào đều nhận giá trị 1, kết quả là 0 trong những trường hợp còn lại; phép tính OR cho kết quả là 0 khi và chỉ khi cả 2 bit đầu vào đều nhận

giá trị 0, kết quả là 1 trong những trường hợp còn lại; phép tính XOR cho kết quả là 1 khi và chỉ khi hai bit đầu vào có giá trị khác nhau, kết quả là 0 trong những trường hợp còn lại.

Như vậy, phép tính AND có kết quả tương tự phép nhân trong hệ nhị phân nên còn được gọi là phép nhân logic, phép tính OR có kết quả 3 dòng tương tự phép cộng trong hệ nhị phân nên còn được gọi là phép cộng logic, phép tính XOR còn được gọi là phép độc quyền.

Các phép tính cơ sở cũng được mở rộng cho các byte theo quy tắc xử lí phép tính theo từng bit hoặc từng cặp bit tương ứng.

Ví dụ 5:

x	1	0	1	0	1	0	1	1
NOT x	0	1	0	1	0	1	0	0

Ví dụ 6:

x	1	0	1	0	1	0	1	1
y	1	1	0	0	1	1	0	0
x AND y	1	0	0	0	1	0	0	0
x OR y	1	1	1	0	1	1	1	1
x XOR y	0	1	1	0	0	1	1	1

1. Thực hiện các phép tính sau:

a) $10010001_2 + 11100101_2$; b) $1010_2 \times 101_2$.

2. Xác định tính đúng/sai của các nhận định sau đây.

a) Nếu một trong hai bit là 1 thì phép AND sẽ trả về kết quả là 1.

b) Phép OR chỉ trả về kết quả là 0 nếu cả hai bit đều là 0.

c) Phép NOT đảo bit 1 thành bit 0 và ngược lại.

d) Phép XOR chỉ trả về kết quả là 0 khi cả hai bit giống nhau.

- Phép cộng và phép nhân hai số nhị phân được thực hiện tương tự phép cộng và phép nhân hai số thập phân.
- Phép NOT thực hiện việc phủ định trên từng bit, bit đầu vào có giá trị 0 sẽ cho kết quả là 1, bit đầu vào có giá trị 1 sẽ cho kết quả là 0.
- Phép AND cho kết quả là 1 khi và chỉ khi hai bit đầu vào có giá trị 1, cho kết quả là 0 trong các trường hợp còn lại.
- Phép OR cho kết quả là 0 khi và chỉ khi hai bit đầu vào có giá trị 0, cho kết quả là 1 trong các trường hợp còn lại.
- Phép XOR cho kết quả là 1 khi và chỉ khi hai bit đầu vào có giá trị khác nhau, cho kết quả là 0 trong các trường hợp còn lại.

Chân trời sáng tạo

LUYỆN TẬP

1. Xác định tính đúng/sai của các nhận định sau đây.

a) Hệ nhị phân được ứng dụng trong tin học vì máy tính không thể hiểu được các hệ đếm khác.

b) Hệ nhị phân được ứng dụng trong tin học vì chiếm ít dung lượng bộ nhớ khi biểu diễn các số tự nhiên ở hệ thập phân.

c) Hệ nhị phân được ứng dụng trong tin học vì người dùng dễ dàng nhận biết các linh kiện điện tử trong máy tính và biểu diễn hai giá trị 1 và 0 tương ứng với hai trạng thái Có và Không.

d) Hệ nhị phân được ứng dụng trong tin học vì việc xử lí các phép tính trong hệ nhị phân nhanh hơn các phép tính trong hệ thập phân.

2. Chuyển đổi các số nhị phân 11111_2, 101010101_2 sang số thập phân.

3. Chuyển đổi các số thập phân 16_{10}, 31_{10}, 129_{10} sang hệ nhị phân.

4. Thực hiện các phép tính sau đây:

a) $1000111_2 + 11010_2$;

b) $11001_2 \times 101_2$;

c) 00111000_2 AND 11001111_2;

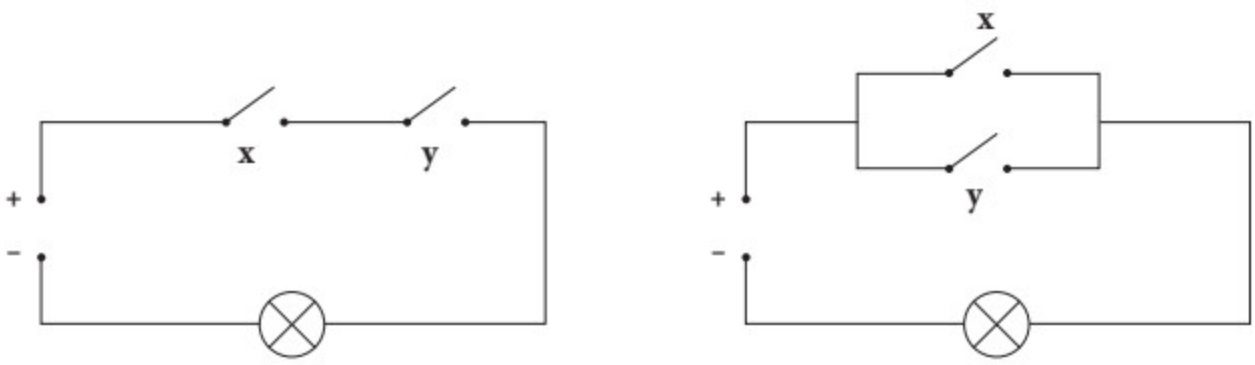
d) 10110001_2 OR 11000011_2;

e) 10110001_2 XOR 01001010_2;

g) NOT 11011101_2.

VẬN DỤNG

Quy ước rằng công tắc đóng thể hiện giá trị 1 (có dòng điện chạy qua) và công tắc mở thể hiện giá trị 0 (không có dòng điện chạy qua); đèn sáng (thể hiện giá trị 1) khi có dòng điện chạy qua và đèn tắt (thể hiện giá trị 0) khi không có dòng điện chạy qua. Ví dụ, *Hình 3a* biểu diễn mạch điện với bóng đèn và hai công tắc được mắc nối tiếp, bóng đèn sáng (nhận giá trị 1) khi và chỉ khi cả hai công tắc đều đóng (nhận giá trị 1) và bóng đèn tắt (nhận giá trị 0) trong các trường hợp còn lại. Đặt x và y là các biến logic thể hiện giá trị trạng thái của hai công tắc trong các sơ đồ mạch điện ở *Hình 3*. Hãy sử dụng các phép tính cơ bản AND, OR, XOR, NOT để thể hiện giá trị trạng thái của đèn theo x và y.



Hình 3a *Hình 3b*

Hình 3. Sơ đồ mạch điện

BÀI A6 SỐ HOÁ HÌNH ẢNH, ÂM THANH

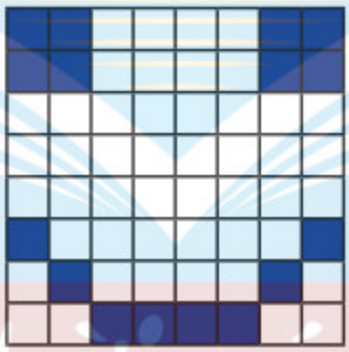
MỤC TIÊU **Sau bài học này, em sẽ:**

- Giải thích được sơ lược về việc số hoá hình ảnh, âm thanh.

KHỞI ĐỘNG

Quan sát lưới ô vuông 8×8 ở *Hình 1*, có một số ô đã được tô màu để tạo thành hình mặt cười. Với quy ước ghi giá trị 1 cho các ô được tô màu, giá trị 0 cho các ô không được tô màu, hình mặt cười có thể được ghi thành dãy các chuỗi nhị phân, liên kết với nhau bằng dấu chấm như sau:

11000011.11000011.00000000.00000000.00000000.10000001.01000010.00111100



Hình 1. Lưới ô vuông vẽ hình mặt cười

Sử dụng lưới ô vuông 8×8 và quy ước ghi thành dãy các chuỗi nhị phân như trên, em dựng một số hình đơn giản (trái tim, ngôi sao, mũi tên), chuyển dãy các chuỗi nhị phân tương ứng cho bạn để dựng lại trên lưới ô vuông 8×8 của bạn và kiểm tra tính chính xác giữa hình ảnh của bạn với hình ảnh của em. Em có nhận xét gì về quy trình chuyển đổi từ hình ảnh sang dãy các chuỗi nhị phân vừa thực hiện?

KHÁM PHÁ

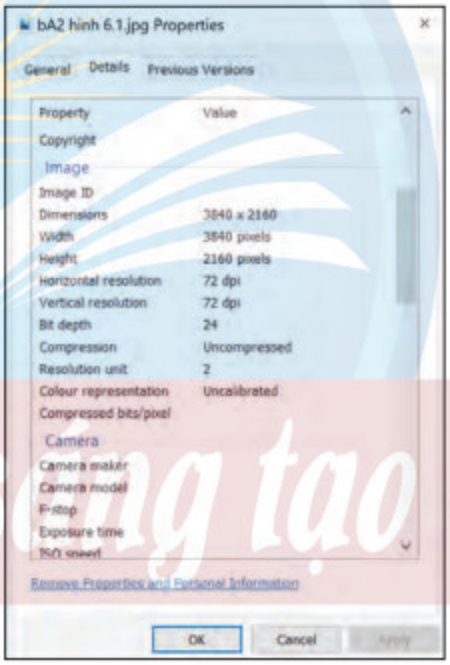
1. Số hoá hình ảnh

Để bảo tồn hình ảnh được lâu dài, tránh nguy cơ hư hỏng, mất mát đồng thời tạo thuận lợi hơn cho việc chia sẻ, truy cập và lưu trữ, việc chuyển đổi hình ảnh từ dạng vật lí sang dạng số (số hoá hình ảnh) là một yêu cầu cần thiết và mang lại nhiều lợi ích.

Lưới ô vuông được sử dụng để chia một hình ảnh ở dạng vật lí thành nhiều ô vuông rất nhỏ, được đánh thứ tự từ trái sang phải, từ trên xuống dưới. Mỗi ô vuông mang 1 màu đồng nhất, gọi là một phần tử ảnh (pixel), cũng được gọi là điểm ảnh theo cách nói thông dụng hằng ngày (*Hình 2*). Việc chia nhỏ hình ảnh như trên được gọi là quy trình rời rạc hoá hình ảnh.

Hình 2. Hình ảnh sau khi được rời rạc hoá thành các điểm ảnh

Độ phân giải của hình ảnh được tính bằng tích của hai số đếm điểm ảnh theo chiều ngang và chiều dọc, là tổng số điểm ảnh được chia nhỏ. Cùng một kích thước hình ảnh, độ phân giải cao đồng nghĩa với kích thước một điểm ảnh nhỏ, ảnh mịn và ngược lại, độ phân giải thấp đồng nghĩa với kích thước một điểm ảnh to, ảnh thô. Ví dụ, khi nháy phải chuột vào một tệp ảnh, chọn **Properties**, trong cửa sổ hiện ra, chọn thẻ **Details** (*Hình 3*) và đọc được kích thước của hình ảnh là 3840×2160. Như vậy ảnh có chiều rộng 3840 điểm ảnh, chiều cao 2160 điểm ảnh và tổng cộng được chia thành 8 294 400 điểm ảnh.



Hình 3. Thuộc tính kích thước của hình ảnh

Sau khi đã rời rạc hoá hình ảnh thành các điểm ảnh mang màu đồng nhất, em cần chuyển đổi các điểm ảnh này thành các chuỗi số nhị phân mang thông tin màu sắc để có thể bắt đầu việc số hoá toàn bộ hình ảnh.

Hệ màu được sử dụng phổ biến trong các thiết bị hiển thị hình ảnh hiện nay là hệ màu RGB, dựa trên 3 màu cơ sở là Đỏ (Red), Xanh lục (Green), Xanh dương (Blue). Ba màu cơ sở được trộn chung với nhau theo tỉ lệ khác nhau (cường độ khác nhau) tạo ra những màu sắc khác nhau. Trong hệ màu RGB, cường độ mỗi màu được biểu diễn bằng 1 byte (8 bit) nên còn

được gọi là hệ màu 24 bit. Mỗi màu được biểu diễn bằng bộ 3 số nguyên (a, b, c), mỗi số lấy giá trị từ 0 đến 255 biểu diễn cường độ của các màu Đỏ, Xanh lục, Xanh dương tương ứng. Giá trị mỗi màu được viết ở dạng bộ 3 số nguyên hoặc ở dạng nhị phân 24 bit hay dạng thập lục phân. *Bảng 1* ví dụ một số màu và biểu diễn giá trị màu ở các dạng khác nhau.

Bảng 1. Biểu diễn một số màu trong hệ RGB dưới dạng số nguyên, nhị phân, thập lục phân

Màu	Bộ 3 số nguyên	Dạng nhị phân	Dạng thập lục phân
Đen	(0, 0, 0)	00000000 00000000 00000000	00 00 00
Trắng	(255, 255, 255)	11111111 11111111 11111111	FF FF FF
Đỏ	(255, 0, 0)	11111111 00000000 00000000	FF 00 00
Xanh lục	(0, 255, 0)	00000000 11111111 00000000	00 FF 00
Xanh dương	(0, 0, 255)	00000000 00000000 11111111	00 00 FF
Vàng	(255, 255, 0)	11111111 11111111 00000000	FF FF 00

Sau khi rời rạc hoá hình ảnh, sắp xếp các chuỗi nhị phân mang thông tin màu sắc của từng điểm ảnh theo thứ tự từ trái sang phải, từ trên xuống dưới, em sẽ nhận được dữ liệu số hoá của hình ảnh đó dưới dạng bản đồ các bit màu sắc, được gọi là ảnh bitmap, lưu vào các tệp có phần mở rộng là .bmp.

Ảnh bitmap có ưu điểm là lưu trữ được toàn bộ thông tin màu sắc của từng điểm ảnh nên đảm bảo được chất lượng hình ảnh không bị mất mát và có khả năng tái hiện nguyên vẹn hình ảnh ban đầu trên các thiết bị hiển thị hình ảnh khác. Tuy nhiên, ảnh bitmap có nhược điểm chiếm rất nhiều dung lượng khi lưu trữ. Do đó, việc tối ưu hoá lưu trữ dữ liệu hình ảnh số được triển khai theo hai phương pháp: nén tệp, lúc xem ảnh thì giải nén, không gây giảm chất lượng (tệp lưu trữ có phần mở rộng .png); chủ động giảm bớt một số thông tin, chịu mất một phần chất lượng hình ảnh khi xem (tệp lưu trữ có phần mở rộng .jpeg hoặc .jpg).

Ngày nay, việc số hoá hình ảnh có thể được thực hiện bởi các thiết bị thông dụng như máy ảnh số, điện thoại thông minh, máy quét hình ảnh,…

1. Biết rằng màu xanh ngọc được biểu diễn bởi bộ 3 số nguyên (0, 255, 255), hãy cho biết dạng nhị phân và dạng thập lục phân tương ứng dùng để biểu diễn màu xanh ngọc.

2. Xác định tính đúng/sai của các nhận định sau đây về hình ảnh số có phần mở rộng .jpeg.

a) Kích thước tệp nhỏ, tiết kiệm được chi phí lưu trữ, tải trên các trang web được nhanh hơn.

b) Chất lượng ảnh vẫn được đảm bảo như các ảnh bitmap.

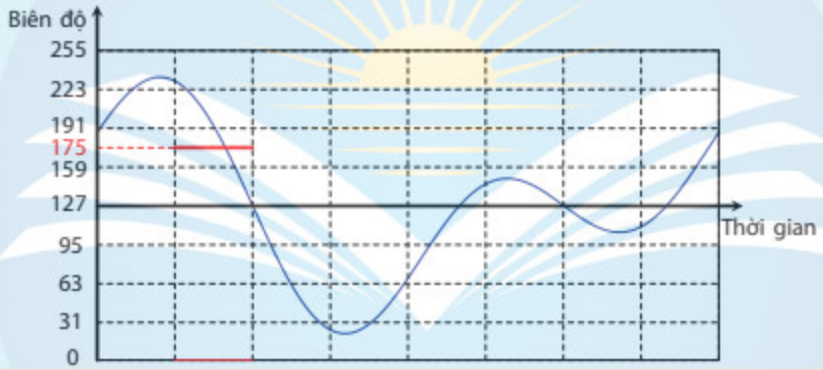
c) Các website hiện nay không sử dụng các định dạng ảnh nào khác ngoài định dạng .jpeg.

d) Ảnh lưu trữ với phần mở rộng .jpeg là ảnh nén, cần được giải nén khi xem.

- Số hoá hình ảnh giúp bảo tồn hình ảnh được lâu dài, tránh nguy cơ hư hỏng, tạo điều kiện thuận lợi cho việc chia sẻ, truy cập và lưu trữ.
- Để số hoá hình ảnh, cần rời rạc hoá hình ảnh thành các điểm ảnh mang màu đồng nhất, sau đó sắp xếp các chuỗi nhị phân mang thông tin màu sắc của từng điểm ảnh theo thứ tự từ trái sang phải, từ trên xuống dưới, tạo thành dữ liệu số hoá của hình ảnh.
- Hình ảnh số hoá có thể được lưu thành tệp có phần mở rộng .bmp, .png, .jpeg, .jpg,...

2. Số hoá âm thanh

Âm thanh được truyền đi bằng sóng âm qua môi trường. Trong thực tế, sóng âm có dạng hình sin (*Hình 4*) với trục hoành thể hiện thời gian, trục tung thể hiện biên độ của tín hiệu âm thanh. Tín hiệu âm thanh có đồ thị liên tục như *Hình 4* được gọi là tín hiệu âm thanh tương tự (analog). Cùng với sự phát triển của văn hoá nghệ thuật, nhu cầu về việc bảo tồn, lưu trữ, chia sẻ, xử lí các bản ghi âm thanh và lưu trữ dưới dạng số (số hoá âm thanh) là một yêu cầu cần thiết, mang lại nhiều lợi ích.



Hình 4. Chu kì lấy mẫu và thang giá trị mẫu

Chân trời sáng tạo

Phương pháp cơ bản để số hoá âm thanh là điều chế mã xung (Pulse Code Modulation – PCM), được thực hiện theo các bước như sau:

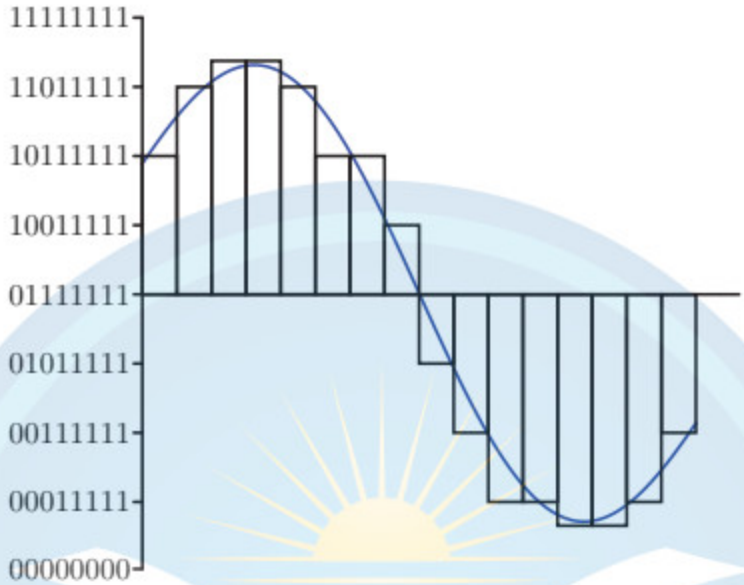
1 Lấy mẫu

Giá trị biên độ của tín hiệu âm thanh được đo liên tục tại những thời điểm rời rạc, cách đều nhau. Khoảng thời gian giữa hai lần lấy mẫu được gọi là chu kì lấy mẫu. Số mẫu ghi nhận trong 1 giây được gọi là tần số lấy mẫu. Tần số lấy mẫu chuẩn âm thanh chất lượng cao được chọn ở mức 44.1 kHz (44 100 mẫu âm thanh ghi nhận mỗi giây), gấp đôi tần số tín hiệu nghe được, giúp hạn chế việc mất thông tin.

2 Biểu diễn giá trị mẫu

Chọn thang đo giá trị biên độ bao gồm các mức đều nhau, ví dụ 256 mức ($256 = 2^8$, các mức biểu diễn được bằng chuỗi nhị phân có độ dài 8 bit). Biên độ tín hiệu được quy đổi theo tỉ lệ trên thang lấy mẫu và làm tròn. Trong *Hình 4*, biên độ tín hiệu đo được trong chu kì lấy mẫu

được quy đổi thành giá trị 175, tương ứng với chuỗi nhị phân 10101111_2. *Hình 5* (đồ thị hình bậc thang) thể hiện một phần tín hiệu âm thanh ở *Hình 4* (đồ thị hình sin) sau khi được lấy mẫu và biểu diễn giá trị dưới dạng các chuỗi nhị phân. Thang đo càng có nhiều mức (chuỗi nhị phân càng dài), âm thanh ghi nhận được càng sát với âm thanh gốc (sắc nét, trung thực), nên độ dài của chuỗi nhị phân biểu diễn mẫu âm thanh còn được gọi là độ nét của âm thanh (resolution). Các bản nhạc số hiện nay đang sử dụng độ nét ở mức 8 bit, 16 bit, 24 bit.



Hình 5. Lượng tử hoá tín hiệu âm thanh thành các giá trị số tương ứng với các mức gần nhất

3 Biểu diễn âm thanh

Toàn bộ các giá trị biên độ sau khi quy đổi được ghi lại theo trình tự thời gian, tạo thành biểu diễn âm thanh dưới dạng số, lưu thành tệp với phần mở rộng .wav. Ví dụ, với tần suất lấy mẫu 16 kHz, độ nét 8 bit, lượng dữ liệu âm thanh ghi nhận được mỗi giây là $8 \times 16000 = 128000$ bit, kí hiệu 128 kbps. Lượng bit được dùng để ghi nhận dữ liệu âm thanh mỗi giây được gọi là tốc độ bit (bitrate).

Trong thực tế, người ta sử dụng các thiết bị ghi âm với mạch điện tử chuyển đổi tín hiệu tương tự sang tín hiệu số (Analog to Digital Converter - ADC) để thực việc việc số hoá âm thanh. Các thiết bị phát âm thanh số có mạch điện tử chuyển đổi tín hiệu số về lại tín hiệu tương tự (Digital to Analog Converter - DAC) để phát âm thanh ra loa hoặc tai nghe.

Phương pháp số hoá âm thanh PCM cho chất lượng âm thanh trung thực nhưng có nhược điểm là kích thước tệp tin âm thanh số khá lớn. Để việc chia sẻ được thuận tiện hơn, người ta đã đưa ra 2 phương pháp chính để giảm kích thước tệp: nén dữ liệu khi lưu trữ, giải nén khi nghe, không làm giảm chất lượng âm thanh (định dạng lossless, tệp lưu trữ có phần mở rộng .flac, .alac,…); bỏ bớt một phần thông tin âm thanh (giảm tần số lấy mẫu, giảm độ nét, giảm số kênh lưu trữ,…), đảm bảo chất lượng âm thanh ở mức chấp nhận được (định dạng lossy, tệp lưu trữ có phần mở rộng .mp3, .m4a, .ogg,…).

1. Tốc độ bit 320 kbps (còn được viết là 320 kb/s) nghĩa là gì?

2. Bạn Nam nhận được một tệp âm thanh số hoá với tốc độ bit 128 kbps, dung lượng khoảng 50 MB. Hỏi tệp âm thanh của bạn Nam nhận được có thời lượng dài khoảng bao nhiêu phút?

- Số hoá âm thanh giúp bảo tồn các bản ghi âm thanh được lâu dài, tránh nguy cơ hư hỏng, tạo điều kiện thuận lợi cho việc chia sẻ, lưu trữ, chỉnh sửa và nâng cao chất lượng âm thanh.
- Phương pháp cơ bản để số hoá âm thanh là điều chế mã xung, được thực hiện theo các bước: Lấy mẫu; Biểu diễn giá trị mẫu; Biểu diễn giá trị âm thanh.
- Dữ liệu âm thanh sau khi số hoá có thể được lưu thành tệp có phần mở rộng .wav, .flac, .mp3, .m4a, .ogg,...

LUYỆN TẬP

1. Thảo luận với bạn và sử dụng lưới ô vuông 8×8 để số hoá một hình đơn giản có màu sắc (mặt cười, ngôi sao, trái tim, mũi tên,...), sử dụng dạng biểu diễn thập lục phân của màu sắc và chuyển dữ liệu số hoá cho nhóm khác dựng lại hình ảnh, sau đó kiểm tra mức độ trùng khớp giữa hai hình ảnh.
2. Khi lưu trữ trên máy tính, với cùng độ phân giải, ảnh màu hay ảnh trắng đen có dung lượng lớn hơn? Tại sao?
3. Chất lượng trình diễn âm thanh số hoá có phụ thuộc vào tốc độ bit (bit rate) hay không? Tại sao?

VẬN DỤNG

1. Một đĩa nhạc nén .mp3 với nhiều tệp âm thanh có tốc độ bit 128 kbps và tổng dung lượng là 580 MB. Hãy ước lượng tổng thời gian của tất cả các tệp âm thanh lưu trong đĩa nhạc nén trên.
2. Để chuẩn bị tài liệu cho dự án tìm hiểu về một di tích lịch sử tại địa phương, bạn An cần chụp 10 bức ảnh và ghi 5 tệp âm thanh bằng điện thoại thông minh của mình. Biết rằng: mỗi bức ảnh có độ sâu màu 24 bit, kích thước 2000×1000 pixels; mỗi tệp âm thanh có độ dài 5 phút, tần suất 44.1 kHz, được biểu diễn bởi chuỗi nhị phân 8 bit. Hãy ước tính tổng dung lượng cần chuẩn bị trên điện thoại thông minh của bạn An để lưu trữ các tài liệu trên.

Chân trời sáng tạo

BÀI A7 SỐ HOÁ VĂN BẢN, BẢNG MÃ CHUẨN QUỐC TẾ

MỤC TIÊU **Sau bài học này, em sẽ:**

- Giải thích sơ lược về việc số hoá văn bản.
- Giải thích sơ lược về chức năng của bảng mã chuẩn quốc tế (ví dụ: Unicode).

KHỞI ĐỘNG

Quy ước mỗi kí tự (không dấu, không phân biệt chữ thường, chữ in hoa) tương ứng với thứ tự trong bảng chữ cái (*Bảng 1*), bạn Minh đã chuyển một "mật thư" đến bạn Lan với nội dung như sau: 20 15 15 09 20 08 09 03 08 19 08 15 03 10 20 09 14 08 15 03 10

Bảng 1. Đặt tương ứng mỗi kí tự với thứ tự trong bảng chữ cái

A	B	C	D	E	F	G	H	I	J	K	L	M
01	02	03	04	05	06	07	08	09	10	11	12	13

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	26

Bạn Minh đã chuyển nội dung gì cho bạn Lan? Em và các bạn áp dụng quy tắc tương tự để tạo mật thư và trao đổi với nhau.

KHÁM PHÁ

1. Bảng mã ASCII

Tương tự với việc số hoá hình ảnh và âm thanh, việc chuyển đổi văn bản từ dạng vật lí sang dạng số (số hoá văn bản) cũng là một yêu cầu cần thiết và mang lại nhiều lợi ích. Số hoá văn bản giúp tiết kiệm không gian và chi phí lưu trữ văn bản giấy, sách cùng những tài liệu in; bảo vệ thông tin tránh khỏi các hư hỏng do thời tiết, thiên tai,…; tạo điều kiện thuận lợi cho việc quản lí, bảo mật, chia sẻ, truy cập, cộng tác và tìm kiếm văn bản.

Phát triển từ bảng mã điện tín, bảng mã ASCII (American Standard Code for Information Interchange) là bảng mã kí tự đầu tiên được thiết kế để số hoá các kí tự trong máy tính. Được công bố là bảng mã kí tự tiêu chuẩn từ năm 1963 bởi Hiệp hội tiêu chuẩn Hoa Kì (American Standards Association, ASA), bảng mã ASCII được sử dụng rộng rãi trên hầu hết tất cả các máy tính và thiết bị điện tử.

Bảng mã ASCII bao gồm 128 kí tự với 33 kí tự không in được (kí tự điều khiển) và 95 kí tự in được (các chữ cái, chữ số và các kí tự đặc biệt). Mỗi kí tự trong bảng mã ASCII được biểu diễn bằng một số nguyên nhận giá trị từ 0 đến 127 và sử dụng chuỗi nhị phân 7 bit (điểm mã – code point) để lưu trữ trong máy tính (*Bảng 2*).

Bảng 2. Một số giá trị kí tự trong bảng mã ASCII

Nhị phân	Thập phân	Nội dung
011 0000	48	0
011 0001	49	1
011 0010	50	2
011 0011	51	3
011 0100	52	4
011 0101	53	5
011 0110	54	6
011 0111	55	7
011 1000	56	8
011 1001	57	9
011 1010	58	:
011 1011	59	;
011 1100	60	<
011 1101	61	=
011 1110	62	>
011 1111	63	?
100 0000	64	@
100 0001	65	A
100 0010	66	B
100 0011	67	C
100 0100	68	D
100 0101	69	E

Nhị phân	Thập phân	Nội dung
100 0110	70	F
100 0111	71	G
100 1000	72	H
100 1001	73	I
100 1010	74	J
100 1011	75	K
100 1100	76	L
100 1101	77	M
100 1110	78	N
100 1111	79	O
101 0000	80	P
101 0001	81	Q
101 0010	82	R
101 0011	83	S
101 0100	84	T
101 0101	85	U
101 0110	86	V
101 0111	87	W
101 1000	88	X
101 1001	89	Y
101 1010	90	Z
101 1011	91	[

Với 128 kí tự, bảng mã ASCII chỉ vừa đủ sử dụng cho việc số hoá các kí tự tiếng Anh. Để có thể số hoá các kí tự đặc biệt của các ngôn ngữ khác (tiếng Hy Lạp có các kí tự α, β,…; tiếng Nga có các kí tự Б, Д,…), bảng mã ASCII được mở rộng từ 128 kí tự (sử dụng chuỗi nhị phân 7 bit) tăng lên thành 256 kí tự (sử dụng chuỗi nhị phân 8 bit). Trong bảng mã ASCII mở rộng (bảng mã ASCII 8 bit), 128 kí tự đầu tiên được giữ nguyên so với bảng mã ASCII 7 bit, 128 điểm mã bổ sung được các quốc gia sử dụng để gán mã cho các kí tự riêng của mình.

Như vậy, với bảng mã ASCII mở rộng, các quốc gia đã có thể biểu diễn văn bản thuộc ngôn ngữ của mình dưới dạng dãy các chuỗi nhị phân 8 bit, đặt tiền đề cho việc số hoá văn bản.

Mặc dù có nhiều bản mã khác đã được phổ biến với số lượng nhiều hơn các kí tự được mã hoá, bảng mã ASCII vẫn được sử dụng rộng rãi, đặc biệt trong các giao thức truyền thông như HTTP và SMTP. Bảng mã ASCII giữ một vị trí và vai trò quan trọng trong lịch sử phát triển của ngành Công nghệ thông tin.

Tìm hiểu về bản mã ASCII 7 bit, em hãy xác định mã nhị phân của các kí tự C, T, S, T và tiến hành mã hoá (biểu diễn bằng dãy các chuỗi nhị phân 7 bit) dòng văn bản sau:

Chan Troi Sang Tao

- ASCII là bảng mã chuẩn đầu tiên được xây dựng để mã hoá kí tự thành chuỗi nhị phân có độ dài 7 bit.
- Bảng mã ASCII mở rộng sử dụng các chuỗi nhị phân có độ dài 8 bit để mã hoá 256 kí tự: 128 kí tự của bản ASCII 7 bit và 128 điểm mã còn lại dùng cho các nước mã hoá những kí tự đặc biệt trong ngôn ngữ của mình.

2. Bảng mã Unicode và tiếng Việt trong Unicode

Với sự phổ biến của công nghệ thông tin trên thế giới, bảng mã ASCII mở rộng với 256 kí tự không thể đáp ứng được yêu cầu mã hoá đối với các ngôn ngữ như tiếng Trung Quốc, tiếng Hàn Quốc, tiếng Ả Rập,… (sử dụng nhiều kí tự khác các chữ cái La Tinh). Bên cạnh đó, sự xuất hiện nhiều ngôn ngữ trong cùng một văn bản cũng đặt ra yêu cầu về việc phải thiết kế một bảng mã duy nhất có khả năng mã hoá được gần như tất cả các kí tự phổ biến của nhiều ngôn ngữ trên thế giới.

Năm 1991, Hiệp hội Unicode đã công bố phiên bản đầu tiên của Tiêu chuẩn Unicode (The Unicode Standard) như một bộ mã chuẩn quốc tế, được thiết kế để làm bộ mã duy nhất cho tất cả các ngôn ngữ khác nhau trên thế giới, từ ngôn ngữ sử dụng bộ chữ cái La Tinh kết hợp với các kí tự riêng (tiếng Anh, tiếng Việt, tiếng Nga, tiếng Pháp,…), cho đến các ngôn ngữ sử dụng kí tự tượng hình phức tạp (tiếng Trung Quốc, tiếng Nhật Bản, tiếng Hàn Quốc,…), hỗ trợ được cả các ngôn ngữ viết từ phải sang trái (tiếng Ả Rập).

Bảng mã Unicode sử dụng chuỗi có độ dài tối đa 32 bit (4 byte) để mã hoá các kí tự với tổng cộng hơn 4,2 tỉ điểm mã, mỗi kí tự được gán một điểm mã duy nhất (cách mã hoá UTF-32, 32 bit Unicode Tranformation Unicode).

Bên cạnh việc sử dụng chuỗi nhị phân có độ dài 4 byte để mã hoá kí tự, nhằm tạo thuận lợi cho việc phổ biến và lập trình, bảng mã Unicode cho phép sử dụng chuỗi thập lục phân 8 kí tự tương ứng với 4 byte nhị phân (đặt liền sau chuỗi U+, ví dụ U+0000004D là mã hoá của kí tự M theo cách mã hoá UTF-32). Cách mã hoá UTF-8 sử dụng chuỗi 8 bit nên tự động tương thích với ASCII mở rộng, qua đó tạo ra lợi thế lớn trong việc chuyển đổi các phần mềm có sẵn (chỉ làm việc với các biến có độ dài 8 bit) sang hỗ trợ Unicode, sử dụng được đa ngôn ngữ. Vì phần lớn kí tự thuộc các ngôn ngữ dùng bảng chữ cái La Tinh (tiếng Việt, tiếng Anh, tiếng Pháp, tiếng Tây Ban Nha,…) đều nằm trong 256 điểm mã đầu tiên, sử dụng UTF-8 là một giải pháp tối ưu cho việc mã hoá kí tự trong các phần mềm soạn thảo văn bản, phần mềm đa ngôn ngữ hiện nay.

Năm 2001, Bộ Khoa học, Công nghệ và Môi trường ban hành TCVN 6909:2001, quy định bộ mã tiếng Việt 16-bit với tập kí tự cơ bản (119 kí tự thuộc 6 nhóm: chữ cái, dấu thanh, dấu nguyên âm, chữ số, kí hiệu, dấu câu) và tập kí tự mở rộng (120 kí tự nguyên âm mang dấu thanh), phù hợp với bộ mã Unicode (phiên bản 3.0) về vị trí trong mặt phẳng đa ngôn ngữ căn bản. Trong Phụ lục 1 kèm theo Nghị định số 30/2020/NĐ-CP về công tác văn thư, do Thủ tướng Chính phủ ban hành ngày 05/03/2020, việc sử dụng phông chữ Times New Roman, bộ mã kí tự Unicode theo TCVN 6909:2001, màu đen được quy định thành thể thức và kĩ thuật trình bày văn bản hành chính ở Việt Nam.

Với quy định về việc các ngôn ngữ phải sử dụng lại những kí tự đã được xác định trước, Unicode tránh được tình trạng thiếu nhất quán do các ngôn ngữ sử dụng cùng một kí tự nhưng tự gán mã khác nhau. Do đó, Unicode được sử dụng như chuẩn mã hoá văn bản trong những ứng dụng đa ngôn ngữ như trình duyệt web, trình biên dịch phần mềm, các trò chơi điện tử trực tuyến,... Phiên bản mới nhất của bảng mã Unicode (phiên bản 15.1.0) được công bố vào ngày 12/09/2023, mã hoá được tổng cộng 149 813 kí tự của gần như toàn bộ các ngôn ngữ đang được sử dụng trên thế giới.

Tìm hiểu về bộ mã kí tự tiếng Việt 16 bit (quy định trong TCVN 6909:2001), em xác định mã thập lục phân của các kí tự (từ U+0020 đến U+201D) và mã hoá dòng văn bản sau:

Em học sách Tin học Chân trời Sáng tạo

- Bảng mã Unicode được sử dụng như chuẩn mã hoá văn bản trong những ứng dụng đa ngôn ngữ.
- TCVN 6909:2001, quy định bộ mã tiếng Việt 16-bit với tập kí tự cơ bản, tập kí tự mở rộng, đồng nhất điểm mã với các kí tự tương đương trong bộ mã Unicode (phiên bản 3.0) trong mặt phẳng đa ngôn ngữ căn bản.

3. Số hoá văn bản

Việc chuyển đổi văn bản từ dạng vật lí sang dạng tệp tin (số hoá văn bản) là quá trình chuyển đổi toàn bộ dữ liệu trong văn bản (kí tự, hình ảnh) thành các chuỗi nhị phân (số hoá kí tự, số hoá hình ảnh) và lưu trữ trong các tệp ở bộ nhớ ngoài.

Việc số hoá văn bản có thể được thực hiện một cách thủ công bằng cách sử dụng các phần mềm soạn thảo văn bản (Notepad, MS Word, Open Office,...) và sử dụng phần mềm bộ gõ (Unikey, Vietkey,...) để nhập từng kí tự trong văn bản vào tệp văn bản trong máy tính. Sử dụng bảng mã Unicode, với mỗi kí tự được nhập từ bàn phím, kết hợp cùng với kiểu gõ, quy tắc bỏ dấu (dấu thanh, dấu mũ), bộ gõ tiếng Việt sẽ mã hoá thành mã kí tự tiếng Việt tương ứng và lưu trữ vào tệp văn bản. Ví dụ, với kiểu gõ Telex, cách đánh dấu ngay sau nguyên âm, cách mã hoá UTF-8, hai kí tự "a", "s" nhập liền kề từ bàn phím sẽ được kết hợp và đặt tương ứng với 1 mã kí tự Unicode U+E1 (chữ a thường với dấu sắc) lưu trữ trong tệp văn bản.

Ngày nay, việc số hoá văn bản có thể được thực hiện một cách tự động bằng cách sử dụng máy quét văn bản hoặc các ứng dụng thông minh được trang bị công nghệ nhận dạng kí tự quang học (Optical Character Recognition – OCR). Sau khi văn bản được quét và lưu trữ dưới dạng tệp hình ảnh, công nghệ OCR sẽ tiến hành quét trên tệp hình ảnh vừa được lưu trữ để nhận dạng từng kí tự và chuyển đổi thành văn bản có thể chỉnh sửa được (*Hình 1*).

Hình 1. Sử dụng ứng dụng công nghệ OCR trên điện thoại thông minh để số hoá văn bản

Việc số hoá văn bản tự động tuy có lợi thế về tốc độ triển khai nhưng chưa đạt được độ chính xác cao như khi nhập văn bản thủ công. Tuy nhiên, với sự phát triển của công nghệ nhận dạng, sự hỗ trợ của các thành tựu trí tuệ nhân tạo, số hoá văn bản tự động sẽ là một lựa chọn tối ưu dành cho việc chuyển đổi số lượng lớn các văn bản từ dạng vật lí sang dạng số. Đặc biệt, đối với thư viện và các cơ sở nghiên cứu, số hoá văn bản một cách tự động sẽ đáp ứng được nhu cầu số hoá và bảo tồn các tác phẩm kinh điển, chia sẻ rộng rãi những bản sách hay và quý hiếm, mở rộng phạm vi người sử dụng, bảo quản sách và tài liệu được lâu hơn,...

- Số hoá văn bản là quá trình chuyển nội dung văn bản từ dạng vật lí sang dạng tệp.
- Số hoá văn bản có thể thực hiện thủ công bằng cách sử dụng phần mềm soạn thảo văn bản kết hợp với phần mềm bộ gõ. Số hoá văn bản cũng có thể thực hiện tự động bằng cách sử dụng máy quét văn bản có trang bị công nghệ OCR.

LUYỆN TẬP

1. Hãy xác định tính đúng/sai của các nhận định sau đây về bảng mã Unicode:

a) Cho phép tất cả các ngôn ngữ của các nước trên thế giới đều có thể mã hoá được trong cùng một bảng mã chung.

b) Không hỗ trợ việc sử dụng nhiều ngôn ngữ đồng thời trong cùng một ứng dụng.

c) Chỉ hỗ trợ các kí tự tiếng Anh cơ bản.

d) Không hỗ trợ các kí tự đặc biệt, kí tự tượng hình, kí tự Toán học.

2. Việc số hoá văn bản tự động có thể thực hiện qua các ứng dụng trên điện thoại thông minh. Em hãy tìm hiểu và giới thiệu với các bạn một ứng dụng dùng để chuyển đổi ảnh chụp văn bản thành nội dung tệp văn bản có thể chỉnh sửa được.

VẬN DỤNG

1. Dựa vào bảng mã ASCII 7 bit, em hãy viết một tin ngắn (dùng mã thập phân) để trao đổi với bạn về nội dung học môn Tin học ở tiết học sau.

2. Công nghệ OCR đã được phát triển và ứng dụng trong nhiều lĩnh vực trong cuộc sống. Em và các bạn cùng tìm hiểu và giới thiệu về một ứng dụng thực tế của công nghệ OCR.

Chân trời sáng tạo

Chủ đề B

MẠNG MÁY TÍNH VÀ INTERNET

BÀI B1 MẠNG MÁY TÍNH, INTERNET VÀ TÁC ĐỘNG ĐẾN CUỘC SỐNG

MỤC TIÊU **Sau bài học này, em sẽ:**

- So sánh được mạng LAN và Internet.
- Trình bày được những thay đổi về chất lượng cuộc sống, phương thức học tập và làm việc trong xã hội mà ở đó mạng máy tính được sử dụng phổ biến.

KHỞI ĐỘNG

Quan sát *Hình 1* và cho biết, ứng dụng nào có thể sử dụng mà không cần Internet?

Hình 1a. Google *Hình 1b. Zalo* *Hình 1c. Word*

Hình 1. Các ứng dụng trên máy tính

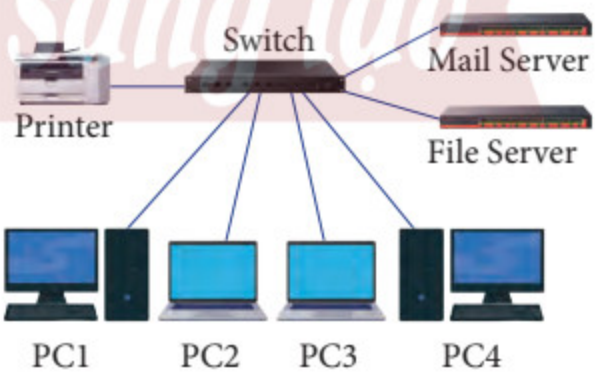
KHÁM PHÁ

1. Mạng nội bộ và mạng Internet

Trong thời đại số ngày nay, các hệ thống mạng đóng vai trò quan trọng trong việc kết nối và truyền tải dữ liệu giữa các thiết bị và người dùng. Trong số các loại mạng, mạng nội bộ và mạng Internet là hai hình thức chính, mỗi loại phục vụ các mục đích khác nhau trong việc kết nối và chia sẻ dữ liệu.

a) Mạng nội bộ

Mạng nội bộ (còn gọi là mạng LAN[1]), là hệ thống mạng được thiết kế để kết nối các thiết bị và tài nguyên trong một phạm vi hạn chế như văn phòng, toà nhà hoặc khu vực nhỏ (*Hình 2*). Mạng nội bộ thường được triển khai với công nghệ và giao thức mạng như Ethernet hoặc wifi. Mạng nội bộ cho phép các thiết bị trong cùng một hệ thống (chẳng hạn như máy tính, máy in, máy chủ và thiết bị lưu trữ) chia sẻ tài nguyên và dữ liệu với nhau.



Hình 2. Mạng LAN kết nối các thiết bị trong một khu vực giới hạn về địa lí

Kể tên một số ứng dụng của mạng nội bộ mà em biết.

[1] LAN: viết tắt của Local Area Network.

b) Mạng Internet

Mạng Internet là hệ thống toàn cầu các máy tính và các thiết bị kết nối với nhau dựa trên mô hình TCP/IP[2]. Mạng Internet cho phép truy cập và truyền tải thông tin, dữ liệu và các dịch vụ trực tuyến trên toàn thế giới. Internet cung cấp nền tảng cho nhiều dịch vụ trực tuyến như: email, trang web, mạng xã hội, dịch vụ lưu trữ đám mây, trò chơi trực tuyến, dịch vụ thương mại điện tử,… Mạng Internet trở thành một phần không thể thiếu của cuộc sống ngày nay, ảnh hưởng đến hầu hết các khía cạnh của xã hội và kinh tế. Theo số liệu thống kê của Liên minh viễn thông quốc tế (ITU[3]) đến năm 2023, ước tính có hơn 5,3 tỉ người dùng Internet toàn cầu, xấp xỉ 66% dân số thế giới tại thời điểm đó.

Mạng Internet không được quản lí bởi một tổ chức duy nhất. Tuy nhiên, có một số tổ chức có ảnh hưởng đối với quản lí và phát triển của Internet. Cụ thể, ở phạm vi quốc tế, có thể kể đến như: Tổ chức phi lợi nhuận quản lí hệ thống tên miền và các phân cấp tên miền (ICANN[4]), Tổ chức phát triển và tiêu chuẩn hoá các giao thức và tiêu chuẩn liên quan đến Internet (IETF[5]),… Ngoài ra, ở mỗi quốc gia còn có các tổ chức quản lí mạng Internet cấp quốc gia, ví dụ như ở Việt Nam là Bộ Thông tin và Truyền thông.

c) So sánh giữa mạng LAN và Internet

Mạng LAN và Internet có nhiều điểm khác biệt cơ bản về phạm vi địa lí, tốc độ, cấu trúc, tính sở hữu, tính bảo mật và tính ứng dụng. *Bảng 1* so sánh chi tiết giữa mạng LAN và mạng Internet.

Bảng 1. So sánh mạng LAN và mạng Internet

	Mạng LAN	Mạng Internet
Phạm vi địa lí	Có phạm vi nhỏ, thường chỉ trong một toà nhà, văn phòng.	Không có giới hạn về mặt địa lí.
Tốc độ	Có tốc độ truyền dữ liệu rất cao, từ 100 Mbps đến 10 Gbps hoặc hơn, tuỳ thuộc vào công nghệ và thiết bị.	Có tốc độ Internet thấp hơn, phụ thuộc vào nhiều yếu tố như hạ tầng mạng, lưu lượng truy cập và công nghệ kết nối.
Cấu trúc	Kết nối các thiết bị trong phạm vi nhỏ.	Là mạng lưới gồm nhiều mạng nội bộ kết nối với nhau.
Tính sở hữu	Có chủ sở hữu.	Không có chủ sở hữu.
Tính bảo mật	Dễ dàng kiểm soát và bảo mật.	Đòi hỏi nhiều biện pháp bảo mật phức tạp.
Tính ứng dụng	Chia sẻ tài nguyên như máy in, file, máy chủ nội bộ,...	Truy cập thông tin, thương mại điện tử, truyền thông xã hội, dịch vụ đám mây, email, video trực tuyến và chia sẻ tài nguyên.

[2] TCP/IP: viết tắt của Transmission Control Protocol/Internet Protocol.

[3] ITU: viết tắt của International Telecommunication Union.

[4] ICANN: viết tắt của The Internet Corporation for Assigned Names and Numbers.

[5] IETF: viết tắt của Internet Engineering Task Force.

- Mạng nội bộ là hệ thống kết nối các thiết bị trong một phạm vi nhỏ, như văn phòng hoặc toà nhà, cho phép chia sẻ tài nguyên và dữ liệu giữa các thiết bị như máy tính, máy in và server thông qua công nghệ Ethernet hoặc wifi.
- Mạng Internet là hệ thống toàn cầu kết nối các thiết bị, cho phép chia sẻ tài nguyên, thông tin, dữ liệu và truy cập các dịch vụ trực tuyến như email, trang web, mạng xã hội và thương mại điện tử.

2. Mạng Internet tác động đến cuộc sống

Mạng Internet có ảnh hưởng lớn đến cuộc sống thông qua nhiều ứng dụng khác nhau. Dưới đây là một số ứng dụng của Internet có tác động đến cuộc sống:

Truy cập thông tin: Internet cung cấp nguồn thông tin phong phú và đa dạng trên mọi chủ đề từ khắp nơi trên thế giới. Nhờ đó, người dùng có thể tra cứu thông tin, đọc báo, xem video,… ở bất kì đâu một cách dễ dàng. Ví dụ, ngư dân giờ đây có thể truy cập Internet tại các vùng biển đảo.

Giao tiếp và liên lạc: Internet cho phép người dùng giao tiếp và liên lạc với nhau một cách hiệu quả qua email, tin nhắn, cuộc gọi bằng âm thanh, hình ảnh thông qua các ứng dụng như Zalo, Viber, Messenger,…

Mạng xã hội: Các nền tảng mạng xã hội như Facebook, Instagram,... đã trở thành phương tiện quan trọng để kết nối và giao tiếp với bạn bè, người thân và cộng đồng trực tuyến.

Làm việc từ xa: Internet cho phép kết nối các doanh nghiệp, nhà tuyển dụng, chuyên viên, người lao động ở khắp mọi nơi trên thế giới. Nhờ đó, giúp người lao động có thể làm việc từ bất kì đâu với kết nối Internet. Chẳng hạn, các bác sĩ có thể tư vấn phẫu thuật trực tuyến nhờ vào Internet (*Hình 3*).

Hình 3. Các bác sĩ tại bệnh viện Hữu nghị Việt Đức đang tư vấn phẫu thuật trực tuyến[6]

[6] Nguồn: báo Điện tử Chính phủ.

Giáo dục và đào tạo: Internet mang lại cơ hội học tập không giới hạn cho mọi người thông qua các khoá học trực tuyến, bài giảng video và tài liệu học trực tuyến từ các trường học và tổ chức giáo dục trên toàn thế giới (*Hình 4*).

Hình 4. Học sinh học trực tuyến thông qua Internet

Thương mại điện tử: Internet mang đến cho người dùng khả năng mua sắm trực tuyến từ các cửa hàng online, sử dụng dịch vụ giao hàng nhanh, và truy cập vào các nền tảng thương mại điện tử phổ biến như eBay, Lazada, Tiki,... giúp việc mua sắm trở nên thuận tiện và linh hoạt hơn.

Giải trí: Internet cung cấp một loạt các dịch vụ giải trí như xem phim, nghe nhạc, đọc truyện tranh trực tuyến, chơi game trực tuyến và xem video trực tuyến,… giúp người dùng có thể thư giãn sau những giờ học tập, làm việc căng thẳng.

Internet mang lại cơ hội cho sự sáng tạo và phát triển cá nhân. Hãy chia sẻ một ví dụ về cách Internet đã giúp em phát triển kĩ năng sáng tạo hoặc khám phá sở thích mới. (Ví dụ: chơi nhạc cụ mới, học ngoại ngữ, học một môn thể thao mới,…).

Mạng Internet là cánh cửa mở ra vô vàn ứng dụng và trải nghiệm bất kể khoảng cách địa lí hay rào cản về ngôn ngữ. Người dùng có thể sử dụng mạng Internet để truy cập thông tin, giao tiếp và liên lạc, thực hiện công việc từ xa, học trực tuyến, mua bán trực tuyến,…

Chân trời sáng tạo

LUYỆN TẬP

1. Kể một số ứng dụng của mạng Internet mà em biết đã tác động đến cuộc sống của con người.
2. Thảo luận nhóm về tình huống: "Nếu một ngày không có Internet, cuộc sống của em sẽ bị ảnh hưởng như thế nào?".

VẬN DỤNG

Em hãy tìm hiểu và cho biết các mạng 4G, 5G có phải là mạng Internet không? Vì sao em nghĩ như vậy?

BÀI B2 INTERNET VẠN VẬT VÀ ĐIỆN TOÁN ĐÁM MÂY

MỤC TIÊU **Sau bài học này, em sẽ:**

- Nêu được một số dịch vụ cụ thể mà điện toán đám mây cung cấp cho người dùng.
- Nêu được khái niệm Internet vạn vật (IoT).
- Nêu được ví dụ cụ thể về thay đổi trong cuộc sống mà IoT đem lại. Phát biểu ý kiến cá nhân về ích lợi của IoT.

KHỞI ĐỘNG

Thu phí không dừng là hình thức thu phí tự động thông qua thẻ định danh RFID[1] dán trên kính lái hoặc đèn xe giúp xe qua trạm thu phí dễ dàng mà không cần dừng lại (*Hình 1*). Khi phương tiện di chuyển tới trạm, máy đọc sẽ quét chiếc thẻ này, từ đó tự động trừ tiền trong tài khoản giao thông của khách hàng. Hình thức thu phí này giúp các phương tiện rút ngắn thời gian chờ đợi khi qua trạm. Theo em, hệ thống này có những ưu điểm gì khi so với trạm thu phí truyền thống?

Hình 1. Làn thu phí không dừng – xu hướng của giao thông thông minh (Nguồn ảnh: vetc.com.vn)

KHÁM PHÁ

1. Internet vạn vật

Internet vạn vật là hệ thống mạng lưới kết nối bao gồm các phương tiện và vật dụng, các thiết bị thông minh được cài đặt các cảm biến, phần mềm chuyên dụng giúp chúng có thể tự động kết nối, thu thập và trao đổi dữ liệu trên cơ sở hạ tầng Internet mà không nhất thiết có sự tương tác trực tiếp giữa con người với con người hay con người với máy tính.

Internet vạn vật có thể ứng dụng trên nhiều lĩnh vực khác nhau, có thể được kể đến như nhà thông minh, nông nghiệp thông minh,…

Nhà thông minh: Internet vạn vật đang định hình một cách sống hiện đại và tiện ích, nơi mà mọi thiết bị trong ngôi nhà được kết nối và tương tác với nhau thông qua mạng Internet. Sự tích hợp của Internet vạn vật trong nhà thông minh không chỉ mang lại trải nghiệm tiện lợi mà còn an toàn và hiệu quả về mặt năng lượng. Trong nhà thông minh, các thiết bị âm thanh, ánh sáng, điều hoà nhiệt độ, giải trí, vô tuyến truyền hình,… được kết nối với một

[1] RFID (Radio Frequency Identification): công nghệ sử dụng sóng radio để nhận dạng và theo dõi các đối tượng thông qua thẻ gắn trên chúng.

ứng dụng điều khiển từ xa, cho phép người dùng kiểm soát mọi lúc, mọi nơi. Đồng thời, các thiết bị an ninh như camera, cảm biến chuyển động được tích hợp để tự động gửi cảnh báo và hình ảnh khi phát hiện hoạt động không mong muốn. Bên cạnh đó, thiết bị điều khiển nhiệt độ và ánh sáng được tích hợp với các cảm biến nhiệt độ và ánh sáng tự động,... để có thể tự động tuỳ chỉnh nhiệt độ, ánh sáng phù hợp với con người và nhiều tiện ích khác (*Hình 2*).

Hình 2. Nhà thông minh

Nông nghiệp thông minh: Khi áp dụng các ứng dụng của nông nghiệp thông minh, người nông dân có thể theo dõi đồng ruộng theo thời gian thực bằng cách sử dụng một tập hợp các cảm biến và thiết bị di động kết nối Internet. Dưới mặt đất, các cảm biến thu thập thông số về thuộc tính của đất, nồng độ thuốc trừ sâu, độ ẩm, độ pH,... Trên bầu trời, các máy bay không người lái (drone) giám sát liên tục và tìm kiếm các vùng cây trồng bị héo úa, nhiễm bệnh, có thay đổi bất thường (*Hình 3*). Từ những thông tin đó, các thiết bị tưới tiêu, bón phân tự động điều chỉnh dung lượng theo thời gian/khu vực, thậm chí các thiết bị này có thể tiến đến khu vực cần can thiệp một cách chính xác.

Hình 3. Máy bay không người lái được ứng dụng trong nông nghiệp thông minh

Em hãy tìm xung quanh lớp học những đồ vật nào nên được tích hợp khả năng kết nối với mạng Internet. Theo em, chúng sẽ tăng cường công dụng hay hỗ trợ con người tốt hơn như thế nào?

Internet vạn vật là hệ thống kết nối các thiết bị thông minh qua Internet, giúp chúng tự động thu thập và trao đổi dữ liệu mà không cần sự tương tác trực tiếp của con người, ứng dụng trong nhiều lĩnh vực như nhà thông minh và nông nghiệp thông minh.

2. Điện toán đám mây

Điện toán đám mây là một mô hình cho phép người dùng truy cập và sử dụng các tài nguyên, dịch vụ công nghệ thông qua Internet. Thay vì phải đầu tư vào hệ thống, nền tảng hay phần cứng riêng, mọi thứ đều được cung cấp dưới dạng "dịch vụ" từ các nhà cung cấp đám mây. Người dùng không cần phải hiểu sâu về công nghệ hay quản lí cơ sở hạ tầng, mà chỉ cần kết nối để sử dụng. Các dịch vụ này luôn sẵn sàng khi cần, với chi phí hợp lí, chất lượng cao và khả năng tương thích rộng rãi, mang lại sự tiện lợi và linh hoạt cho mọi đối tượng sử dụng.

Điện toán đám mây có ba mô hình phổ biến: SaaS, PaaS và IaaS.

SaaS (Software as a Service, Phần mềm như một dịch vụ) là loại dịch vụ điện toán đám mây dễ thấy nhất trong đời sống hằng ngày. Đây là các phần mềm mà người dùng có thể truy

cập trực tiếp qua Internet mà không cần phải cài đặt hoặc quản lí chúng. Tiêu biểu là các dịch vụ thư điện tử (Gmail, Hotmail,...), các dịch vụ tin học văn phòng (Microsoft Office 365, Google Workspace,…),…

PaaS (Platform as a Service, Nền tảng như một dịch vụ) là dịch vụ cung cấp môi trường cho lập trình viên phát triển và triển khai ứng dụng mà không phải lo về quản lí hạ tầng. Điều này giúp họ tập trung vào việc lập trình, trong khi hệ thống tự động xử lí các vấn đề như bảo trì và mở rộng. Ví dụ, em muốn xây dựng một trang web, em chỉ cần lập trình mã nguồn, nền tảng PaaS sẽ lo phần còn lại như cung cấp máy chủ, cơ sở dữ liệu và bảo mật. Một số dịch vụ PaaS phổ biến có thể kể đến như Google App Engine, Heroku, AWS Lambda, Google Maps,...

IaaS (Infrastructure as a Service, Cơ sở hạ tầng như một dịch vụ) cung cấp máy chủ và hạ tầng qua Internet. Với IaaS, thay vì phải mua máy chủ vật lí, người dùng có thể thuê các máy chủ ảo từ các nhà cung cấp và chỉ trả tiền cho những tài nguyên họ sử dụng. IaaS cho phép người dùng toàn quyền kiểm soát hệ điều hành, ứng dụng và dữ liệu của họ, nhưng không cần phải quản lí phần cứng vật lí. Một số ví dụ điển hình là Amazon Web Services, Microsoft Azure, Google Cloud và một số hạ tầng lưu trữ phổ biến là Google Drive, Microsoft OneDrive, AWS S3,... (*Hình 2*).

Hình 2a. Google Drive

Hình 2b. Microsoft OneDrive

Hình 2. Google Drive và Microsoft OneDrive là hai dịch vụ điện toán đám mây tiêu biểu

Hãy kể tên một số ứng dụng điện toán đám mây và nêu chức năng của nó mà em biết.

- Điện toán đám mây là công nghệ cho phép người dùng truy cập và sử dụng tài nguyên máy tính như lưu trữ, máy chủ và phần mềm thông qua Internet thay vì phải quản lí trực tiếp. Có ba mô hình chính là SaaS, PaaS và IaaS.
- Các cá nhân và doanh nghiệp có thể dễ dàng tiếp cận và sử dụng công nghệ một cách hiệu quả nhờ chi phí thấp và không cần kiến thức chuyên sâu.

LUYỆN TẬP

1. Em hãy phân tích lợi ích của giải pháp nhà thông minh.

2. Em hãy phân tích những lợi ích khi sử dụng dịch vụ điện toán đám mây.

VẬN DỤNG

Em hãy cùng với bạn tìm hiểu về ứng dụng của Internet vạn vật trong các lĩnh vực khác như chăm sóc và theo dõi sức khoẻ, xe tự hành, đô thị thông minh,...

BÀI B3 SỬ DỤNG MỘT SỐ DỊCH VỤ TIỆN ÍCH TRÊN MÁY TÍNH VÀ INTERNET

MỤC TIÊU **Sau bài học này, em sẽ:**

- Sử dụng được một số chức năng xử lí thông tin trên máy PC và thiết bị số, ví dụ dịch tự động văn bản hay tiếng nói.
- Khai thác được một số nguồn học liệu mở trên Internet.

Hình 1. Cuộc hội thoại giao tiếp với người nước ngoài

KHỞI ĐỘNG

Quan sát *Hình 1*, bạn Anna muốn hỏi bạn Hoa đường đi đến trạm xe buýt gần nhất nhưng bạn Anna chỉ nói được tiếng Anh, còn bạn Hoa chỉ nói được tiếng Việt. Em hãy gợi ý cách để hai bạn trao đổi được thông tin nhanh nhất.

THỰC HÀNH

Nhiệm vụ 1. Sử dụng công cụ Google Translate để dịch thuật

Google Translate là một công cụ dịch ngôn ngữ trực tuyến do Google phát triển, giúp người dùng dịch văn bản từ ngôn ngữ này sang ngôn ngữ khác một cách nhanh chóng và hiệu quả. Công cụ này đã trở thành một phần quan trọng trong cuộc sống hằng ngày của nhiều người trên toàn thế giới.

Yêu cầu: Truy cập Google Translate tại translate.google.com để:

a) Sử dụng Google Translate dịch từ tiếng Anh sang tiếng Việt.

b) Thực hiện dịch bằng giọng nói.

Hướng dẫn:

a) Các bước sử dụng Google Translate như sau:

1 Truy cập vào địa chỉ translate.google.com.

2 Lựa chọn loại dữ liệu cần dịch: văn bản, hình ảnh, tài liệu và website. Ở đây ta chọn văn bản.

3 Lựa chọn ngôn ngữ nguồn (tiếng Anh).

4 Lựa chọn ngôn ngữ đích (tiếng Việt).

5 Nhập văn bản cần dịch ở bên trái. Văn bản kết quả hiện ra ở bên phải.

Hình 2. Giao diện của công cụ Google Translate

b) Thực hiện nháy chọn biểu tượng 🎤 trên *Hình 2* để dịch bằng giọng nói.

Nhiệm vụ 2. Sử dụng nguồn học liệu mở từ Internet

Yêu cầu: Truy cập trang web igiaoduc.vn, em hãy tìm kiếm và đọc sách Toán 10 – Tập một (Chân trời sáng tạo) trực tuyến.

Hướng dẫn:

Truy cập vào sách giáo khoa trực tuyến trên Kho học liệu số igiaoduc.vn thông qua các bước như sau:

Hình 3a. Tìm và chọn học liệu muốn xem chi tiết nội dung

Hình 3b. Giao diện học liệu

Hình 3. Giao diện trang web Kho học liệu số igiaoduc.vn

LUYỆN TẬP

1. Em hãy sử dụng công cụ Google Tranlate để dịch câu "Tôi tự hào là người Việt Nam" sang năm ngôn ngữ khác nhau.
2. Em hãy tìm hiểu và nêu các bước dịch qua hình ảnh của công cụ Google Translate bằng thiết bị thông minh.
3. Hãy tìm hiểu chức năng dịch toàn bộ trang web của công cụ Google Translate và dịch một bài viết từ Wikipedia tiếng Anh (en.wikipedia.org) sang tiếng Việt.
4. Em hãy tìm hiểu và trình bày các bước sử dụng chức năng tìm kiếm ở ô tìm kiếm để tìm kiếm học liệu trên học liệu mở igiaoduc.vn.

VẬN DỤNG

Hãy truy cập kho học liệu mở quốc tế bằng cách đăng kí tài khoản tại trang Coursera (coursera.org) và đăng kí một khoá học.

Gợi ý: *Hình 4* mô tả một khoá học về Python miễn phí trên trang Coursera, em có thể đăng kí học bằng cách nháy chọn **Enroll for Free** để được học tập miễn phí.

Hình 4. Giao diện khoá học về Python do Google cung cấp trên trang Coursera

Em có biết

Thuật ngữ "Học liệu mở" thường được sử dụng để mô tả các tài nguyên giáo dục trực tuyến miễn phí mà các tổ chức giáo dục cung cấp cho cộng đồng không yêu cầu người học phải trả tiền. Các tài nguyên này bao gồm bài giảng, bài thuyết trình, bài thực hành, video và các tài liệu khác có thể giúp người học tự học và nắm bắt kiến thức mới.

Trên thế giới, các tổ chức giáo dục, đặc biệt là các trường đại học và tổ chức nghiên cứu, thường tạo ra các trang web để chia sẻ tài nguyên giáo dục của họ.

Ví dụ như kho học liệu của Viện Công nghệ Massachusetts (ocw.mit.edu), kho học liệu của Đại học Stanford (online.stanford.edu/free-courses),… hay các nền tảng miễn phí hoặc thu phí thấp như Coursera (coursera.org), edX (edx.org). Việt Nam cũng có một kho học liệu mở do Bộ Giáo dục & Đào tạo Việt Nam xây dựng mà em có thể truy cập thông qua đường dẫn igiaoduc.vn.

Chân trời sáng tạo

BÀI B4 SỬ DỤNG INTERNET AN TOÀN

MỤC TIÊU **Sau bài học này, em sẽ:**

- Nêu được những nguy cơ và tác hại nếu tham gia các hoạt động trên Internet một cách bất cẩn và thiếu hiểu biết. Trình bày được một số cách để phòng những tác hại đó.
- Biết cách tự bảo vệ dữ liệu của cá nhân.
- Nêu được một vài cách phòng vệ khi bị bắt nạt trên mạng.
- Trình bày được sơ lược về phần mềm độc hại. Sử dụng được một số công cụ thông dụng để ngăn ngừa và diệt phần mềm độc hại.

KHỞI ĐỘNG

Em có từng biết đến qua báo chí, truyền hình về trường hợp lừa đảo qua Internet nào không? Nếu có, em hãy kể lại cách thức lừa đảo đó. Theo em, chúng ta cần làm gì để tránh bị lừa đảo trong tình huống tương tự?

KHÁM PHÁ

1. Nguy cơ và tác hại khi tham gia các hoạt động trên Internet

a) Nguy cơ và tác hại

Internet mang đến vô số lợi ích cho người dùng, tuy nhiên, nó cũng tiềm ẩn nhiều nguy cơ có thể ảnh hưởng tiêu cực đến cuộc sống của chúng ta. Hiểu rõ những nguy cơ này là bước đầu tiên để bảo vệ bản thân và sử dụng Internet một cách an toàn và hiệu quả.

Nguy cơ về bảo mật thông tin: Kẻ xấu có thể đánh cắp thông tin cá nhân như tên, địa chỉ, số điện thoại, thông tin tài chính,… thông qua các trang web giả mạo, email lừa đảo, phần mềm độc hại.

Nguy cơ về nội dung độc hại:

– Nội dung bạo lực/khiêu dâm: Internet tạo điều kiện cho việc lan truyền nhanh chóng và dễ dàng các nội dung bạo lực, khiếm nhã, khiến người dùng vô tình tiếp xúc và bị ảnh hưởng tiêu cực về tâm lí, nhận thức, cũng như thuần phong mĩ tục.

– Tin tức giả mạo: Tin tức giả mạo có thể gây hiểu lầm, hoang mang cho dư luận và ảnh hưởng đến an ninh xã hội.

– Lời nói kích động thù địch: Lời nói kích động thù địch có thể dẫn đến phân biệt đối xử, bạo lực và xung đột giữa các nhóm người.

– Nguy cơ về sức khoẻ: Việc sử dụng Internet quá mức có thể dẫn đến nghiện Internet, gây ảnh hưởng đến sức khoẻ tinh thần, thể chất và các mối quan hệ xã hội. Việc nhìn

màn hình máy tính quá lâu có thể gây mỏi mắt, khô mắt và các vấn đề về thị lực. Một số tác hại khác của việc sử dụng Internet thời gian dài là rối loạn giấc ngủ, lười vận động,…

Nguy cơ về đạo đức:

– Lừa đảo trực tuyến: Kẻ xấu có thể sử dụng Internet để lừa đảo người khác, chiếm đoạt tài sản hoặc thông tin cá nhân.

– Bắt nạt trực tuyến: Bắt nạt trực tuyến có thể gây tổn thương tinh thần và ảnh hưởng đến cuộc sống của nạn nhân.

b) Một số biện pháp

Bảo vệ dữ liệu cá nhân:

– Cài đặt phần mềm diệt virus và chống phần mềm độc hại.

– Sử dụng mật khẩu mạnh (mật khẩu có độ dài ít nhất là 8 kí tự, có chữ viết hoa, chữ số và kí tự đặc biệt). Tránh sử dụng cùng một mật khẩu cho nhiều tài khoản khác nhau và không chia sẻ mật khẩu với bất kì ai.

– Không chia sẻ thông tin cá nhân nhạy cảm như địa chỉ nhà riêng, số điện thoại, số căn cước công dân,... trên Internet.

– Cảnh giác với các email, liên kết không rõ nguồn gốc.

– Cẩn thận khi đăng tải hình ảnh cá nhân lên mạng xã hội.

– Cài đặt quyền riêng tư khi sử dụng các nền tảng mạng xã hội để tránh người lạ biết được thông tin cá nhân.

Phòng chống tin giả:

– Đảm bảo rằng thông tin đến từ một nguồn chính thống, đáng tin cậy.

– Tạo thói quen kiểm chứng thông tin trước khi truy cập.

Giảm thiểu nguy cơ về sức khoẻ:

– Giới hạn thời gian sử dụng thiết bị điện tử và điện thoại thông minh mỗi ngày để tránh ảnh hưởng đến sức khoẻ.

– Áp dụng một số quy tắc bảo vệ sức khoẻ khi sử dụng thiết bị:

+ Thực hiện quy tắc "20-20-20": Nghỉ mắt 20 giây mỗi 20 phút, nhìn vào vật cách xa 20 feet (khoảng 6 mét) để bảo vệ mắt.

+ Sử dụng chế độ lọc ánh sáng xanh và hạn chế sử dụng thiết bị ít nhất một giờ trước khi đi ngủ để cải thiện chất lượng giấc ngủ.

+ Sau mỗi giờ sử dụng thiết bị nên thực hiện các bài tập giãn cơ ngắn để giảm căng thẳng cơ bắp.

1. Mật khẩu "yếu" là các mật khẩu có thể dễ dàng được đoán ra hoặc là những từ ngữ phổ biến, thông dụng. Em hãy trình bày những đặc điểm mà em nghĩ tạo nên một mật khẩu "mạnh" và có thể bảo vệ được thông tin cá nhân của em?

2. Theo em, có nên đặt một mật khẩu duy nhất cho tất cả mọi tài khoản dùng trên Internet không? Vì sao?

- Internet mang đến vô số lợi ích cho người dùng, tuy nhiên, bên cạnh những lợi ích đó là những nguy cơ tiềm ẩn mà người dùng cần lưu ý. Người dùng cần nâng cao nhận thức về các nguy cơ tiềm ẩn, thực hiện các biện pháp phòng ngừa như bảo mật thông tin, sử dụng Internet một cách hợp lí và an toàn.
- Không chia sẻ thông tin cá nhân nhạy cảm như địa chỉ nhà riêng, số điện thoại, số căn cước công dân,... trên Internet.

2. Phòng vệ khi bị bắt nạt qua Internet

Bắt nạt qua mạng (cyberbullying) là hành vi sử dụng công nghệ thông tin và truyền thông để lặp đi lặp lại những hành vi thù địch nhằm gây tổn thương hoặc quấy rối người khác trên các nền tảng mạng xã hội, tin nhắn, email, diễn đàn, trò chơi điện tử,... Theo tổ chức UNICEF[1], các hành vi có thể bao gồm: Lan truyền những lời nói dối hoặc đăng những bức ảnh đáng xấu hổ; Gửi tin nhắn hoặc đe doạ; Giả danh ai đó và lấy danh nghĩa của họ gửi những tin nhắn ác ý cho người khác (*Hình 1*).

Kẻ bắt nạt có thể dễ dàng che giấu danh tính, khiến nạn nhân khó xác định và tìm kiếm sự giúp đỡ. Thông tin lan truyền nhanh chóng trên Internet và có thể tiếp cận đến nhiều người, khiến nạn nhân cảm thấy xấu hổ và tổn thương hơn. Hậu quả là nạn nhân nếu nhẹ thì mất tập trung, sa sút kết quả học tập, nặng sẽ bị trầm cảm, lo âu, stress, cô lập, xa lánh bạn bè và gia đình.

Hình 1. Bắt nạt qua mạng để lại nhiều tổn thương về sức khoẻ và tinh thần

Một số biện pháp để tránh bị bắt nạt qua Internet:

– Không trả lời bất kì tin nhắn hoặc văn bản nào được gửi bởi những kẻ bắt nạt, đe doạ.

– Chặn tài khoản bắt nạt, báo cáo vi phạm. Tạm khoá tài khoản cá nhân để hạn chế đến mức thấp nhất việc tiếp nhận thông tin tiêu cực về bản thân.

– Lưu giữ bằng chứng về việc bắt nạt trên mạng phục vụ việc tố cáo hành vi xúc phạm, tấn công.

– Thông báo nhà trường, gia đình để có sự hỗ trợ từ mọi người. Nếu không có người thân xung quanh, báo với tổng đài Trẻ em 111 (xem *Hình 2*). Với những vụ việc có mức độ nghiêm trọng, báo cơ quan Công an gần nhất để xử lí theo quy định của pháp luật.

Hình 2. Tổng đài Quốc gia Bảo vệ Trẻ em 111 - nơi các em được bảo mật hoàn toàn thông tin

[1] UNICEF: United Nations International Children's Emergency Fund - Quỹ Nhi đồng Liên Hợp Quốc, là một quỹ cứu tế được thành lập bởi Liên Hợp Quốc, có mục đích bảo vệ và hỗ trợ quyền lợi của trẻ em trên toàn thế giới.

Em hãy nêu các biện pháp bảo vệ bản thân khi bị bắt nạt qua mạng.

Bắt nạt qua mạng là vấn đề nhức nhối cần được quan tâm và giải quyết. Nếu bị bắt nạt, hãy liên hệ với gia đình, nhà trường và cơ quan chức năng để được trợ giúp cần thiết.

3. Phần mềm độc hại

Phần mềm độc hại (malware) là những chương trình máy tính được thiết kế để tấn công, gây hại và thậm chí phá huỷ hệ thống máy tính. Chúng có thể ăn cắp dữ liệu cá nhân, làm hỏng tệp, thậm chí khiến máy tính không thể sử dụng được. Sau khi lây nhiễm vào máy tính, phần mềm độc hại có thể thực hiện nhiều hành vi khác nhau, bao gồm: tự sao chép và lan truyền, ẩn náu trong hệ thống, gửi thông tin nhạy cảm ra bên ngoài, làm hỏng hoặc mã hoá dữ liệu, tải và cài đặt các phần mềm độc hại khác,...

a) Các loại phần mềm độc hại khác nhau

Virus: là một loại phần mềm độc hại có khả năng tự chèn mã của nó vào các chương trình, tệp tin hoặc tài liệu khác. Khi virus đã xâm nhập vào vật chủ, nó sẽ ở trạng thái ẩn cho đến khi mã của nó được kích hoạt. Một số loại virus có thể sao chép mã của mình sang các tệp tin hoặc máy tính khác, từ đó lây lan nhanh chóng và gây thiệt hại trên diện rộng.

Sâu máy tính (Worm): là một loại phần mềm độc hại có khả năng di chuyển qua các kết nối Internet tìm kiếm mục tiêu tấn công. Sâu máy tính trở nên nguy hiểm vì chúng lợi dụng các lỗ hổng bảo mật trên máy tính, nên một khi đã xâm nhập, sâu máy tính rất khó bị ngăn chặn do khả năng di chuyển nhanh và lan rộng khi tìm kiếm các mục tiêu tiếp theo.

Trojan: Lấy cảm hứng từ câu chuyện "Ngựa gỗ thành Troy - thần thoại Hy Lạp", là một loại phần mềm độc hại được nguỵ trang thành một tệp tin bình thường. Hầu hết Trojan có mục đích kiểm soát máy tính của người dùng, đánh cắp dữ liệu và cài thêm phần mềm độc hại khác lên máy tính của nạn nhân.

Ransomware: Là một dạng phần mềm độc hại, hoạt động bằng cách mã hoá toàn bộ dữ liệu trên máy tính của nạn nhân để tống tiền. Nếu nạn nhân không trả cho kẻ tấn công một khoản tiền nhất định trong thời gian quy định, dữ liệu của họ sẽ bị mất vĩnh viễn.

Các loại phần mềm khác: Adware (phần mềm cố gắng chèn quảng cáo quá mức), Spyware (phần mềm gián điệp để theo dõi, thu thập thông tin cá nhân ra ngoài), Scareware (phần mềm hù doạ người dùng rằng máy tính của họ đã nhiễm virus, từ đó ép buộc họ mua/tải xuống phần mềm giả mạo), Cryptojacking (phần mềm chiếm quyền và buộc máy tính khai thác tiền ảo cho kẻ tấn công),...

b) Cách thức lây lan của các phần mềm độc hại

Phần mềm độc hại có thể lây lan qua nhiều cách khác nhau. Sau đây là một số phương thức lây lan phổ biến nhất của các phần mềm độc hại:

Khai thác lỗ hổng phần mềm: Các lỗ hổng này có thể tồn tại trong hệ điều hành, ứng dụng hoặc phần mềm khác. Khi phát hiện được lỗ hổng, kẻ tấn công có thể tạo ra phần mềm độc hại nhằm mục tiêu vào lỗ hổng đó để xâm nhập và thực hiện các hành vi độc hại.

Lừa đảo: Kẻ tấn công có thể sử dụng email giả mạo, tin nhắn SMS, trang web giả mạo hoặc các phương thức lừa đảo khác để đánh lừa người dùng nhấp vào các liên kết độc hại, tải xuống tệp đính kèm bị nhiễm phần mềm độc hại hoặc tiết lộ thông tin cá nhân nhạy cảm.

Qua USB và các thiết bị lưu trữ di động: Phần mềm độc hại có thể được lưu trữ trên USB và các thiết bị lưu trữ di động khác. Khi người dùng cắm thiết bị bị nhiễm phần mềm độc hại vào máy tính của mình, phần mềm độc hại có thể tự động cài đặt và lây nhiễm thiết bị.

Qua wifi công cộng: Mạng wifi công cộng thường không có bảo mật cao, do đó, kẻ tấn công có thể dễ dàng xâm nhập và theo dõi hoạt động của người dùng. Chúng có thể đánh cắp dữ liệu cá nhân nhạy cảm hoặc cài đặt phần mềm độc hại lên thiết bị của người dùng.

Qua quảng cáo độc hại: Quảng cáo độc hại có thể được hiển thị trên các trang web không uy tín hoặc trong các ứng dụng. Khi người dùng nhấp vào các quảng cáo này, họ có thể bị chuyển hướng đến các trang web bị nhiễm hoặc tải xuống phần mềm độc hại.

c) Cách phòng chống phần mềm độc hại

Phần mềm độc hại ngày càng trở nên tinh vi và nguy hiểm hơn, đe doạ dữ liệu và quyền riêng tư của người dùng. Do đó, việc trang bị cho bản thân những kiến thức và biện pháp phòng chống hiệu quả là vô cùng quan trọng. Dưới đây là một số cách thức hữu ích để bảo vệ thiết bị khỏi phần mềm độc hại:

– Cài đặt phần mềm diệt virus và chống phần mềm độc hại có uy tín.

– Cập nhật hệ điều hành và phần mềm thường xuyên.

– Sử dụng mật khẩu mạnh và bảo mật thông tin cá nhân.

– Sao lưu dữ liệu thường xuyên.

Đặc biệt, vì tính chất dễ dàng chia sẻ thông tin mà Internet trở thành một công cụ ưa thích của kẻ tấn công nhằm lan truyền các phần mềm độc hại. Do vậy, người dùng Internet cần luôn cảnh giác và thực hiện các biện pháp phòng ngừa để bảo vệ bản thân như:

Tránh truy cập các trang web không uy tín:

– Hạn chế truy cập các trang web có nội dung không rõ nguồn gốc, quảng cáo tràn lan hoặc có dấu hiệu lừa đảo.

– Kiểm tra kĩ địa chỉ trang web trước khi truy cập.

– Sử dụng các công cụ chặn quảng cáo để hạn chế tiếp xúc với các quảng cáo độc hại.

Cẩn thận khi truy cập vào các liên kết:

– Không nhấp vào các liên kết trong email, tin nhắn SMS hoặc trên mạng xã hội từ những người gửi không quen biết hoặc từ các nguồn không đáng tin cậy.

– Di chuột qua liên kết để xem địa chỉ đích trước khi truy cập.

Không tải xuống tệp từ các nguồn không xác định:

– Chỉ tải xuống tệp từ các trang web uy tín và có nguồn gốc rõ ràng.

– Cẩn thận với các tệp đính kèm trong email, tin nhắn SMS hoặc trên mạng xã hội từ những người gửi không quen biết.

– Nếu có thể, quét tệp bằng phần mềm diệt virus trước khi mở hoặc tải xuống.

Em hãy nêu các biện pháp phòng ngừa phần mềm độc hại để bảo vệ bản thân khi sử dụng Internet.

- Phần mềm độc hại là phần mềm được tạo ra nhằm mục đích gây hại cho máy tính hoặc mạng máy tính. Chúng có thể đánh cắp dữ liệu cá nhân, phá huỷ tệp tin, xâm hại quyền riêng tư, thậm chí tống tiền người dùng. Các loại phần mềm độc hại phổ biến bao gồm Virus, Worm, Trojan, Ransomware, Spyware, Adware,...
- Người dùng cần thực hiện các biện pháp phòng ngừa như cài đặt phần mềm diệt virus và chống phần mềm độc hại uy tín, cập nhật phần mềm thường xuyên, cẩn trọng khi truy cập Internet, sử dụng mật khẩu mạnh, bảo mật thông tin cá nhân, đồng thời nâng cao nhận thức về an ninh mạng.

LUYỆN TẬP

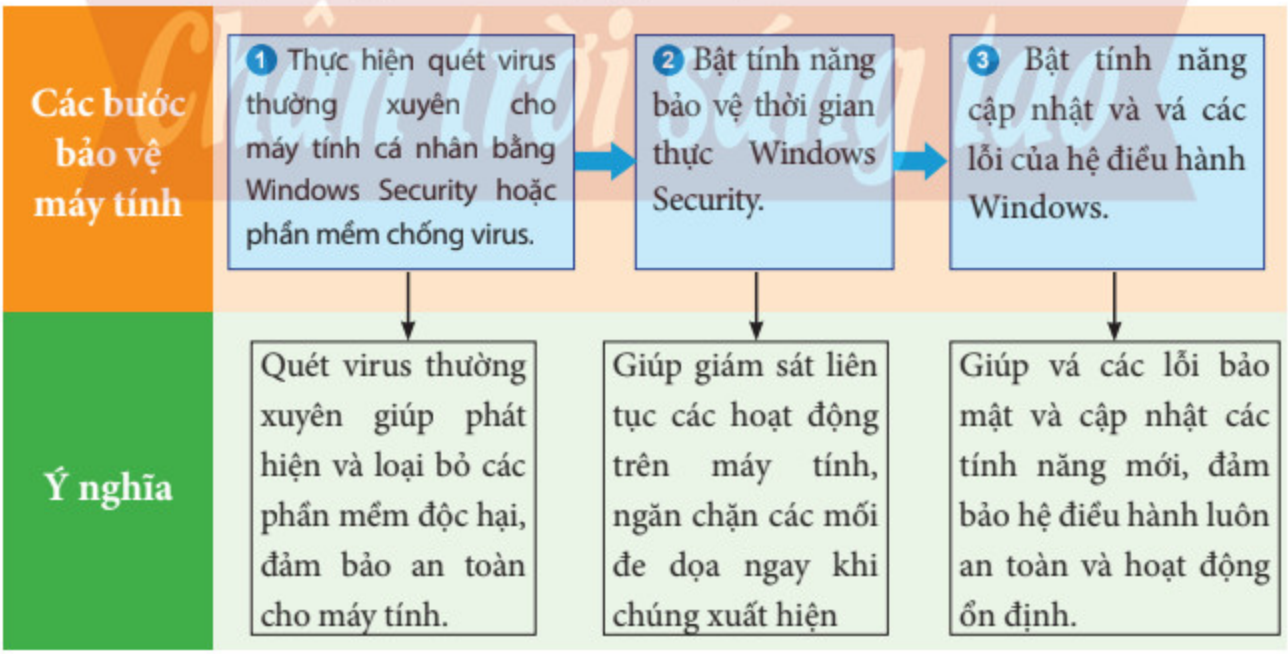
1. Hãy kể tên một vài phần mềm diệt virus mà em biết.

2. Theo em, nếu sử dụng điện thoại thông minh để truy cập các nguồn tài nguyên không uy tín, độc hại thì điện thoại thông minh có bị dính phần mềm độc hại không?

THỰC HÀNH

Nhiệm vụ. Bảo vệ máy tính cá nhân bằng phần mềm Windows Security

Để phát hiện và ngăn chặn các phần mềm độc hại, việc sử dụng phần mềm quét và bảo vệ máy tính là cần thiết; một số phần mềm phổ biến gồm BKAV, Kaspersky,... Trong bài thực hành này, em sẽ thực hiện các biện pháp bảo vệ máy tính cá nhân dùng hệ điều hành Windows 10 thông qua các bước lớn lần lượt là quét virus, bật tính năng bảo vệ thời gian thực và bật tính năng luôn luôn cập nhật của hệ điều hành Windows. Những bước này đảm bảo máy tính cá nhân của em phát hiện được các virus và nguy cơ mới, cũng như bảo vệ máy tính của em liên tục trong quá trình sử dụng (*Hình 3*).



Hình 3. Các bước để bảo vệ máy tính cá nhân

Yêu cầu: Thực hiện các bước bảo vệ máy tính cá nhân.

Hướng dẫn:

1 Thực hiện quét virus cho máy tính cá nhân bằng Windows Security

1a Vào mục **Settings** của hệ điều hành Windows và nháy chọn **Update & Security** (*Hình 4*). Cửa sổ **Update & Security** hiện ra như *Hình 5*.

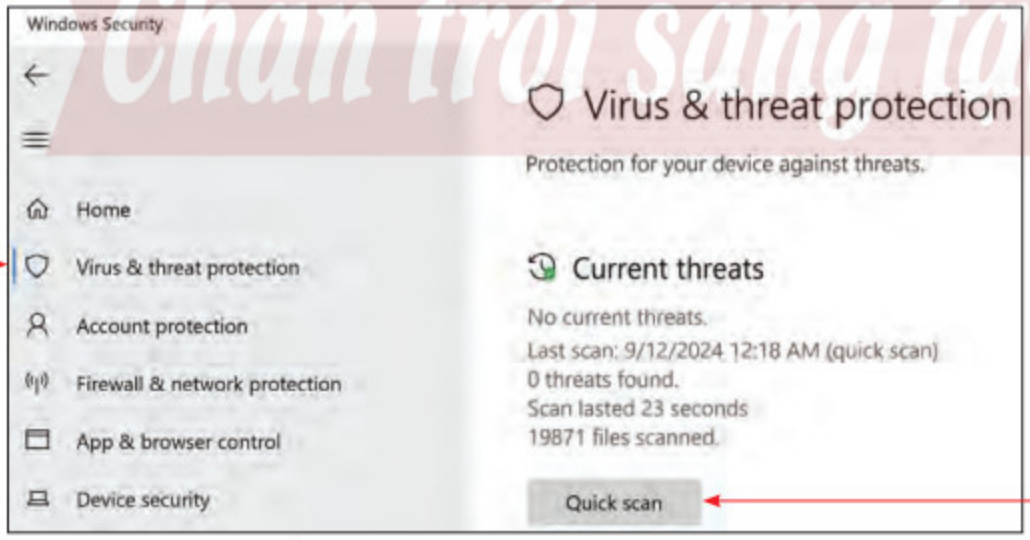
*Hình 4. Chức năng **Update & Security***

1b Tại cửa sổ **Update & Security**, ở giao diện bên trái, nháy chọn **Windows Security**.

1c Nháy chọn **Open Windows Security**. Cửa sổ **Windows Security** hiện ra như *Hình 6*.

*Hình 5. Chọn để mở giao diện của phần mềm **Windows Security***

1d Ở cửa sổ **Windows Security** vừa xuất hiện, chọn **Virus & threat protection**.

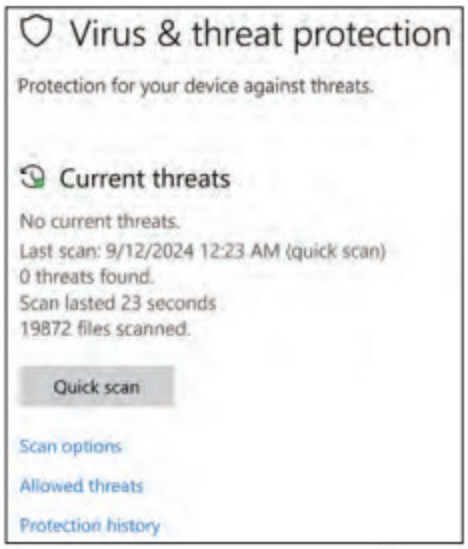


1e Chọn **Quick scan** để quét nhanh virus của máy tính cá nhân.

Hình 6. Chọn chức năng Quick scan (Quét nhanh)

Chân trời sáng tạo

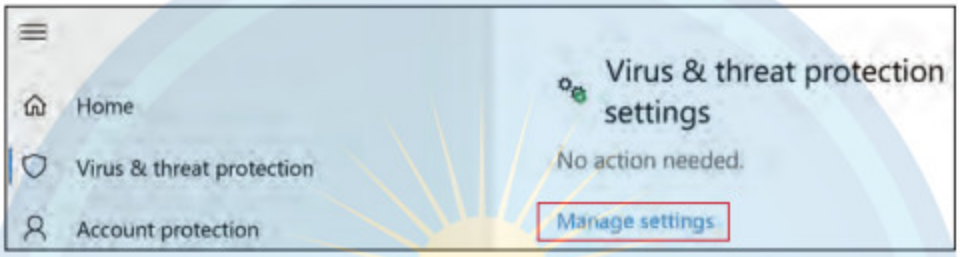
1g Chờ quá trình quét virus hoàn tất, phần mềm **Windows Security** sẽ hiển thị các mối nguy hại đã được hệ thống xử lí (*Hình 7*).



Hình 7. Windows Security hiển thị kết quả của lượt quét virus

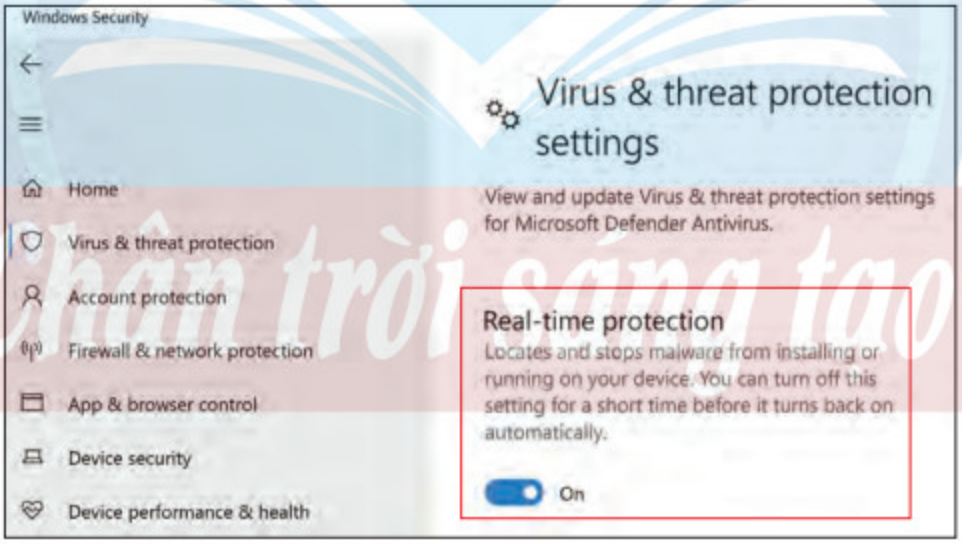
2 Bật tính năng bảo vệ thời gian thực của Windows Security

2a Thực hiện các bước từ 1a tới 1d. Tại mục **Virus & threat protection settings**, chọn vào đường dẫn **Manage settings** (*Hình 8*).



*Hình 8. Chọn tuỳ chỉnh cho mục **Virus & threat protection settings***

2b Bật chức năng **Real-time protection** (*Hình 9*).



*Hình 9. Bật tuỳ chỉnh **Real-time protection***

VẬN DỤNG

BKAV Home là phần mềm của người Việt giúp bảo vệ máy tính miễn phí, cung cấp các tính năng cơ bản như quét virus, bảo vệ thời gian thực và chống mã độc. Em hãy tìm hiểu và trình bày các bước thực hiện cài đặt phần mềm BKAV Home (https://www.bkav.com.vn/bkav-home) trên máy tính.

Chủ đề D

ĐẠO ĐỨC, PHÁP LUẬT VÀ VĂN HOÁ TRONG MÔI TRƯỜNG SỐ

BÀI D1 VẤN ĐỀ BẢN QUYỀN VÀ GIAO TIẾP TRONG MÔI TRƯỜNG SỐ

MỤC TIÊU **Sau bài học này, em sẽ:**

- Giải thích được một số khía cạnh pháp lí của vấn đề bản quyền, của việc sở hữu, sử dụng và trao đổi thông tin trong môi trường số. Nêu được ví dụ minh hoạ.
- Nêu được ví dụ minh hoạ sự vi phạm bản quyền thông tin và sản phẩm số, qua ví dụ đó giải thích được sự vi phạm đã diễn ra thế nào và có thể dẫn đến hậu quả gì.
- Nêu được ví dụ về những tác hại của việc chia sẻ và phổ biến thông tin một cách bất cẩn.
- Nêu được một vài biện pháp đơn giản và thông dụng để nâng cao tính an toàn và hợp pháp của việc chia sẻ thông tin trong môi trường số.
- Nêu được một số vấn đề nảy sinh về pháp luật, đạo đức, văn hoá khi việc giao tiếp qua mạng trở nên phổ biến.

KHỞI ĐỘNG

Bạn An sử dụng phần mềm xây dựng sơ đồ tư duy tóm tắt nội dung kiến thức đã học trong học kì I của môn Toán 10 và ghi tên mình là tác giả. Bạn An đăng tải sơ đồ lên tài khoản mạng xã hội để lưu lại và chia sẻ miễn phí cho mọi người. Bạn Bình tải sơ đồ về, xoá tên của bạn An và đăng tải lại lên tài khoản mạng xã hội của mình để tiếp tục chia sẻ miễn phí cho mọi người.

Theo em, bạn Bình làm như vậy có phù hợp hay không? Nếu em là bạn của bạn Bình và biết rõ nguồn gốc của sơ đồ tư duy đang được chia sẻ, em sẽ khuyên bạn Bình như thế nào?

KHÁM PHÁ

1. Vấn đề bản quyền trong môi trường số

Quyền tác giả là quyền hợp pháp mà một người sáng tạo có đối với tác phẩm do họ tạo ra hoặc sở hữu. Quyền tác giả tồn tại ngay từ thời điểm tác phẩm được tạo ra. Các cá nhân, tổ chức không được xâm phạm các quyền của tác giả (quản lí, sử dụng, khai thác

giá trị của tác phẩm) nếu chưa nhận được sự đồng ý của tác giả. Quyền tác giả bảo vệ các tác phẩm văn học, nghệ thuật và khoa học. Quyền tác giả được quy định theo Luật Sở hữu trí tuệ, số 50/2005/QH11, do Quốc hội ban hành ngày 29/11/2005, sửa đổi và bổ sung vào các năm 2009, 2019, 2022.

a) Quyền sở hữu trí tuệ và quyền tác giả

Căn cứ vào Điều 4 của Luật Sở hữu trí tuệ, quyền sở hữu trí tuệ là quyền của tổ chức, cá nhân đối với tài sản trí tuệ, bao gồm quyền tác giả và quyền liên quan đến quyền tác giả, quyền sở hữu công nghiệp và quyền đối với giống cây trồng.

Quyền tác giả là quyền của tổ chức, cá nhân đối với tác phẩm do mình sáng tạo ra hoặc sở hữu, bao gồm cả những biến thể của tác phẩm (bản phái sinh). Ví dụ, quyền tác giả được bảo lưu khi một cuốn sách được dịch sang một ngôn ngữ khác, một phần mềm máy tính được điều chỉnh lại để có thể sử dụng trên điện thoại di động,…

Quyền tác giả bao gồm quyền nhân thân (đặt tên cho tác phẩm; được đứng tên và được nêu tên thật hoặc bút danh khi công bố tác phẩm; được bảo vệ sự toàn vẹn của tác phẩm, không cho cá nhân hoặc tổ chức khác sửa chữa, cắt xén hoặc xuyên tạc dưới bất kì hình thức nào gây phương hại đến danh dự và uy tín của tác giả,…) và quyền tài sản (làm tác phẩm phái sinh; truyền đạt tác phẩm đến công chúng; cho thuê bản gốc hoặc bản sao;…).

Như vậy, trong KHỞI ĐỘNG, sơ đồ tư duy do bạn An xây dựng hoàn toàn nên bạn An là tác giả và là chủ sở hữu quyền tác giả đối với sơ đồ tư duy này. Tất cả các hành vi vi phạm quyền nhân thân (xoá tên tác giả, chỉnh sửa sơ đồ không thông qua ý kiến của tác giả) cũng như vi phạm quyền tài sản (chia sẻ lại tác phẩm trên mạng, làm tác phẩm phái sinh,...) đều là các hành vi xâm phạm quyền tác giả.

Bản quyền và quyền tác giả về bản chất là hai thuật ngữ không hoàn toàn tương đồng. Quyền tác giả bảo hộ các quyền nhân thân và quyền tài sản. Bản quyền ưu tiên bảo vệ quyền lợi đầu tư về kinh tế của người sở hữu quyền tác giả đối với tác phẩm (có thể không phải là người sáng tạo ra tác phẩm). Trong quá trình hội nhập quốc tế, hai thuật ngữ này cũng dần được đồng nhất. Các văn bản pháp luật của Việt Nam hiện nay sử dụng quyền tác giả làm thuật ngữ chính thức.

Căn cứ vào Điều 28 của Luật Sở hữu trí tuệ, vi phạm quyền tài sản hay quyền nhân thân đều là vi phạm quyền tác giả. Như vậy, các hành vi xâm phạm quyền tác giả, vi phạm bản quyền, rất đa dạng và có thể liệt kê một số hành vi tiêu biểu như sau: công bố, phân phối tác phẩm mà không được sự cho phép của tác giả; mạo danh tác giả; sao chép tác phẩm mà không được sự cho phép của tác giả hoặc chủ sở hữu quyền tác giả; nhân bản, sản xuất bản sao, phân phối, trưng bày hoặc truyền đạt tác phẩm đến công chúng qua mạng truyền thông và các phương tiện kĩ thuật số mà không được sự cho phép của chủ sở hữu quyền tác giả; cố ý huỷ bỏ hoặc vô hiệu hoá các biện pháp kĩ thuật do chủ sở hữu quyền tác giả thực hiện để bảo vệ quyền tác giả đối với tác phẩm;…

b) Bản quyền trong lĩnh vực Công nghệ thông tin

Căn cứ vào Điều 22 của Luật Sở hữu trí tuệ, chương trình máy tính là tập hợp các chỉ dẫn được thể hiện dưới dạng các lệnh, các mã,… khi gắn vào một phương tiện mà máy tính đọc được có khả năng làm cho máy tính thực hiện một công việc, đạt được một kết quả cụ thể. Chương trình máy tính được bảo hộ bản quyền tương tự như các tác phẩm văn học, dù được thể hiện dưới dạng mã nguồn hay mã máy.

Hành vi vi phạm bản quyền trong lĩnh vực Công nghệ thông tin bên cạnh những hành vi đã nêu ở mục a), còn bao gồm thêm các hành vi tiêu biểu như sau: giả mạo trang thông tin điện tử của cá nhân, tổ chức khác; tạo đường dẫn trái phép đối với các tên miền của tổ chức, cá nhân sử dụng hợp pháp tên miền đó; chiếm đoạt mã nguồn của phần mềm; tập hợp và chia sẻ miễn phí các phần mềm vi phạm bản quyền, phần mềm được phá khoá;…

Như vậy, đối với lĩnh vực Công nghệ thông tin, khi mua một phần mềm, cần phân biệt rõ giữa việc mua bản quyền với mua quyền sử dụng. Đối với phần mềm được mua bản quyền, người mua có quyền thực hiện các hoạt động kinh doanh đối với sản phẩm, có quyền điều chỉnh, thay đổi và bổ sung thêm các nội dung, tính năng mới. Đối với phần mềm được mua quyền sử dụng, bản quyền của phần mềm không thuộc về người mua, do đó người mua không có quyền thực hiện các hoạt động khác với quy định của quyền sử dụng do đơn vị phát hành quy định. Ví dụ, khi em mua hệ điều hành Windows, mua bộ phần mềm văn phòng Microsoft Office (Word, Excel, PowerPoint,…) đều chỉ là mua quyền sử dụng, được tính theo số máy được cài đặt. Nếu chỉ mua quyền sử dụng cho một máy, khi tiếp tục cài đặt phần mềm cho máy thứ hai là đang vi phạm bản quyền. Tuy nhiên, bên cạnh đó, người sử dụng hợp pháp phần mềm được bảo hộ có quyền sao chép phần mềm để lưu trữ dự phòng và thay thế phần mềm bị hỏng mà không cần phải xin phép hoặc trả thêm tiền bản quyền.

Trong lĩnh vực Công nghệ thông tin, với đặc điểm dễ sao chép và dễ phát tán với quy mô lớn, các hành vi vi phạm bản quyền đối với sản phẩm số sẽ gây thiệt hại rất lớn cho đơn vị sản xuất và phát hành, gây ảnh hưởng trực tiếp đến khả năng tái đầu tư phát triển sản phẩm của chủ sở hữu sản phẩm số, hạn chế sự phát triển của tập thể nghiên cứu ứng dụng công nghệ thông tin.

1. Trong KHỞI ĐỘNG, hành vi của bạn Bình (tải sơ đồ về, xoá tên bạn An và đăng tải lại lên tài khoản mạng xã hội dưới tên của mình để tiếp tục chia sẻ miễn phí) đã vi phạm những nội dung nào của quyền tác giả theo Điều 28 của Luật Sở hữu trí tuệ?

2. Các hành vi vi phạm bản quyền được quy định mức xử phạt vi phạm hành chính một cách chi tiết từ Điều 4 đến Điều 35 trong Nghị định số 131/2013/NĐ-CP, ban hành ngày 16/10/2013. Hãy tìm hiểu Nghị định này và trình bày các mức xử phạt vi phạm hành chính đối với hành vi vận chuyển, tàng trữ hàng hoá sao chép lậu.

3. Bạn Châu là một học sinh đam mê công nghệ và thích tìm hiểu về các phần mềm miễn phí phục vụ cho việc học tập. Bạn Châu mua USB có dung lượng 16 GB với giá 80 000 đồng

dùng để lưu trữ tất cả các phần mềm miễn phí mà bạn tìm hiểu được và bán lại cho các bạn học sinh khác có nhu cầu tìm hiểu và sử dụng phần mềm với giá 100 000 đồng. Hành vi của bạn Châu có vi phạm Luật Sở hữu trí tuệ hay không? Bạn Châu cần điều chỉnh lại hành vi như thế nào để vừa có thể giúp đỡ được các bạn học khác trong việc tiếp cận các phần mềm học tập miễn phí, vừa không vi phạm pháp luật?

- Quyền tác giả là quyền của tổ chức, cá nhân đối với tác phẩm do mình sáng tạo ra hoặc sở hữu, bao gồm cả những biến thể của tác phẩm (bản phái sinh).
- Việc tự ý sử dụng các sản phẩm số khi chưa được sự cho phép của tác giả hoặc người sở hữu bản quyền là hành vi vi phạm pháp luật.

2. Chia sẻ thông tin trong môi trường số một cách an toàn và hợp pháp

Ngày nay, với sự phát triển của mạng xã hội, việc chia sẻ thông tin qua môi trường số ngày càng trở nên phổ biến vì nhanh chóng và tiện lợi. Tuy nhiên, việc chia sẻ và phổ biến thông tin một cách bất cẩn trên các mạng xã hội có thể gây ra một số tác hại nghiêm trọng như sau: lộ thông tin cá nhân lưu trữ trong máy tính; vi phạm pháp luật khi chia sẻ thông tin số;…

Thông tin cá nhân của mỗi người là một dữ liệu quan trọng và nhạy cảm. Thông tin cá nhân số (danh tính số) bao gồm thông tin cá nhân và các thông tin tài khoản mạng xã hội,… nếu bị kẻ xấu cố ý đánh cắp được sẽ sử dụng vào mục đích giả mạo danh tính, thực hiện các hành vi lừa đảo, đe doạ tống tiền, đánh cắp tài sản,… đối với bạn bè và người thân. Như vậy, khi chia sẻ thông tin trên mạng, bản thân mỗi người phải có ý thức tự bảo vệ danh tính số của mình, đồng thời cũng phải chú ý giữ gìn, không tuỳ tiện tiết lộ danh tính số của những người khác. Ngoài ra, mỗi người dùng cũng cần tự trang bị các kĩ năng cơ bản về bảo mật và an toàn trong môi trường số như sử dụng mật khẩu mạnh (bao gồm chữ thường, chữ hoa và kí tự đặc biệt), cài đặt một số phần mềm an ninh mạng phổ biến để phòng chống và phát hiện các nội dung giả mạo, nhận diện được những phần mềm đánh cắp thông tin cá nhân,…

Khi tham gia vào môi trường số, tất cả các thao tác tìm kiếm, đọc, nhận xét và chia sẻ thông tin của mỗi người dùng đều được các máy chủ lưu trữ và tập hợp lại theo quy định của nhà quản lí mạng xã hội cũng như của các cơ quan chức năng. Tập hợp tất cả các thông tin về hoạt động của người dùng trong môi trường số ngày càng trở thành một dữ liệu quan trọng, giúp các nhà quản lí cơ quan, trường học có thêm cơ sở để đi đến quyết định tuyển dụng nhân sự, cấp học bổng học tập, hỗ trợ sinh hoạt,… Với việc truy cập vào các trang thông tin thiếu kiểm duyệt, đọc và chia sẻ những thông tin chưa chính xác, chưa qua kiểm duyệt và cũng có thể là sai sự thật, người dùng không những đang trực tiếp tạo ảnh hưởng xấu lên

hồ sơ trực tuyến của mình, mà còn có thể đang vô tình thực hiện các hành vi vi phạm pháp luật về việc lan truyền các nội dung sai trái, tiếp tay cho các phần tử xấu chống phá nhà nước Việt Nam.

Như vậy, để chia sẻ thông tin trong môi trường số được an toàn và hợp pháp, mỗi người dùng cần lưu ý xác định những nội dung sau: lí do và mục đích của việc chia sẻ thông tin; độ tin cậy của thông tin sẽ được chia sẻ, đảm bảo thông tin không thuộc dạng bị cấm chia sẻ, phổ biến như kích động bạo lực, cổ xuý hủ tục mê tín dị đoan,…; việc chia sẻ thông tin có vi phạm quyền sở hữu trí tuệ hay không; sự phù hợp của thông tin với đối tượng được chia sẻ; khả năng việc chia sẻ thông tin có thể ảnh hưởng đến hồ sơ trực tuyến của cá nhân và những người có liên quan như thế nào;…

1. Việc chia sẻ các dữ liệu sau đây sẽ có ảnh hưởng như thế nào đến hồ sơ trực tuyến của mỗi cá nhân?

a) Các thành tích, giải thưởng đã đạt được trong năm học vừa qua.

b) Giả mạo thông báo của nhà trường về việc cho học sinh nghỉ học.

c) Hình ảnh học sinh tham gia các hoạt động thiện nguyện giúp đỡ người có hoàn cảnh khó khăn.

d) Những nhận xét tiêu cực về các hoạt động ngoại khoá của nhà trường.

e) Thành tích cá nhân về việc phát hiện, tấn công các lỗ hổng bảo mật trên website của nhà trường và các cơ quan, tổ chức khác.

2. Lớp em có chụp hình về việc tham gia hoạt động thiện nguyện cung cấp thực phẩm và nấu bữa trưa cho trẻ em dân tộc miền núi. Để việc chia sẻ các hình ảnh này được an toàn và hợp pháp trên mạng xã hội, em và các bạn cần xem xét và đảm bảo những yếu tố nào?

Em cần có ý thức và hành động đảm bảo tính an toàn và hợp pháp khi chia sẻ thông tin trong môi trường số; không tuỳ tiện cung cấp thông tin cá nhân của bản thân hay người khác cho các cá nhân, tổ chức không đáng tin cậy.

3. Một số vấn đề về đạo đức, văn hoá và pháp luật khi giao tiếp qua mạng trở nên phổ biến

Giao tiếp qua mạng đã và đang trở thành một phần không thể thiếu trong cuộc sống hiện đại. Song hành cùng với sự nhanh chóng và tiện lợi mà nó mang lại, việc giao tiếp qua mạng cũng làm nảy sinh nhiều vấn đề về đạo đức, văn hoá và pháp luật cần có quan tâm đúng mức.

Về đạo đức: Với khả năng ẩn danh trên mạng xã hội, một số người dùng có thể lợi dụng chức năng này để hành động một cách thiếu trách nhiệm, chia sẻ thông tin chưa qua kiểm chứng, xúc phạm và bôi nhọ cá nhân và tổ chức khác. Điều này vi phạm nghiêm trọng quyền riêng tư của mỗi người dùng, tạo ra những luồng thông tin tiêu cực cho cả cộng đồng trực tuyến.

Về văn hoá: Mỗi quốc gia, mỗi vùng miền đều có những quy định về văn hoá riêng. Cho dù là cùng thuộc lãnh thổ Việt Nam nhưng giữa các khu vực dân cư khác nhau cũng có những quy định văn hoá khác nhau. Khi giao tiếp trong không gian số, người dùng cần tôn trọng và tuân thủ các quy định về văn hoá này, tránh gây ra những hiểu lầm và xung động không cần thiết.

Vấn đề về pháp luật: Giao tiếp qua mạng cũng chịu sự quản lí và quy định của các văn bản pháp luật. Với tốc độ xử lí nhanh, khả năng phát tán rộng khắp, các hành vi vi phạm bản quyền, mạo danh người khác, phát tán thông tin sai lệch,… đều có khả năng gây hậu quả nghiêm trọng và bị xử lí theo pháp luật.

Một số hành vi không phù hợp khi giao tiếp trên mạng có thể kể đến như: chia sẻ thông tin không phù hợp; công bố thông tin cá nhân hay tổ chức mà không nhận được sự cho phép; gửi thư rác, tin nhắn rác; vi phạm bản quyền khi sử dụng dữ liệu và phần mềm; bắt nạt qua mạng; lừa đảo qua mạng; ứng xử thiếu văn hoá;…

Để giải quyết những vấn đề trên, người dùng cần nâng cao nhận thức về văn hoá, đạo đức và pháp luật khi giao tiếp qua mạng, tôn trọng quyền riêng tư của người dùng khác và tuân thủ các quy định của pháp luật. Hãy nhớ rằng, hành động của mỗi người dùng trên mạng đều có thể ảnh hưởng đến những người dùng khác cũng như ảnh hưởng đến cả cộng đồng trực tuyến.

1. Em hãy nêu những vấn đề về đạo đức, văn hoá và pháp luật có thể xuất hiện khi tham gia vào các hoạt động sau trên mạng xã hội:

a) Tham gia nhóm giải bài tập Toán Trung học phổ thông.

b) Tham gia nhóm hoạt động thiện nguyện, chia sẻ thông tin quyên góp quần áo, sách vở ủng hộ các hoàn cảnh khó khăn trong xã hội.

c) Tham gia nhóm chia sẻ các phần mềm giáo dục.

2. Em hãy thảo luận với bạn để đưa ra một số ví dụ về hành vi vi phạm đạo đức, văn hoá, pháp luật khi giao tiếp qua mạng.

Việc ứng xử văn minh, lịch sự trên mạng không chỉ thể hiện văn hoá, đạo đức của mỗi người, mỗi cộng đồng, mà còn đảm bảo việc chấp hành các quy định của pháp luật khi giao tiếp trong không gian số.

Chân trời sáng tạo

LUYỆN TẬP

1. Thảo luận với bạn để nhận xét xem các hành vi sau đây có vi phạm pháp luật hay không? Tại sao?

   a) Bạn Duy rất thích tìm hiểu về kĩ thuật lập trình phần mềm. Bạn đã lấy một phần mềm chỉnh sửa ảnh mã nguồn mở, lập trình thêm một số tính năng hữu ích và phát hành lại trên mạng Internet. Vì là phần mềm mã nguồn mở, bạn Duy không cần phải trả tiền bản quyền phần mềm gốc cũng như không phải chia sẻ lợi nhuận nhận được từ việc phát hành lại phần mềm sau khi chỉnh sửa.

   b) Bạn Cúc đã mua bản quyền phần mềm chỉnh sửa video phục vụ cho dự án riêng của mình. Bạn ấy được phép sao chép phần mềm đó để lưu trữ dự phòng để chủ động tự cài đặt lại khi phần mềm đang sử dụng bị hỏng.

   c) Bạn Huy dùng điện thoại chụp lại nội dung sách tham khảo của bạn Dũng để sử dụng cho việc ôn tập thi học kì I của bản thân.

2. Bạn Hoa là một nữ sinh được nhiều bạn nam quan tâm vì học giỏi, hạnh kiểm tốt và xinh xắn. Bạn Lan là một nữ sinh ngồi cùng bàn với bạn Hoa. Một hôm, bạn Lan vô tình chụp được bức ảnh bạn Hoa ngủ gật trong lớp và đã gửi cho bạn Hoa xem. Bạn Hoa có yêu cầu bạn Lan đừng chia sẻ ảnh cho người khác xem vì sợ làm mất hình tượng của mình. Tuy nhiên, bạn Lan vẫn đăng tải bức ảnh đó lên mạng xã hội.

   a) Theo em, hành vi của bạn Lan có vi phạm pháp luật không? Tại sao?

   b) Trong trường hợp bạn Hoa không có ý kiến gì về việc làm của bạn Lan thì hành vi của bạn Lan có được xét là vi phạm pháp luật không? Tại sao?

VẬN DỤNG

1. Bạn Hùng là học sinh giỏi môn Toán và rất nhiệt tình hỗ trợ các bạn học sinh khác trong việc giải các bài tập Toán hay, khó được tập hợp từ các thầy cô trong trường cũng như các trường khác. Trong một lần, lời giải của bạn Hùng có chút nhầm lẫn. Bạn Minh phát hiện ra nội dung nhầm lẫn này và đã tự tìm tòi ra được cách khắc phục nội dung nhầm lẫn của bạn Hùng. Bạn Minh cảm thấy cực kì tự hào và đã nhanh chóng chụp hình hai lời giải của mình và của bạn Hùng, đăng lên mạng xã hội để so sánh. Theo em, hành vi của bạn Minh là nên hay không nên? Việc làm này có vi phạm pháp luật hay không? Nếu không vi phạm pháp luật thì tại sao bạn Minh vẫn không nên thực hiện hành vi trên?
2. Hãy nêu một ví dụ thực tế về việc vi phạm bản quyền trong môi trường số mà em biết và giải thích vì sao đó là hành vi vi phạm pháp luật?
3. Bạn Khoa sưu tầm các phần mềm được chia sẻ trên mạng Internet và đăng tải nội dung liên kết đến các trang chia sẻ phần mềm tại trang cá nhân Facebook của mình để dễ tìm kiếm, dễ chia sẻ cho các bạn học khác. Hành vi của bạn Khoa có vi phạm pháp luật hay không? Em hãy chỉ ra những nguy cơ mà bạn Khoa có thể gặp phải và đề xuất những điều chỉnh cần thiết.

BÀI D2 LUẬT CÔNG NGHỆ THÔNG TIN VÀ LUẬT AN NINH MẠNG

MỤC TIÊU **Sau bài học này, em sẽ:**

- Trình bày và giải thích được một số nội dung cơ bản của Luật Công nghệ thông tin; Nghị định về quản lí, cung cấp, sử dụng các sản phẩm và dịch vụ công nghệ thông tin; Luật An ninh mạng. Nêu được ví dụ minh hoạ.
- Vận dụng được luật và nghị định nêu trên để giải thích được tính hợp pháp của một hành vi nào đó trong lĩnh vực quản lí, cung cấp, sử dụng các sản phẩm và dịch vụ công nghệ thông tin.

KHỞI ĐỘNG

Hai bạn An và Bình chơi thân với nhau và thường xuyên chia sẻ thông tin với nhau. Do có một sự hiểu lầm nên hai bạn An và Bình không còn thân thiết với nhau nữa. Vì giận bạn Bình, bạn An đăng tải một số thông tin riêng của bạn Bình lên tài khoản mạng xã hội của mình để trêu chọc, làm xấu hình ảnh của bạn Bình đối với các bạn khác trong lớp. Em có suy nghĩ gì về hành động của bạn An? Hành vi chia sẻ thông tin của bạn An có phải là một hành vi vi phạm pháp luật hay không?

KHÁM PHÁ

Chân trời sáng tạo

1. Các văn bản pháp luật liên quan đến công nghệ thông tin

Để quản lí, điều chỉnh hành vi của cá nhân và tổ chức hoạt động trong lĩnh vực Công nghệ thông tin, Quốc hội Việt Nam, Chính phủ, Bộ Thông tin và Truyền thông đã nghiên cứu, xây dựng, ban hành các văn bản pháp luật như: Luật Công nghệ thông tin (2006); Luật An ninh mạng (2018); Nghị định về quản lí, cung cấp, sử dụng Internet và thông tin trên mạng (2013); Bộ quy tắc ứng xử trên mạng xã hội (2021),… (xem Phụ lục 1).

Các bộ luật, nghị định và quyết định nêu trên quy định rõ trách nhiệm trong hoạt động cung cấp các sản phẩm và dịch vụ công nghệ thông tin của tổ chức, cá nhân và trách nhiệm quản lí của các cơ quan nhà nước, tạo hành lang pháp lí an toàn, đảm bảo cho các hoạt động ứng dụng và phát triển sản phẩm công nghệ thông tin của cá nhân và đơn vị. Riêng đối với người dùng cá nhân, các văn bản trên tập trung vào các quy định về việc sử dụng không gian mạng và bảo vệ bản quyền của các sản phẩm số.

Căn cứ vào Luật Công nghệ thông tin, tổ chức và cá nhân hoạt động ứng dụng và phát triển công nghệ thông tin tại Việt Nam có các quyền lợi và nghĩa vụ như *Hình 1*.

Quyền lợi

- Được thiết lập trang thông tin điện tử theo quy định của pháp luật (Điều 23).
- Được Nhà nước khuyến khích tham gia nghiên cứu – phát triển công nghệ, sản phẩm công nghệ thông tin (Điều 38, 49, 40, 41, 74).
- Có quyền truy nhập và sử dụng thông tin trong cơ sở dữ liệu quốc gia, bộ, ngành, địa phương (Điều 58, 59).
- Được bảo vệ quyền, lợi ích hợp pháp khi sử dụng sản phẩm, dịch vụ công nghệ thông tin theo quy định của pháp luật (Điều 67).

Nghĩa vụ

- Chịu trách nhiệm về nội dung thông tin số của mình trên môi trường mạng (Điều 9, 23).
- Không được sử dụng thông tin số để: Chống phá Nhà nước, kích động bạo lực, kích động dâm ô, mê tín dị đoan; Xúc phạm uy tín của tổ chức, danh dự, nhân phẩm, uy tín của công dân; Quảng cáo, tuyên truyền hàng hoá, dịch vụ thuộc danh mục cấm (Điều 12).
- Tuân thủ luật về chống thư rác: Không được che giấu tên của mình, giả mạo tên của tổ chức, cá nhân khác khi gửi thông tin trên môi trường mạng; Không được tiếp tục gửi thông tin quảng cáo đến người tiêu dùng nếu người đó thông báo không đồng ý nhận thông tin quảng cáo (Điều 70).
- Không được tạo ra, cài đặt, phát tán virus máy tính, phần mềm gây hại vào thiết bị số của người khác (Điều 71).
- Bảo đảm bí mật thông tin riêng hợp pháp của tổ chức, cá nhân trao đổi, truyền đưa, lưu trữ trên môi trường mạng (Điều 72).

Hình 1. Quyền lợi và trách nhiệm của tổ chức, cá nhân khi tham gia hoạt động quản lí, cung cấp, sử dụng các sản phẩm dịch vụ công nghệ thông tin

1. Trong tình huống ở KHỞI ĐỘNG, hành vi bạn An chia sẻ thông tin của bạn Bình có phải là một hành vi vi phạm pháp luật hay không? Nội dung vi phạm của bạn An được quy định cụ thể trong điều nào của Luật Công nghệ thông tin?

2. Hãy nhận xét tính đúng, sai của các nhận định sau.

a) Người dùng nào cũng có quyền xây dựng trang thông tin điện tử để đăng tải thông tin theo sở thích của cá nhân.

b) Mỗi học sinh đều có quyền xây dựng chương trình virus và chuyển đến các người dùng khác để kiểm tra chất lượng của chương trình virus như một cách rèn luyện khả năng lập trình phần mềm.

c) Bạn Linh tạo tài khoản mạng xã hội với họ tên và ảnh đại diện là ảnh của bạn Mai, một bạn học sinh giỏi của lớp. Bạn Linh sưu tầm trên mạng và đăng tải lời giải bài tập của các môn học lên tài khoản xã hội trên với mục đích thu hút sự theo dõi của nhiều học sinh cũng như người dùng khác. Hành vi của bạn Linh đang thực hiện đã vi phạm pháp luật.

d) Bạn Khánh ghi hình cảnh các bạn học sinh ẩu đả với nhau và chuyển cho các bạn học sinh khác trong trường cùng xem. Vì bạn Khánh chỉ ghi hình và gửi clip, không trực tiếp tham gia vào cuộc ẩu đả nên không vi phạm pháp luật.

Khi tham gia vào hoạt động ứng dụng và phát triển công nghệ thông tin tại Việt Nam, các tổ chức và cá nhân đều được hưởng quyền lợi và phải thực hiện các nghĩa vụ quy định trong những văn bản pháp luật như Luật Công nghệ thông tin, Luật An toàn thông tin mạng, Luật An ninh mạng,…

2. Một số quy định pháp luật liên quan đến người dùng trên không gian mạng

Được Quốc hội khoá 14 thông qua vào ngày 12/06/2018, Luật An ninh mạng quy định về hoạt động bảo vệ an ninh quốc gia và bảo đảm trật tự, an toàn xã hội trên không gian mạng; trách nhiệm của cơ quan, tổ chức, cá nhân có liên quan. Luật An ninh mạng, cùng với Luật Công nghệ thông tin, quy định một trong những trách nhiệm quan trọng nhất của người dùng trên không gian mạng là đảm bảo tính đúng đắn của thông tin cung cấp cũng như chia sẻ.

Cụ thể, khoản 2, Điều 12 của Luật Công nghệ thông tin quy định cấm cung cấp, trao đổi, lưu trữ, sử dụng thông tin số nhằm các mục đích sau: Chống Nhà nước Cộng hoà xã hội chủ nghĩa Việt Nam, phá hoại khối đoàn kết toàn dân; Kích động bạo lực, tuyên truyền chiến tranh xâm lược, gây hận thù giữa các dân tộc và nhân dân các nước, phá hoại thuần phong mĩ tục của dân tộc; Tiết lộ bí mật nhà nước, bí mật quân sự, an ninh, kinh tế, đối ngoại và những bí mật khác đã được pháp luật quy định; Xuyên tạc, vu khống, xúc phạm uy tín của tổ chức, danh dự, nhân phẩm, uy tín của công dân;…

Ngoài ra, khoản 1, Điều 8 của Luật An ninh mạng cấm sử dụng không gian mạng để thực hiện các hành vi như: Tổ chức, hoạt động, câu kết, xúi giục, mua chuộc, lừa gạt, lôi kéo, đào tạo và huấn luyện người chống Nhà nước Cộng hoà xã hội chủ nghĩa Việt Nam; Xuyên tạc lịch sử, phủ nhận thành tựu cách mạng, phá hoại khối đại đoàn kết dân tộc, xúc phạm

tôn giáo, phân biệt đối xử về giới, phân biệt chủng tộc; Thông tin sai sự thật gây hoang mang trong Nhân dân, gây thiệt hại cho hoạt động kinh tế - xã hội, gây khó khăn cho hoạt động của cơ quan nhà nước hoặc người thi hành công vụ, xâm phạm quyền và lợi ích hợp pháp của cơ quan, tổ chức, cá nhân khác;…

Những hành vi vi phạm pháp luật về đưa tin trên mạng xã hội còn được các cơ quan quản lí nhà nước cụ thể hoá kèm theo các mức phạt cụ thể trong khoản 1, Điều 101 của Nghị định số 15/2020/NĐ-CP, cụ thể như sau: Phạt tiền từ 20 000 000 đồng đến 30 000 000 đồng đối với hành vi tiết lộ thông tin thuộc danh mục bí mật nhà nước, bí mật đời tư của cá nhân và bí mật khác mà chưa đến mức truy cứu trách nhiệm hình sự; Phạt tiền từ 10 000 000 đồng đến 20 000 000 đồng đối với hành vi lợi dụng mạng xã hội để cung cấp, chia sẻ thông tin giả mạo, thông tin sai sự thật, xuyên tạc, vu khống, xúc phạm uy tín của cơ quan, tổ chức, danh dự, nhân phẩm của cá nhân;…

Với sự phát triển nhanh và phổ biến rộng khắp của mạng xã hội, ngày 17/06/2021, Bộ Thông tin Truyền thông đã ban hành Quyết định số 874/QĐ-BTTTT về Bộ quy tắc ứng xử trên mạng xã hội. Ngoài nội dung yêu cầu các tổ chức, cá nhân tuân thủ các quy định về cung cấp thông tin lên mạng xã hội đã được các văn bản luật quy định, Bộ quy tắc cung cấp thêm các khuyến cáo về việc tuân thủ các quy định của nhà cung cấp dịch vụ mạng xã hội; khuyến khích người dùng sử dụng tên thật; đăng tải và chia sẻ các thông tin tích cực, những tấm gương người tốt việc tốt; quảng bá hình ảnh đất nước, con người và văn hoá Việt Nam ra thế giới.

Là một người dùng mạng xã hội có trách nhiệm, chúng ta không chia sẻ các bài viết vi phạm pháp luật, tin giả, tin rác. Ngay cả trong trường hợp khi việc chia sẻ thông tin không vi phạm pháp luật, chúng ta cũng cần phải cân nhắc cả đến khía cạnh đạo đức, mức độ ảnh hưởng tiêu cực có thể có đối với những người liên quan đến thông tin được chia sẻ. Một thông tin vô hại với người này có thể mang lại một hiểm hoạ không lường trước được cho người khác.

Chân trời sáng tạo

1. Em tìm hiểu Nghị định số 15/2020/NĐ-CP và trao đổi với các bạn về những hành vi mà học sinh, người dùng mạng Internet, có thể đã vô tình vi phạm và mức quy định xử phạt tương ứng.

2. Hành vi chiếm quyền sử dụng tài khoản mạng xã hội của người khác để thực hiện hành vi lừa đảo, chiếm đoạt tài sản bạn bè, người thân của chủ tài khoản, đòi tiền chuộc của nạn nhân,… có phải là một hành vi vi phạm pháp luật không? Hành vi này đã vi phạm vào những điều khoản nào được quy định trong Luật An ninh mạng?

Khi tham gia trao đổi thông tin trong môi trường số, người dùng cần tự trang bị một số kiến thức cơ bản về những hành vi nên và không nên thực hiện, được quy định cụ thể trong Luật An ninh mạng, Bộ quy tắc ứng xử trên mạng xã hội,... để trở thành một người dùng mạng có trách nhiệm.

LUYỆN TẬP

1. Em hãy phân loại các hành vi sau dựa trên 5 tiêu chí (Vi phạm an ninh quốc gia; Tấn công mạng; Phát tán phần mềm độc hại; Vi phạm bản quyền và sở hữu trí tuệ; Phân biệt chủng tộc và xúc phạm danh dự, nhân phẩm):

a) Lan truyền trên mạng xã hội bức ảnh về nạn phân biệt chủng tộc.

b) Sao chép nội dung bài viết khoa học từ các trang web khi chưa nhận được sự đồng ý từ tác giả.

c) Phát tán virus nhằm đánh cắp dữ liệu tài chính.

d) Đăng tải thông tin bản đồ Việt Nam không đúng chủ quyền quốc gia.

e) Chia sẻ bài viết, video thể hiện không đúng đường lối, quan điểm của Đảng và nhà nước.

g) Giả mạo trang thông tin điện tử của các cá nhân khác để trục lợi.

h) Xâm nhập tài khoản mạng xã hội của cá nhân khác để thu lợi bất chính.

i) Phát trực tiếp sự kiện thể thao quốc tế, buổi hoà nhạc, chương trình truyền hình,... trên trang cá nhân.

k) Sử dụng quyền của lực lượng chuyên trách bảo vệ an ninh mạng để đánh cắp thông tin, xâm phạm bí mật nhà nước.

l) Gửi, chia sẻ các liên kết chứa virus để đánh cắp thông tin cá nhân.

2. Bạn Khôi và bạn Long là học sinh giỏi môn Tin học và ưa thích tìm hiểu các nội dung về bảo mật mạng. Trong một lần thử nghiệm, bạn Khôi đã phát hiện ra lỗi bảo mật trong website của trường. Bạn Khôi đã trao đổi với bạn Long để cùng nhau lên kế hoạch thay đổi thông tin trang chủ của nhà trường, hiển thị một hình ảnh vui mắt lên trang chủ nhân dịp chuẩn bị nghỉ Tết Dương lịch, sau đó sẽ hồi phục lại nguyên trạng như cũ, với mục đích trải nghiệm và thử thách khả năng của bản thân cả hai bạn Khôi và Long. Hành vi của cả hai bạn Khôi và Long có phải là một hành vi vi phạm pháp luật hay không? Nếu là hành vi vi phạm pháp luật, hành vi đó đã vi phạm vào những điều khoản nào đã được quy định trong Luật Công nghệ thông tin và Luật An ninh mạng?

Chân trời sáng tạo

3. Theo Luật Công nghệ Thông tin, trách nhiệm bảo vệ dữ liệu cá nhân thuộc về bên nào?

A. Nhà cung cấp dịch vụ công nghệ thông tin.

B. Người sử dụng dịch vụ.

C. Nhà cung cấp đường truyền mạng Internet.

D. Chính phủ.

VẬN DỤNG

1. Là học sinh, đồng thời cũng là một người dùng mạng, em có quyền và nghĩa vụ gì khi tham gia hoạt động quản lí, cung cấp, sử dụng các sản phẩm, dịch vụ công nghệ thông tin?

2. Em hãy quan sát các tình huống sau và cho biết ai thực hiện đúng và ai thực hiện chưa đúng quyền và nghĩa vụ khi tham gia hoạt động quản lí, cung cấp, sử dụng các sản phẩm, dịch vụ công nghệ thông tin. Tại sao?

a) Bạn Hoà tự mình lập được trang web giới thiệu sản phẩm thủ công truyền thống của gia đình, giới thiệu đến toàn bộ các bạn bè trên mạng xã hội, bước đầu tìm kiếm và mở rộng thị trường.

b) Bạn Chiến và bạn Kiệt đang bàn nhau thử thách trình độ lập trình và bảo mật dữ liệu bằng cách tạo ra chương trình đánh cắp thông tin trong máy tính của nhau.

c) Vì mong muốn nhiều người biết đến doanh nghiệp của mình nên chị Dung đã thu thập số điện thoại, địa chỉ thư điện tử của nhiều phụ huynh học sinh và gọi điện, gửi thư quảng cáo về doanh nghiệp.

d) Cô Liên là chủ một trang trại chăn nuôi. Bên cạnh việc thường xuyên trao đổi với cán bộ khuyến nông ở địa phương, cô còn đọc thêm thông tin trên cổng thông tin điện tử của Bộ Nông nghiệp và phát triển nông thôn, Cục Chăn nuôi.

3. Ba bạn Bảo, Minh, Toàn cùng nhau hợp tác để tạo ra một ứng dụng giải trí sử dụng trên điện thoại thông minh. Được sự đồng ý của cả ba bạn, bạn Bảo đã chia sẻ ứng dụng miễn phí trên tài khoản mạng xã hội của mình. Một thời gian sau, bạn Bảo nhận được thư điện tử từ đại diện một đơn vị phát triển phần mềm lớn trên thế giới, đề nghị mua lại bản quyền của ứng dụng để phát triển thành phiên bản thương mại hoá. Bạn Bảo đã đồng ý và nhận toàn bộ tiền bán bản quyền phần mềm, không thông báo cũng như chia lại tiền bán bản quyền phần mềm cho hai bạn Minh và Toàn.

Hành vi của bạn Bảo có phải là hành vi vi phạm pháp luật không? Nếu có, hành vi của bạn Bảo đã vi phạm vào nội dung quy định trong điều khoản của luật nào?

Chủ đề F

GIẢI QUYẾT VẤN ĐỀ VỚI SỰ TRỢ GIÚP CỦA MÁY TÍNH

BÀI F1 NGÔN NGỮ LẬP TRÌNH BẬC CAO VÀ MÔI TRƯỜNG LẬP TRÌNH PYTHON

MỤC TIÊU **Sau bài học này, em sẽ:**

- Hiểu được vì sao cần có ngôn ngữ lập trình bậc cao.
- Làm quen với môi trường lập trình Python.
- Biết tạo và thực thi một chương trình Python đơn giản.

KHỞI ĐỘNG

Quan sát 3 chương trình dưới đây cùng thực hiện một nhiệm vụ in ra màn hình dòng chữ "Hello World!". Em hãy cho biết chương trình nào dễ hiểu nhất?

Bảng 1. Minh hoạ chương trình in ra màn hình "Hello World!"

Ngôn ngữ máy

```
00000000:
01001000 01100101 01101100
01101100 01101111 00100000
01010111 01101111 01110010
01101100 01100100 00100001
```

Ngôn ngữ Python

```
print('Hello World!')
```

Hợp ngữ (Assembly) x86

```
.MODEL SMALL
.STACK 100H
.DATA
MASSAGE DB 'Hello World!$'
.CODE
MAIN PROC
    MOV AX, @DATA
    MOV DS, AX
    MOV AH, 09h
    MOV DX, OFFSET MASSAGE
    INT 21h
MAIN ENDP
END MAIN
```

Chân trời sáng tạo

KHÁM PHÁ

1. Ngôn ngữ lập trình bậc cao

Để điều khiển máy tính giải quyết các vấn đề, con người cần sử dụng tập hợp các chỉ dẫn để máy có thể hiểu và thực hiện, qua đó tạo ra một ngôn ngữ giao tiếp giữa người và máy.

Ngôn ngữ máy là ngôn ngữ duy nhất mà máy tính có thể trực tiếp hiểu và thực hiện được. Ngôn ngữ máy được viết thông qua các mã nhị phân hoặc mã thập lục phân (mã HEX), làm nền tảng cơ bản cho sự hoạt động của máy tính. Các bộ vi xử lí khác nhau có thể sử dụng những ngôn ngữ máy khác nhau.

Ngôn ngữ lập trình là một hệ thống các quy tắc và cú pháp được sử dụng để viết chương trình máy tính, là ngôn ngữ ở mức cao hơn so với ngôn ngữ máy, giúp con người đọc hiểu và dễ dàng xây dựng được các tập lệnh, các chương trình để điều khiển máy tính. Tuy nhiên, máy tính không hiểu được các chương trình viết bằng ngôn ngữ lập trình bậc cao. Vì vậy, các chương trình này phải được dịch sang ngôn ngữ máy nhờ các phần mềm chuyên dụng gọi là chương trình dịch.

Có rất nhiều ngôn ngữ lập trình khác nhau đang được sử dụng trên thế giới, phục vụ cho những ứng dụng và mục đích riêng. Với việc sử dụng cú pháp và từ khoá gần giống với ngôn ngữ tự nhiên của con người, như tiếng Anh, ngôn ngữ lập trình bậc cao cho phép lập trình viên biên soạn các chương trình dễ đọc, dễ chuyển giao, dễ cập nhật, mở rộng trên nhiều nền tảng phần cứng khác nhau.

Ngôn ngữ lập trình bậc cao là ngôn ngữ được sử dụng để xây dựng và phát triển ứng dụng web, trò chơi điện tử, ứng dụng di động,... Ví dụ, JavaScript, Python, Java, C#,… là những ngôn ngữ lập trình bậc cao thông dụng nhất hiện nay.

Theo em, trong 3 chương trình được minh hoạ ở KHỞI ĐỘNG, chương trình nào sẽ được máy tính hiểu trực tiếp?

- Ngôn ngữ lập trình là hệ thống các kí hiệu và quy tắc cú pháp được sử dụng để viết chương trình máy tính, được lập trình viên sử dụng để viết chương trình tạo ra các phần mềm.
- Ngôn ngữ lập trình bậc cao sử dụng cú pháp và từ khoá gần giống với ngôn ngữ tự nhiên của con người, giúp chương trình dễ đọc, dễ hiểu.

Chân trời sáng tạo

2. Môi trường lập trình Python

a) Giới thiệu về môi trường lập trình Python

Python là một ngôn ngữ lập trình bậc cao do Guido van Rossum người Hà Lan nghiên cứu phát triển. Python được thiết kế với mục tiêu tạo ra một ngôn ngữ lập trình đơn giản và dễ đọc, dễ viết giúp các nhà phát triển làm việc hiệu quả hơn. Python được sử dụng phổ biến trong nghiên cứu khoa học, giảng dạy, phát triển ứng dụng thực tế. Phiên bản đầu tiên của Python (phiên bản 0.9.0) được phát hành lần đầu năm 1991.

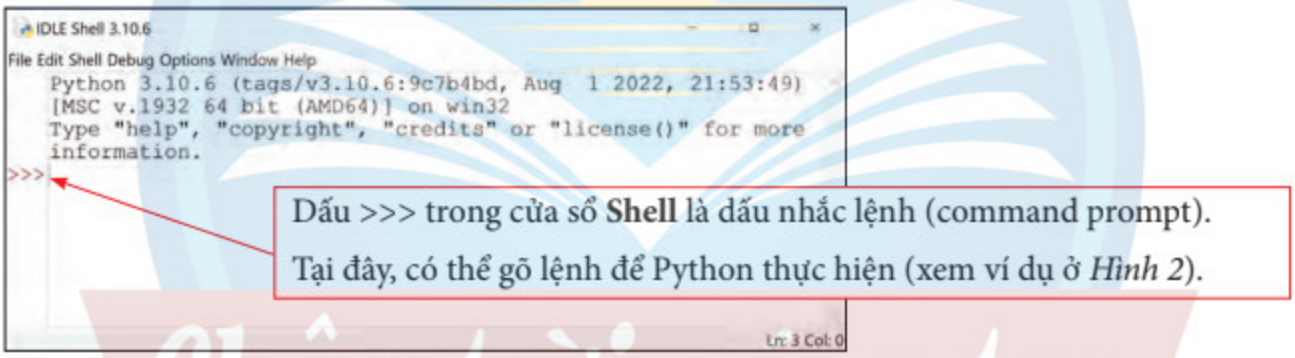
Hiện nay có các phần mềm (hoặc môi trường) để viết và chạy chương trình bằng ngôn ngữ Python như Python IDLE, Pycharm, Visual Studio, PyScripter,... Ngoài ra, cũng có thể sử dụng ứng dụng được tích hợp trên Google Drive là Google Colaboratory để viết chương trình. Nội dung cuốn sách này sẽ sử dụng phần mềm Python IDLE (phiên bản 3.10.6 64-bit) để minh hoạ cho các chương trình viết bằng Python. Biểu tượng của Python IDLE là .

Các cách khởi động phần mềm Python IDLE:

Cách 1: Nháy chọn biểu tượng **IDLE (Python 3.10.6)** trên màn hình Desktop.

Cách 2: Mở **Start**, chọn Python 3.10.6, chọn **IDLE (Python 3.10.6)**.

Sau khi khởi động, cửa sổ ứng dụng **IDLE Shell** của Python (sau đây gọi tắt là cửa sổ **Shell**) xuất hiện như *Hình 1*.



*Hình 1. Cửa sổ **Shell***

b) Viết và thực thi chương trình Python đơn giản

Phần mềm Python IDLE cho phép viết và thực thi chương trình theo một trong hai cách sau:

– Gõ từng dòng lệnh Python trong cửa sổ **Shell** và thực thi trực tiếp.

– Sử dụng trình soạn thảo của phần mềm Python IDLE để viết các dòng lệnh của chương trình, lưu vào tệp .py và thực thi chương trình.

Cửa sổ **Shell** cho phép gõ và thực thi từng dòng lệnh Python hoặc dùng trình soạn thảo Python để viết các lệnh của chương trình, lưu vào tệp .py và chạy chương trình Python.

Ví dụ 1: Tại cửa sổ **Shell**, ở dấu nhắc lệnh >>>, em gõ lệnh `1+2+3`, sau đó gõ phím **Enter**, màn hình Python sẽ in ra kết quả là 6. Tiếp theo, khi gõ lệnh `print('Hello World!')`, sau đó gõ phím **Enter**, màn hình Python sẽ in ra màn hình dòng chữ "Hello World!" (*Hình 2*).

```
>>> 1+2+3
    6
>>> print('Hello World!')
    Hello World!
```

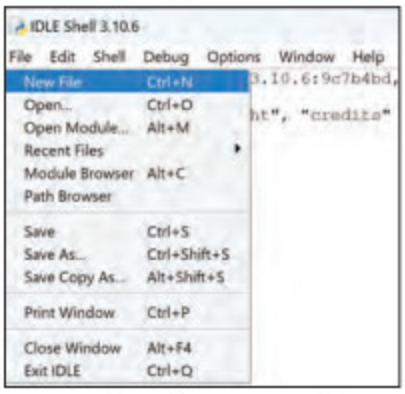
*Hình 2. Gõ lệnh trực tiếp tại cửa sổ **Shell** của Python*

Trong Python, lệnh `print()` có chức năng xuất các giá trị ra màn hình. Cú pháp đơn giản của lệnh `print()` như sau:

```
print(<gt1>,<gt2>,...,<gtN>)
```

với `<gt1>`, `<gt2>`,..., `<gtN>` là các giá trị cần in ra màn hình.

Ví dụ 2: Hướng dẫn các bước để viết, thực thi chương trình in ra màn hình dòng chữ "Hello World!" bằng trình soạn thảo Python.



*Hình 3. Chọn **New File***

❶ Mở cửa sổ soạn thảo chương trình

Tại cửa sổ **Shell**, nháy chọn **File**, chọn **New file** để mở cửa sổ soạn thảo chương trình như *Hình 3*.

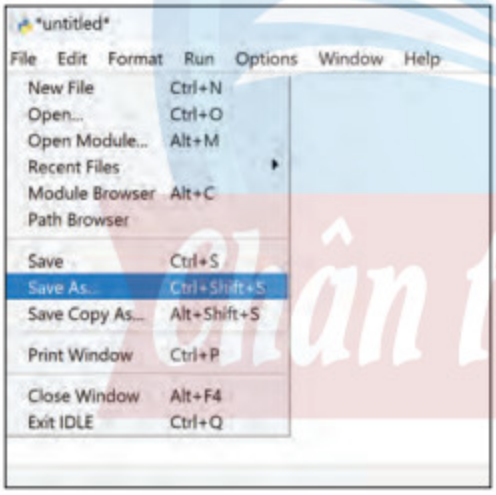
❷ Biên soạn chương trình

Tại vùng soạn thảo chương trình, gõ dòng lệnh của chương trình (*Hình 4*). Ví dụ: `print('Hello World!')`.

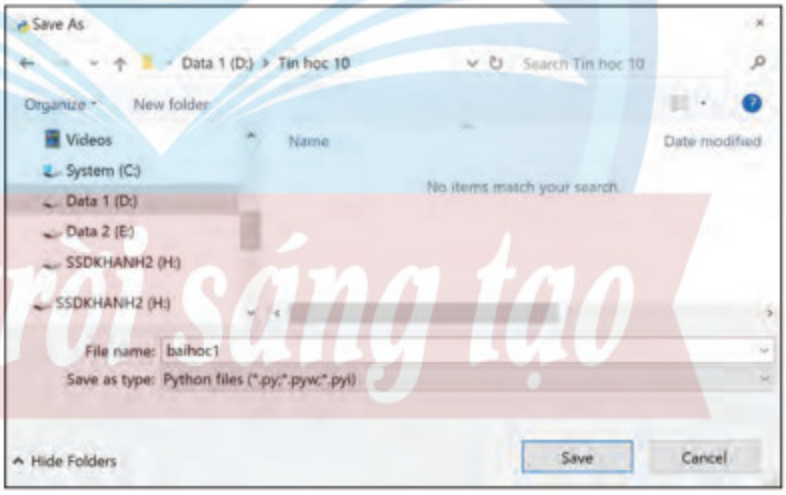
Hình 4. Soạn thảo câu lệnh

❸ Lưu tệp chương trình

Tại cửa sổ soạn thảo chương trình, vào **File**, chọn **Save As** (hoặc nhấn giữ tổ hợp phím **Ctrl + Shift + S**) để lưu tệp chương trình (*Hình 5a*), cửa sổ **Save As** hiện ra như *Hình 5b*. Tệp chương trình sẽ có phần mở rộng mặc định là .py.



*Hình 5a. Vào **File**, chọn **Save As***

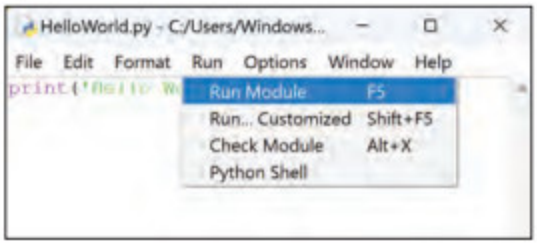


*Hình 5b. Cửa sổ **Save As***

Hình 5. Lưu chương trình Python

❹ Thực thi chương trình

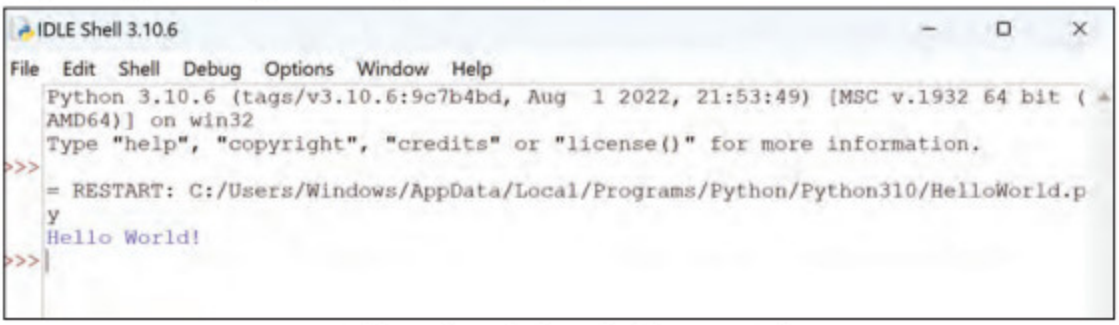
Tại cửa sổ soạn thảo chương trình, gõ phím **F5** hoặc vào **Run**, chọn **Run Module** (*Hình 6*) để thực thi chương trình (hoặc chạy chương trình).



Hình 6. Chạy chương trình

Chân trời sáng tạo

Kết quả chạy chương trình được hiển thị tại cửa sổ **Shell** như *Hình 7*.



Hình 7. Kết quả thực thi chương trình

Viết lệnh để in ra màn hình thông tin ngày tháng hiện tại. Ví dụ: "Hôm nay là thứ Hai, ngày 21 tháng 10, năm 2024".

- Python là ngôn ngữ lập trình bậc cao được sử dụng phổ biến trong giảng dạy và phát triển các phần mềm ứng dụng.
- Python IDLE là môi trường cho phép viết và thực thi chương trình bằng ngôn ngữ Python theo một trong hai cách: thực thi trực tiếp từng dòng lệnh trong cửa sổ **Shell**; sử dụng trình soạn thảo để viết các dòng lệnh, lưu vào tệp .py và thực thi chương trình.

LUYỆN TẬP

1. Viết lệnh `print()` để in ra màn hình dòng chữ `"Ngôn ngữ lập trình Python"`.

2. Nếu nhập các lệnh sau vào cửa sổ **Shell** thì kết quả nhận được là gì?

a) 14+23*6; b) 15/(2+3).

3. Viết chương trình sử dụng các lệnh `print()` để in ra màn hình các thông tin của em, bao gồm: họ và tên, lớp, trường, địa chỉ tương tự thông tin dưới đây.

```
Họ và tên: Lê Duy Anh
Lớp: 10A1
Trường: THPT chuyên Lê Hồng Phong
Địa chỉ: 235 Nguyễn Văn Cừ, Phường 4, Quận 5, TP. Hồ Chí Minh
```

VẬN DỤNG

1. Em hãy tìm hiểu và sử dụng kí tự xuống dòng `\n` để thực hiện yêu cầu ở Bài tập 3 của phần LUYỆN TẬP bằng một dòng lệnh duy nhất.

2. Em tìm hiểu về một ngôn ngữ lập trình bậc cao (JavaScript, Java, Swift, Python,...) và liệt kê các ứng dụng có thể được xây dựng bởi ngôn ngữ lập trình bậc cao đó.

BÀI F2 BIẾN, HẰNG VÀ BIỂU THỨC SỐ HỌC

MỤC TIÊU **Sau bài học này, em sẽ:**

- Hiểu cách sử dụng biến, hằng.
- Hiểu cách sử dụng các phép toán số học và xây dựng biểu thức số học.

KHỞI ĐỘNG

Em hãy viết thuật toán tính diện tích của hình tròn khi biết bán kính.

KHÁM PHÁ

1. Biến và hằng

Việc giải quyết bài toán cần phải sử dụng dữ liệu gồm các hằng và biến.

Hằng là giá trị cố định, không thay đổi trong quá trình thực thi chương trình; ví dụ, số nguyên 5, số thực 2.5, giá trị logic True,... là các hằng.

Nói chung, trong nhiều ngôn ngữ lập trình, biến là một tên của vùng nhớ, được gọi là tên biến, dùng để lưu trữ dữ liệu và có thể bị thay đổi giá trị trong quá trình thực thi chương trình. Ví dụ, biến a là tên của vùng nhớ chứa giá trị 5, sau đó thay đổi vùng nhớ này chứa giá trị 7.

Tuy nhiên, trong ngôn ngữ lập trình Python, biến là một tên, được gọi là tên biến, dùng để "trỏ" đến một vùng nhớ lưu trữ dữ liệu; trong quá trình thực thi chương trình, biến này có thể "trỏ" đến một vùng nhớ khác. Ví dụ, ban đầu biến a "trỏ" đến vùng nhớ chứa giá trị 5, sau đó thay đổi biến a "trỏ" đến vùng nhớ khác chứa giá trị 7.

Quy tắc đặt tên biến trong Python như sau: bắt đầu bằng một chữ cái (a-z, A-Z) hoặc dấu gạch dưới (_), không được bắt đầu bằng số; các kí tự tiếp theo không phải kí tự đặc biệt, không phải khoảng trắng; có phân biệt chữ hoa và chữ thường; không được trùng với các từ khoá (*Bảng 1*).

Bảng 1. Một số từ khoá thường dùng trong Python

and	as	assert	break	class	continue	def	del	elif
else	except	exec	False	finally	for	from	global	if
import	in	is	lambda	None	not	or	pass	print
raise	return	True	try	while	with	yield		

Trong ngôn ngữ lập trình Python, biến được khởi tạo khi thực hiện lệnh gán. Lệnh gán có cú pháp như sau:

```
<tên biến> = <giá trị>
```

Trong đó, `<tên biến>` là tên do người lập trình tự đặt theo quy tắc đặt tên; dấu `=` là dấu gán; `<giá trị>` có thể là số nguyên, số thực, biểu thức số học,... Quan sát *Hình 1*, biến `a` được gán giá trị bằng `5`, biến `b` được gán giá trị bằng `2`, sau đó các giá trị trên được Python sử dụng cho biểu thức tính toán `a*2+b/2` và in ra giá trị bằng `11.0`.

```
>>> a = 5
>>> b = 2
>>> a*2+b/2
11.0
```

Hình 1. Biến a, b được khởi tạo và sử dụng cho việc tính toán

Để cho chương trình dễ đọc, dễ hiểu, tên biến nên tuân theo các quy định do nhóm lập trình viên đưa ra như sau: sử dụng quy ước "camelCase" (viết thường chữ cái đầu tiên và viết hoa chữ cái đầu tiên của các từ tiếp theo, ví dụ `ngayThangNamSinh`), sử dụng tên gợi nhớ, có ý nghĩa (ví dụ dùng tên `hoTen` thay cho `ht`); hạn chế sử dụng dấu gạch dưới (_) để phân cách các từ trong tên biến (ví dụ `ho_Ten`); tránh sử dụng các tên biến tương tự nhau (như `s` và `S`, `item` và `items`, `i1` và `il`).

Ví dụ 1: Trong hoạt động KHỞI ĐỘNG, em sử dụng biến `banKinh` lưu trữ giá trị của bán kính (được nhập từ bàn phím) và biến `pi` lưu giữ giá trị số π (3.14159) để tính diện tích hình tròn. Trong Python, chương trình sẽ báo lỗi khi em đặt tên biến là `ban Kinh` (tên biến không được có khoảng trắng) hoặc `1r` (tên biến không được bắt đầu bằng số).

Không giống với một số ngôn ngữ lập trình bậc cao khác, khi thực hiện lệnh gán giá trị cho biến, Python xác định kiểu dữ liệu của biến là kiểu dữ liệu của giá trị được gán. Vì vậy, trong Python, lập trình viên không cần phải khai báo kiểu dữ liệu của biến. Ví dụ, trong cùng một đoạn lệnh, các lệnh gán sẽ gán biến a với các giá trị thuộc các kiểu dữ liệu khác nhau (*Hình 2*).

```
>>> a = 5    # Biến a nhận giá trị 5, kiểu số nguyên
>>> a*2
10
>>> a = 5.2 # Biến a nhận giá trị 5.2, kiểu số thực
>>> a*10
52.0
```

Hình 2. Biến a được gán các giá trị thuộc các kiểu dữ liệu khác nhau

Lưu ý: Trong *Hình 2*, phần nội dung được tô màu đỏ, bắt đầu bằng kí tự `#`, gọi là nội dung chú thích. Chú thích trong Python không được thực thi, chỉ cung cấp thêm thông tin giải thích về đoạn lệnh, cách hoạt động, mục đích của đoạn lệnh. Chú thích là một nội dung hữu ích đối với người đọc, giúp hiểu rõ hơn ý nghĩa, cách sử dụng, hỗ trợ cho việc chỉnh sửa chương trình sau này.

Ngôn ngữ lập trình Python còn cho phép thực hiện lệnh gán nhiều giá trị đồng thời cho nhiều biến. Quan sát đoạn lệnh trong *Hình 3*, biến `a` được gán giá trị `1`, biến `b` được gán giá trị `2`, biến `c` được gán giá trị `3` và thực hiện tính toán cho ra kết quả là `7`.

```
>>> a,b,c = 1,2,3
>>> a+b*c
7
```

Hình 3. Thực hiện lệnh gán nhiều giá trị cho nhiều biến

Chân trời sáng tạo

Trong Python, hằng là một biến có giá trị không bị thay đổi sau khi được khởi tạo (được gán lần đầu tiên). Để phân biệt giữa biến và hằng, lập trình viên sử dụng toàn bộ chữ in hoa, dùng dấu gạch dưới (_) để phân cách các từ khi đặt tên hằng. Các giá trị cố định, được viết trực tiếp trong mã nguồn cũng là hằng. Ví dụ: `a = 1 + 2` thì `1`, `2` là hằng; `b = True` thì `True` là hằng, `s = "Tin học"` thì `"Tin học"` cũng là hằng.

Ví dụ 2: Trong hoạt động KHỞI ĐỘNG, vì giá trị số π (3.14159) không bị thay đổi trong suốt quá trình thực thi chương trình, em nên đặt tên `PI` để lưu trữ hằng số : `PI = 3.14159`.

```
>>>a,b = 1,2
>>>b,a = a,b
```

Hình 4. Thực hiện lệnh gán nhiều giá trị cho nhiều biến

1. Quan sát đoạn lệnh trong *Hình 4*, em hãy cho biết giá trị của hai biến `a` và `b` sau khi thực hiện xong đoạn lệnh.

2. Giả sử em cần khởi tạo một biến lưu trữ giá trị diện tích hình tròn như ở hoạt động KHỞI ĐỘNG, em nên đặt tên biến như thế nào?

- Hằng là giá trị cố định, không thay đổi trong quá trình thực thi chương trình.
- Trong nhiều ngôn ngữ lập trình, biến là một tên của vùng nhớ dùng để lưu trữ dữ liệu và có thể bị thay đổi giá trị trong quá trình thực thi chương trình.

2. Biểu thức số học và phép toán số học

Khi khởi tạo một biến, giá trị dùng để gán cho biến này là `<biểu thức số học>`. Biểu thức số học có thể là một hằng số, một biến số hoặc là sự kết hợp của các hằng số, biến số với các phép toán số học (*Bảng 2*), tuân theo quy tắc thứ tự ưu tiên của các phép toán số học để tính toán và tạo ra kết quả là một giá trị số. Ví dụ, biểu thức số học được sử dụng để tính diện tích hình tròn trong hoạt động KHỞI ĐỘNG là `PI*banKinh*banKinh` hoặc `PI*banKinh**2`.

Bảng 2. Các phép toán số học

Phép toán	Kí hiệu trong Python	Ví dụ	Kết quả
Cộng	+	3+4	7
Trừ	-	3-4	-1
Nhân	*	3*4	12

Phép toán	Kí hiệu trong Python	Ví dụ	Kết quả
Luỹ thừa	**	3**4	81
Chia	/	3/4	0.75
Chia lấy phần nguyên	//	3//4	0
Chia lấy phần dư	%	3%4	3

Việc gán giá trị của `<biểu thức số học>` cho `<tên biến>` được thực hiện bằng lệnh gán với cú pháp:

```
<tên biến> = <biểu thức số học>
```

Giá trị của `<biểu thức số học>` ở vế phải sẽ được tính toán và gán kết quả cho biến ở vế trái. Ví dụ, lệnh gán `dienTichHinhtron = pi*banKinh**2` có nghĩa là tính diện tích hình tròn bằng biểu thức `pi*banKinh**2` và gán kết quả cho biến `dienTichHinhTron`.

Các cặp ngoặc tròn được sử dụng để xác định thứ tự ưu tiên thực hiện phép tính trong biểu thức số học. Thứ tự ưu tiên thực hiện các phép tính trong Python cũng tương tự như trong toán học. Quan sát *Hình 5*, với giá trị của các biến `a`, `b`, `c` lần lượt là `1`, `2`, `3`, hai biểu thức số học `a+b*c` (phép nhân được tính trước phép cộng) và `(a+b)*c` (phép cộng được tính trước phép nhân) có hai kết quả khác nhau.

```
>>> a,b,c = 1,2,3
>>> a+b*c
7
>>> (a+b)*c
9
```

Hình 5. Minh hoạ về thứ tự thực hiện các phép tính trong Python

Chân trời sáng tạo

1. Em hãy đặt tên biến và viết biểu thức tính diện tích xung quanh, diện tích toàn phần và thể tích của hình hộp chữ nhật khi biết chiều dài, chiều rộng và chiều cao.

2. Quan sát các lệnh ở *Hình 6* và cho biết kết quả nhận được sau khi thực thi.

```
>>> a,b,c = 1,-5,4
>>> delta = b**2-4*a*c
>>> x1 = (-b+delta**(0.5))/(2*a)
>>> x2 = (-b-delta**(0.5))/(2*a)
>>> print(x1,x2)
```

Hình 6. Đoạn lệnh thực thi các thao tác tính toán

3. Viết lệnh tìm phần nguyên của phép chia 2024 cho 26 và dự đoán kết quả nhận được.

- Biểu thức số học có thể là một hằng số, một biến số hoặc là sự kết hợp của các hằng số, các biến số với các phép toán số học. Cặp dấu ngoặc tròn có thể sử dụng để xác định thứ tự ưu tiên thực hiện các phép toán. Trong Python, thứ tự thực hiện các phép toán trong biểu thức số học tương tự như trong toán học.
- Giá trị của biểu thức số học có thể được gán cho một biến bằng lệnh gán với cú pháp `<tên biến> = <biểu thức>`

LUYỆN TẬP

1. Để đặt tên biến lưu trữ điểm kiểm tra trung bình ba môn Ngữ văn, Toán, Ngoại ngữ của các bạn học sinh trong lớp 10A, các bạn An, Bình, Cường, Diễm có các đề xuất đặt tên như sau.

– Bạn An: `dTB`

– Bạn Bình: `Điểm TB`

– Bạn Cường: `Điểm_TB`

– Bạn Diễm: `diemTrungBinh`

Theo em, cách đặt tên nào ở trên sẽ bị báo lỗi trong Python? Nếu theo cách đặt tên gợi nhớ, em chọn đặt tên biến theo đề xuất của bạn nào?

2. Quan sát chương trình bên dưới và cho biết chương trình này thực hiện nhiệm vụ gì? Em hãy giải thích chương trình này có thực thi được trong Python hay không?

```
a,b,c = 3,4,5
p = (a+b+c)/2
S = (p(p-a)(p-b)(p-c))^(0.5)
print(S)
```

3. Hãy viết lệnh tìm phần dư của phép chia 2024 cho 26 và cho biết kết quả nhận được.

VẬN DỤNG

1. Em hãy viết lệnh để tính giá trị của các biểu thức sau trên cửa sổ **Shell**.

a) $12 - \dfrac{3(5+7)}{\sqrt{9}}$.

b) $\sqrt{1+\sqrt{7+\sqrt{3+\sqrt{1}}}}$.

2. Cho trước giá trị `n`, em hãy soạn thảo chương trình để in ra màn hình các giá trị sau:

a) Tổng các số tự nhiên từ `1` đến `n`, biết rằng $1+2+...+n = \dfrac{1}{2}n(n+1)$.

b) Tổng của các số chẵn từ `2` đến `2n`.

c) Tổng các số lẻ từ `1` đến `2n+1`.

d) Tổng lập phương các số tự nhiên từ `1` đến `n`, biết rằng $1^3+2^3+...+n^3 = \dfrac{1}{4}n^2(n+1)^2$.

BÀI F3 CÁC LỆNH VÀO/RA VÀ CHUYỂN ĐỔI KIỂU DỮ LIỆU

MỤC TIÊU **Sau bài học này, em sẽ:**

- Sử dụng được các lệnh vào/ra đơn giản và chuyển đổi một số kiểu dữ liệu cơ bản.

KHỞI ĐỘNG

Qua các bài học đầu tiên của chủ đề F, em có thể viết được một chương trình Python để thực hiện một số thao tác tính toán đơn giản với dữ liệu được nhập bằng lệnh gán và được in ra màn hình bằng lệnh `print()`. Để tăng tính tương tác với người dùng khi thực thi chương trình, em hãy tìm hiểu câu lệnh Python cho phép in ra màn hình thông báo hướng dẫn người dùng nhập dữ liệu từ bàn phím và gán vào biến.

KHÁM PHÁ

1. Các lệnh vào/ra

a) Lệnh vào (hoặc lệnh nhập)

Lệnh `input()` của Python là một lệnh gọi hàm `input()` để nhập dữ liệu từ bàn phím và gán cho một biến. Cú pháp đơn giản của lệnh `input()` như sau:

```
<tên biến> = input([<dòng thông báo>])
```

Trong đó, `<tên biến>` là tên do người lập trình tự đặt theo quy tắc đặt tên, `input()` là lệnh nhập, `[<dòng thông báo>]` là một xâu được hiển thị trên màn hình để nhắc người dùng nhập dữ liệu thích hợp. `<dòng thông báo>` là một tuỳ chọn, có thể ghi hoặc không ghi. Khi thực hiện lệnh gán, hàm `input()` trả về một xâu và gán cho `<tên biến>`.

Ví dụ 1: Nhập họ và tên từ bàn phím và gán vào biến `name`:

```
>>> name = input("Nhập họ và tên: ")
    Nhập họ và tên: Nguyễn Văn A
```

Ví dụ 2: Nhập hai giá trị từ bàn phím và gán vào hai biến `a`, `b`.

```
>>> a = input("Nhập giá trị a: ") # Nhập giá trị và gán vào biến a
    Nhập giá trị a: 5
>>> b = input("Nhập giá trị b: ") # Nhập giá trị và gán vào biến b
    Nhập giá trị b: 3
```

Trong trường hợp muốn nhập đồng thời nhiều giá trị cho nhiều biến khác nhau, ta sử dụng hàm `split()` của kiểu dữ liệu *string* (sẽ được giới thiệu trong bài học F9) để tách thành một hoặc nhiều xâu con. Cú pháp lệnh như sau:

```
<tên biến 1>, <tên biến 2>,…, <tên biến N> = input().split()
```

hoặc:

```
<tên biến 1>, <tên biến 2>,…, <tên biến N> = input([<dòng thông báo>]).split()
```

Ví dụ 3: Nhập ba giá trị được phân cách nhau bởi ít nhất một khoảng trắng và lần lượt gán cho các biến a, b, c.

```
>>>a,b,c = input("Nhập ba giá trị: ").split() # Lệnh nhập ba giá trị
   Nhập ba giá trị: 3 6  7
>>>a
   '3'
>>>b
   '6'
>>>c
   '7'
```

b) Lệnh ra (hoặc lệnh xuất)

Lệnh print() của Python là một lệnh gọi hàm print() để xuất kết quả ra màn hình. Cú pháp của lệnh print() như sau:

```
print(<danh sách biểu thức>[, sep = '<xâu ngăn cách>'][, end = '<xâu kết thúc>'])
```

Với <danh sách biểu thức> là danh sách các hằng số, biến số hoặc biểu thức cần xuất ra màn hình; sep, end là các từ khoá; <xâu ngăn cách> là xâu ngăn cách các giá trị cần hiển thị, mặc định sep = ' ' (mặc định xâu ngăn cách các giá trị là 1 khoảng trắng); <xâu kết thúc> là giá trị được in ra màn hình sau khi thực hiện lệnh print(), mặc định end = '\n' (mặc định sau khi thực hiện lệnh print(), chương trình in ra kí tự xuống dòng mới). sep = '<xâu ngăn cách>' và end = '<xâu kết thúc>' là các tuỳ chọn (có thể ghi hoặc không ghi).

Ví dụ 4: Các biểu thức với <xâu ngăn cách> và <xâu kết thúc> khác nhau.

```
>>>a = input("Nhập họ và tên lót: ")
   Nhập họ và tên lót: Ngô Văn
>>>b = input("Nhập tên: ")
   Nhập tên: A
>>>c = input("Nhập tuổi: ")
   Nhập tuổi: 15
>>>print("Tôi tên:",a,b,",",c,"tuổi")
   Tôi tên: Ngô Văn A , 15 tuổi
>>>print("Tôi tên:",a,b,",",c,"tuổi",sep='-')
   Tôi tên:-Ngô Văn-A-,-15-tuổi
>>>print("Họ:",a,"Tên:",b,c,"tuổi",sep=' ',end='!')
   Họ: Ngô Văn Tên: A 15 tuổi!
```

Trong trường hợp muốn đưa các biểu thức vào trong xâu, ta có thể sử dụng định dạng f-string để hiển thị kết quả ra màn hình với cú pháp như sau:

```
print(f"<xâu 1>{<biểu thức 1>}<xâu 2>{<biểu thức 2>}…<xâu N>{<biểu thức N>}")
```

Trong đó, <biểu thức 1>, <biểu thức 2>,…, <biểu thức N> có thể là biến, biểu thức số học, biểu thức logic,… được đặt bên trong dấu ngoặc nhọn {}. Các biểu thức này sẽ được tính và đưa vào xâu.

Ví dụ 5: Sử dụng f-string trong lệnh print().

```
>>>a = input("Nhập họ: ")
   Nhập họ: Ngô Văn
>>>b = input("Nhập tên: ")
   Nhập tên: A
>>>c = input("Nhập tuổi: ")
   Nhập tuổi: 15
>>>print(f"Họ: {a}, Tên: {b}, Tuổi: {c}")
```

```
Họ: Ngô Văn, Tên: A, Tuổi: 15
>>>print(f"Họ và tên: {a} {b}, Tuổi: {c}")
Họ và tên: Ngô Văn A, Tuổi: 15
```

1. Hãy viết các lệnh nhập *họ tên, giới tính, sở thích* của em và gán vào các biến `ten`, `gioiTinh`, `soThich`. Sau đó dùng lệnh `print()` để hiển thị những thông tin vừa nhập ra màn hình.

2. Hãy viết các lệnh nhập hai số nguyên và gán vào hai biến a, b. Sau đó, sử dụng f-string trong lệnh `print()` để hiển thị kết quả `a+b` và `a×b` theo yêu cầu sau: Giả sử nếu lần lượt nhập 7, 4 cho hai biến `a`, `b`. Kết quả hiển thị là:

7 cộng 4 = 11 và 7 nhân 4 = 28.

- Hàm `input()` dùng để nhập dữ liệu từ bàn phím và trả về một xâu kí tự.
- Hàm `print()` dùng để hiển thị dữ liệu ra màn hình.

2. Ứng dụng chuyển đổi kiểu dữ liệu trong các lệnh vào/ra

Như đã đề cập ở Bài F2, trong Python, kiểu dữ liệu của biến là kiểu dữ liệu của giá trị khi thực hiện lệnh gán và có thể thuộc một trong các kiểu dữ liệu: int (kiểu số nguyên), float (kiểu số thực), str (kiểu xâu kí tự), bool (kiểu logic). Hàm `type()` cho biết kiểu dữ liệu của một biến. Quan sát *Hình 1a*, hai biến `a`, `b` nhận kiểu dữ liệu của biểu thức vế phải của phép gán nên biến `a` có kiểu số nguyên, biến `b` có kiểu số thực.

```
>>>a = 5
>>>b = a/1
>>>type(a)
<class 'int'>
>>>type(b)
<class 'float'>
```

Hình 1a.

```
>>>a = input("Nhập giá trị a: ")
Nhập giá trị a: 5
>>>b = 5
>>>type(a)
<class 'str'>
>>>type(b)
<class 'int'>
```

Hình 1b.

Hình 1. Minh hoạ các kiểu dữ liệu khác nhau

Quan sát *Hình 1b*, nhập dữ liệu từ bàn phím bằng lệnh `input()`, dữ liệu trả về là một xâu kí tự. Như vậy, cần phải chuyển đổi dữ liệu từ kiểu xâu kí tự sang kiểu số để gán cho các biến số và sử dụng trong các biểu thức số học.

Để thực hiện việc chuyển đổi kiểu dữ liệu, Python cung cấp các hàm chuyển đổi như: `int(x)` chuyển đổi `x` thành số nguyên, `float(x)` chuyển đổi `x` thành số thực, `str(x)` chuyển đổi `x` thành xâu kí tự.

Lưu ý rằng, nếu `x` là một xâu kí tự thì `x` phải biểu diễn hằng số nguyên đối với hàm `int(x)` và `x` phải biểu diễn hằng số nguyên/thực đối với hàm `float(x)`.

Quan sát *Hình 2a*, dữ liệu được nhập từ bàn phím là kiểu xâu, không thực hiện chuyển đổi dữ liệu về kiểu số nguyên, phép tính `a+b` được chương trình hiểu là phép nối các xâu, cho ra kết quả là xâu `12`. Ở *Hình 2b*, việc chuyển đổi dữ liệu về kiểu số nguyên được thực hiện ngay sau khi nhập dữ liệu từ bàn phím và gán vào biến `a` và `b`, khi đó phép tính `a+b` được tính toán như bình thường và cho kết quả là số nguyên `3`.

```
>>> a = input("Nhập giá trị a: ")
    Nhập giá trị a: 1
>>> b = input("Nhập giá trị b: ")
    Nhập giá trị b: 2
>>> print(f"{a} + {b} =",a+b)
    1 + 2 = 12
```

Hình 2a.

```
>>> a = int(input("Nhập giá trị a: "))
    Nhập giá trị a: 1
>>> b = int(input("Nhập giá trị b: "))
    Nhập giá trị b: 2
>>> print(f"{a} + {b} =",a+b)
    1 + 2 = 3
```

Hình 2b.

Hình 2. Minh hoạ việc áp dụng chuyển đổi dữ liệu

```
>>> int("5")
    5
>>> int("5.0")
    Traceback (most recent call last):
      File "<pyshell#32>", line 1, in <module>
        int("5.0")
    ValueError: invalid literal for int() with base 10: '5.0'
```

Hình 3a.

```
>>> float(5+3)
    8.0
>>> float("5+3")
    Traceback (most recent call last):
      File "<pyshell#34>", line 1, in <module>
        float("5+3")
    ValueError: could not convert string to float: '5+3'
```

Hình 3b.

Hình 3. Minh hoạ các trường hợp báo lỗi khi sử dụng lệnh int(), float() chuyển đổi xâu

Quan sát *Hình 3a*, `int("5.0")` bị lỗi vì xâu `"5.0"` biểu diễn số thực, không phải là số nguyên. Ở *Hình 3b*, `float("5+3")` bị lỗi vì xâu `"5+3"` biểu diễn một biểu thức, không phải là hằng số.

1. Để in dòng chữ "Chào bạn" ra màn hình, em sẽ sử dụng câu lệnh nào sau đây?

A. `print("Chào bạn")` B. `print(Chào bạn)`

C. `print "Chào bạn"` D. `print Chào bạn`

2. Lệnh `str(1+2)` sẽ cho kết quả nào sau đây?

A. `'12'` B. `3`

C. `'1+2'` D. `'3'`

- Một số kiểu dữ liệu cơ bản trong Python là int (kiểu số nguyên), float (kiểu số thực), str (kiểu xâu kí tự).
- Hàm `int(x)` chuyển đổi x thành số nguyên. Hàm `float(x)` chuyển đổi x thành số thực. Hàm `str(x)` chuyển đổi x thành xâu kí tự.
- Trong một biểu thức, dữ liệu cần được chuyển đổi về kiểu thích hợp để tính toán giá trị của biểu thức này.

LUYỆN TẬP

1. Viết chương trình nhập độ dài cạnh đáy và chiều cao của một tam giác từ bàn phím, sau đó in ra màn hình diện tích của tam giác này.

2. Viết chương trình nhập điểm kiểm tra các môn Toán, Ngữ văn, Tin học, Ngoại ngữ của một học sinh từ bàn phím; sau đó, in điểm trung bình của học sinh đó ra màn hình, với môn Toán và Ngữ văn được tính hệ số 2.

3. Quan sát các câu lệnh bên dưới và cho biết kết quả nhận được là gì?

```
>>>x = int(input("Nhập số thứ nhất: "))
   Nhập số thứ nhất: 5
>>>y = int(input("Nhập số thứ hai: "))
   Nhập số thứ hai: 7
>>>z = (y+x)*(y-x)//y
>>>print(z)
```

A. "3" B. 3.43 C. 3 D. 4

4. Quan sát các câu lệnh bên dưới và cho biết kết quả nhận được là gì?

```
>>>x = int(input("Nhập số thứ nhất: "))
   Nhập số thứ nhất: 5
>>>y = int(input("Nhập số thứ hai: "))
   Nhập số thứ hai: 7
>>>z = (x//y)*x+(x%y)
>>>print(z)
```

A. 3.57 B. 5 C. "5" D. 8.57

VẬN DỤNG

Biết một hộp chứa được 6 lon nước ngọt, một thùng chứa được 4 hộp, một kiện hàng bao gồm 20 thùng. Viết chương trình nhập số lon nước ngọt, in ra màn hình số kiện hàng đóng gói được, số thùng chứa còn dư (chưa đủ để đóng gói thành kiện), số hộp còn dư (chưa đủ để đóng gói thành thùng) và số lon nước ngọt còn dư (chưa đủ đóng gói vào hộp). Ví dụ, với số lon nước ngọt là 500, chương trình sau khi tính toán sẽ in ra màn hình là 1 kiện, 0 thùng, 3 hộp, 2 lon. Như vậy, với đơn hàng 500 lon nước ngọt, nhà cung cấp sẽ phải giao 1 kiện hàng, 3 hộp và 2 lon nước ngọt.

Chân trời sáng tạo

BÀI F4 THỰC HÀNH BIẾN, HẰNG, BIỂU THỨC VÀ CÁC LỆNH VÀO/RA ĐƠN GIẢN

MỤC TIÊU **Sau bài học này, em sẽ:**

- Viết được chương trình đơn giản có sử dụng biến, hằng số, lệnh gán, biểu thức số học, chuyển đổi kiểu dữ liệu và lệnh vào/ra đơn giản.

KHỞI ĐỘNG

Trong VẬN DỤNG ở Bài F3, em đã được tiếp cận một bài toán mang tính thực tế trong hoạt động sản xuất kinh doanh và sử dụng ngôn ngữ lập trình Python để viết chương trình giải quyết. Em hãy trao đổi với các bạn về một bài toán thực tế mà em và các bạn có thể sử dụng những kiến thức về Python để viết chương trình giải quyết, ví dụ như bài toán tính thời gian di chuyển từ nhà đến trường,...

THỰC HÀNH

Nhiệm vụ 1. Viết chương trình tính thời gian di chuyển từ nhà đến trường

Yêu cầu: Minh cần tính thời gian di chuyển từ nhà đến trường. Biết rằng, nhà Minh cách trường khoảng s km, vận tốc trung bình của Minh khi di chuyển bằng xe đạp là 12 km/h. Em hãy viết chương trình tính thời gian Minh di chuyển từ nhà đến trường bằng xe đạp, kết quả nhận được tính theo đơn vị phút và làm tròn đến hàng đơn vị.

Dữ liệu vào: Số thực s là độ dài quãng đường từ nhà của Minh đến trường (đơn vị km).

Dữ liệu ra: Số nguyên t là thời gian Minh di chuyển bằng xe đạp từ nhà đến trường (hiển thị theo giờ và phút, làm tròn đến đơn vị).

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập độ dài quãng đường: 2`	`0 giờ 10 phút`

Hướng dẫn: Sử dụng công thức tính: thời gian = quãng đường/vận tốc.

1. Nhập giá trị s từ bàn phím.
2. Áp dụng công thức tính thời gian: `t = s/12*60`, sử dụng hàm `round()` để làm tròn đến hàng đơn vị.
3. In ra màn hình số giờ (phần nguyên của phép chia `t/60`) và số phút (phần dư của phép chia `t/60`).

Chương trình tham khảo:

```
s = float(input("Nhập độ dài quãng đường: "))
t = round(s/12*60) # Dùng hàm round() để làm tròn
print(t//60,"giờ",t%60,"phút")
```

Nhiệm vụ 2. Viết chương trình tách một số có ba chữ số thành hàng đơn vị, hàng chục và hàng trăm

Yêu cầu: Viết chương trình tách số có ba chữ số thành hàng đơn vị, hàng chục và hàng trăm.

Dữ liệu vào: Số nguyên có ba chữ số.

Dữ liệu ra: Ba chữ số cách nhau một khoảng trắng lần lượt là hàng đơn vị, hàng chục, hàng trăm.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
Nhập số nguyên a: 547	Hàng đơn vị, hàng chục, hàng trăm của a lần lượt là: 7 4 5

Hướng dẫn: Sử dụng chia lấy phần dư (%) và chia lấy phần nguyên (//) để tách số.

1. Nhập số nguyên có ba chữ số từ bàn phím và gán cho biến a.
2. Gán biến `donVi = a%10`
3. `a = a//10` (loại bỏ số hàng đơn vị).
4. Gán biến `hangChuc = a%10`
5. Gán biến `hangTram = a//10`
6. In kết quả `donVi`, `hangChuc`, `hangTram` ra màn hình.

Chương trình tham khảo:

```
a = int(input("Nhập số nguyên a: "))
donVi = a%10      # donVi lưu chữ số hàng đơn vị
a = a//10
hangChuc = a%10   # hangChuc lưu chữ số hàng chục
hangTram = a//10  # hangTram lưu chữ số hàng trăm
print("Hàng đơn vị, hàng chục, hàng trăm của a lần lượt là:")
print(donVi,hangChuc,hangTram)
```

Nhiệm vụ 3. Chương trình tính điểm trung bình có hệ số

Bạn Lâm muốn viết chương trình tính điểm trung bình môn học kì 1 để tính nhanh điểm trung bình môn học kì 1 của các bạn trong lớp. Biết rằng môn Tin học ở học kì 1 có 3 cột điểm kiểm tra đánh giá thường xuyên (hệ số 1), 1 cột điểm kiểm tra đánh giá giữa kì (hệ số 2), 1 cột điểm kiểm tra đánh giá cuối kì (hệ số 3). Em hãy viết chương trình nhập các điểm kiểm tra và in ra màn hình điểm trung bình môn học kì 1 của môn Tin học.

Yêu cầu: Viết chương trình nhập các điểm kiểm tra thường xuyên (3 giá trị, hệ số 1), điểm kiểm tra giữa kì (1 giá trị, hệ số 2), điểm kiểm tra cuối kì (1 giá trị, hệ số 3) và xuất ra điểm trung bình môn học kì 1.

Dữ liệu vào: Các điểm kiểm tra thường xuyên, giữa kì, cuối kì.

Dữ liệu ra: Điểm trung bình môn học kì 1, làm tròn đến 1 chữ số thập phân.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
Nhập điểm đánh giá thường xuyên 1: 7 Nhập điểm đánh giá thường xuyên 2: 8 Nhập điểm đánh giá thường xuyên 3: 8 Nhập điểm đánh giá giữa kì: 8 Nhập điểm đánh giá cuối kì: 8	Điểm trung bình môn là: 7.9

Hướng dẫn: Sử dụng công thức tính điểm trung bình để tính và in ra màn hình điểm trung bình môn.

❶ Nhập các điểm kiểm tra thường xuyên `diemTX1, diemTX2, diemTX3`; điểm kiểm tra giữa kì `diemGK`; điểm kiểm tra cuối kì `diemCK`.

❷ Tính `diemTB = (diemTX1+diemTX2+diemTX3+2*diemGK+3*diemCK)/8`

❸ In ra màn hình điểm trung bình môn, dùng lệnh `round(diemTB,1)` để làm tròn điểm trung bình môn đến 1 chữ số thập phân.

Chương trình tham khảo:

```
diemTX1 = float(input("Nhập điểm đánh giá thường xuyên 1: "))
diemTX2 = float(input("Nhập điểm đánh giá thường xuyên 2: "))
diemTX3 = float(input("Nhập điểm đánh giá thường xuyên 3: "))
diemGK = float(input("Nhập điểm đánh giá giữa kì: "))
diemCK = float(input("Nhập điểm đánh giá cuối kì: "))
diemTB = (diemTX1+diemTX2+diemTX3+2*diemGK+3*diemCK)/8
print("Điểm trung bình môn là:",round(diemTB,1))
```

LUYỆN TẬP

Nhiệm vụ 4. Tính số giao điểm tối đa có thể có của các đường thẳng phân biệt

Yêu cầu: Viết chương trình tính số giao điểm tối đa có thể có của n đường thẳng phân biệt.

Dữ liệu vào: Số nguyên n là số đường thẳng phân biệt.

Dữ liệu ra: Số nguyên là số giao điểm tối đa.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập số đường thẳng: 10`	`Số giao điểm là: 45`

Hướng dẫn: Cứ hai đường thẳng phân biệt sẽ tạo thành tối đa 1 giao điểm. Do đó, chọn hai đường thẳng bất kì trong n đường thẳng là một tổ hợp chập 2 của n. Suy ra số giao điểm là

$$C_n^2 = \frac{n!}{2!(n-2)!} = \frac{n(n-1)}{2}.$$

VẬN DỤNG

Nhiệm vụ 5. Chương trình tính khối lượng

Yêu cầu: Viết chương trình in ra màn hình khối lượng m (kg) của một tấm thép trơn với chiều dài cd (m), chiều rộng cr (m), độ dày h (mm) biết khối lượng riêng D = 7,85 (g/cm^3).

Biết khối lượng tấm thép được tính bằng công thức:

Khối lượng = Chiều dài × Chiều rộng × Độ dày × Khối lượng riêng

Dữ liệu vào: Ba số lần lượt là chiều dài, chiều rộng, độ dày của tấm thép trơn.

Dữ liệu ra: Khối lượng của tấm thép (làm tròn đến hàng đơn vị).

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị

`Nhập chiều dài (đơn vị: m): 6`
`Nhập chiều rộng (đơn vị: m): 1.5`
`Nhập độ dày (đơn vị: mm): 3`
	`Khối lượng tấm thép là: 212 (kg).`

BÀI F5 LỆNH RẼ NHÁNH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Hiểu và sử dụng được các phép toán so sánh và các phép toán logic.
- Sử dụng được lệnh rẽ nhánh if để viết chương trình Python.

KHỞI ĐỘNG

Vào kì nghỉ cuối tuần vừa qua, gia đình bạn An tổ chức tham quan du lịch với giá vé điểm tham quan như ở *Hình 1*.

BẢNG GIÁ VÉ

Ngày thường (từ thứ Hai đến thứ Sáu)	60 000 đồng/vé
Cuối tuần (thứ Bảy và Chủ nhật)	120 000 đồng/vé

Hình 1. Giá vé tham quan khu du lịch

Chương trình tính tiền vé tham quan của bạn An như sau:

```
#Chương trình tính số tiền vé tham quan du lịch
songuoi = int(input("Nhập số người: "))
sotien = songuoi*120000
print("Tổng tiền vé là:",sotien,"đồng.")
```

Nếu gia đình đi tham quan vào ngày thường thì chương trình ở trên có tính đúng số tiền vé phải trả không? Tại sao?

KHÁM PHÁ

1. Biểu thức logic

a) Các phép toán so sánh

Các phép toán so sánh có hai toán hạng là các giá trị có thể so sánh được. Kết quả của phép toán so sánh là một giá trị logic: True (đúng), False (sai). *Bảng 1* liệt kê các phép toán so sánh của Python.

Bảng 1. Các phép toán so sánh

Toán tử	Phép toán	Ví dụ	Kết quả
`==`	So sánh bằng	`3 == 2`	`False`
`!=`	So sánh khác	`3 != 2`	`True`
`>`	Lớn hơn	`3 > 2`	`True`
`>=`	Lớn hơn hoặc bằng	`3 >= 2`	`True`
`<`	Nhỏ hơn	`3 < 2`	`False`
`<=`	Nhỏ hơn hoặc bằng	`3 <= 2`	`False`

b) Các phép toán logic

Toán hạng của các phép toán logic có giá trị True hoặc False. Kết quả của phép toán logic là True hoặc False. *Bảng 2* liệt kê các phép toán logic của Python.

Bảng 2. Các phép toán logic

Phép toán	Ví dụ
not	not a
and	a and b
or	a or b

Bảng 3 là bảng chân lí của phép `not`. *Bảng 4* là bảng chân lí của phép `and` và phép `or`.

Bảng 3. Bảng chân lí của phép not

a	not a
True	False
False	True

Bảng 4. Bảng chân lí của phép and và phép or

a	b	a and b	a or b
True	True	True	True
True	False	False	True
False	True	False	True
False	False	False	False

Lưu ý: Phép `and` cho kết quả là True nếu và chỉ nếu cả hai toán hạng đều là True. Phép `or` cho kết quả là False nếu và chỉ nếu cả hai toán hạng đều là False.

c) Biểu thức logic

Trong lập trình, biểu thức logic có thể là một hằng, một biến, một hàm hoặc là sự kết hợp của các hằng, các biến, các hàm với các toán tử, tuân theo quy tắc thứ tự ưu tiên của các toán tử để tính toán và tạo ra kết quả là một giá trị logic True hoặc False.

Thứ tự thực hiện ưu tiên của các toán tử trong Python được liệt kê trong *Bảng 5*.

Bảng 5. Thứ tự thực hiện ưu tiên của các toán tử trong Python

Toán tử	Mô tả
**	Luỹ thừa
* / // %	Nhân, chia, chia lấy phần nguyên, chia lấy phần dư. Các phép toán này có cùng thứ tự ưu tiên và được thực hiện lần lượt từ trái qua phải.

Toán tử	Mô tả
+ -	Cộng, trừ. Các phép toán này có cùng thứ tự ưu tiên và được thực hiện lần lượt từ trái qua phải.
== != > >= < <=	Các phép toán so sánh
not	Phép not
and	Phép and
or	Phép or

Theo em, các biểu thức sau đây có giá trị True hay False:

a) 300 % 15 == 0

b) 5 >= 5

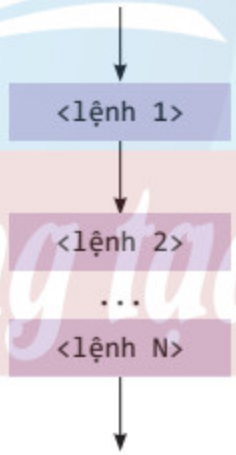
c) 6 >= 7 or 5 < 7

d) 23 // 3 != 8 and 23 % 2 == 1

- Các phép toán so sánh: so sánh bằng, so sánh khác, lớn hơn, lớn hơn hoặc bằng, nhỏ hơn, nhỏ hơn hoặc bằng. Kết quả của phép so sánh là True hoặc False.
- Các phép toán logic: phép not, phép and, phép or. Kết quả của phép logic là True hoặc False.
- Biểu thức logic có thể là một hằng, một biến, một hàm hoặc là sự kết hợp của các hằng, các biến, các hàm với các toán tử, tuân theo quy tắc thứ tự ưu tiên của các toán tử để tính toán và tạo ra kết quả là một giá trị logic True hoặc False.

2. Cấu trúc tuần tự

Cấu trúc tuần tự bao gồm một dãy các lệnh liên tiếp nhau: `<lệnh 1>`, `<lệnh 2>`, ..., `<lệnh N>`. Các lệnh được thực hiện một cách tuần tự, bắt đầu từ `<lệnh 1>` cho đến `<lệnh N>`. *Hình 2* là sơ đồ khối của cấu trúc tuần tự.



Hình 2. Sơ đồ khối của cấu trúc tuần tự

3. Cấu trúc điều kiện

Cấu trúc điều kiện (hoặc cấu trúc rẽ nhánh) cho phép việc thực hiện của chương trình (hoặc dòng điều khiển của chương trình) dựa vào việc kiểm tra một hoặc nhiều điều kiện. Các điều kiện này là các biểu thức logic có giá trị True (đúng) hoặc False (sai).

Trong Python, lệnh `if` là lệnh rẽ nhánh (hoặc lệnh điều kiện) dùng để thực hiện cấu trúc điều kiện và có ba dạng lệnh `if` là lệnh `if`, lệnh `if - else` và lệnh `if - elif - else`.

a) Lệnh if

Cú pháp của lệnh `if` như sau:

```
if <điều kiện>:
    <khối lệnh>
```

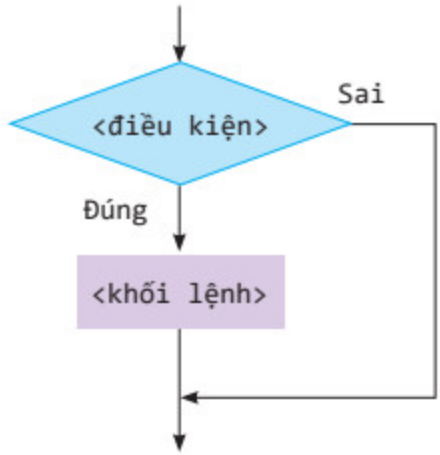
Trong đó, `<điều kiện>` là một biểu thức logic; `<khối lệnh>` bao gồm một hoặc nhiều lệnh.

Cách thực hiện lệnh `if` như sau:

– Nếu `<điều kiện>` là đúng thì thực hiện `<khối lệnh>`.

– Nếu `<điều kiện>` là sai thì không thực hiện `<khối lệnh>`, chương trình thực hiện lệnh kế tiếp (nếu có).

Hình 3 là sơ đồ khối của lệnh `if`.



Hình 3. Sơ đồ khối của lệnh if

Ví dụ 1: Một cửa hàng tạp hoá đang có chiến dịch tri ân khách hàng. Với mỗi khách hàng có hoá đơn trên 10 triệu đồng thì sẽ được chiết khấu 10% tổng số tiền phải trả. Chương trình được viết như sau:

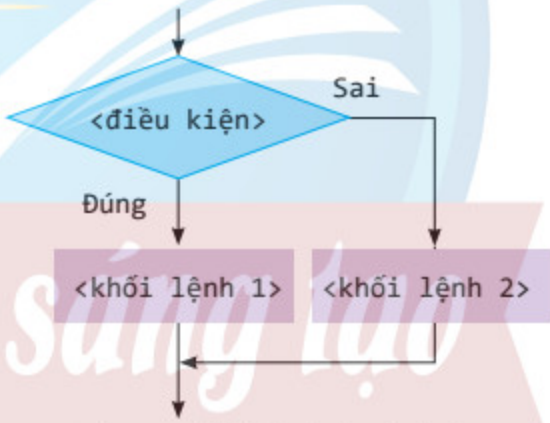
```
tienMua = int(input("Nhập tiền mua hàng: "))
tienChietKhau = 0
if tienMua >= 10000000:
    tienChietKhau = int(tienMua*0.1)
print(f"Tiền chiết khấu: {tienChietKhau}")
```

b) Lệnh if - else

Cú pháp của lệnh `if - else` như sau.

```
if <điều kiện>:
    <khối lệnh 1>
else:
    <khối lệnh 2>
```

Trong đó, `<điều kiện>` là một biểu thức logic; `<khối lệnh 1>`, `<khối lệnh 2>` bao gồm một hoặc nhiều lệnh.



Hình 4. Sơ đồ khối của lệnh if- else

Cách thực hiện lệnh `if - else` như sau:

– Nếu `<điều kiện>` là đúng thì thực hiện `<khối lệnh 1>`.

– Nếu `<điều kiện>` là sai thì thực hiện `<khối lệnh 2>`.

Hình 4 là sơ đồ khối của lệnh `if - else`.

Ví dụ 2: Nhập số nguyên a. Nếu a là số chẵn thì hiển thị a là số chẵn; ngược lại, hiển thị a là số lẻ.

```
a = int(input("Nhập a: "))
if a%2 == 0:
    print(f"{a} là số chẵn.")
else:
    print(f"{a} là số lẻ.")
```

Trường hợp việc gán giá trị cho một biến phụ thuộc vào một điều kiện có dạng như sau:

```
if <điều kiện>:
  <tên biến> = <biểu thức 1>
else:
  <tên biến> = <biểu thức 2>
```

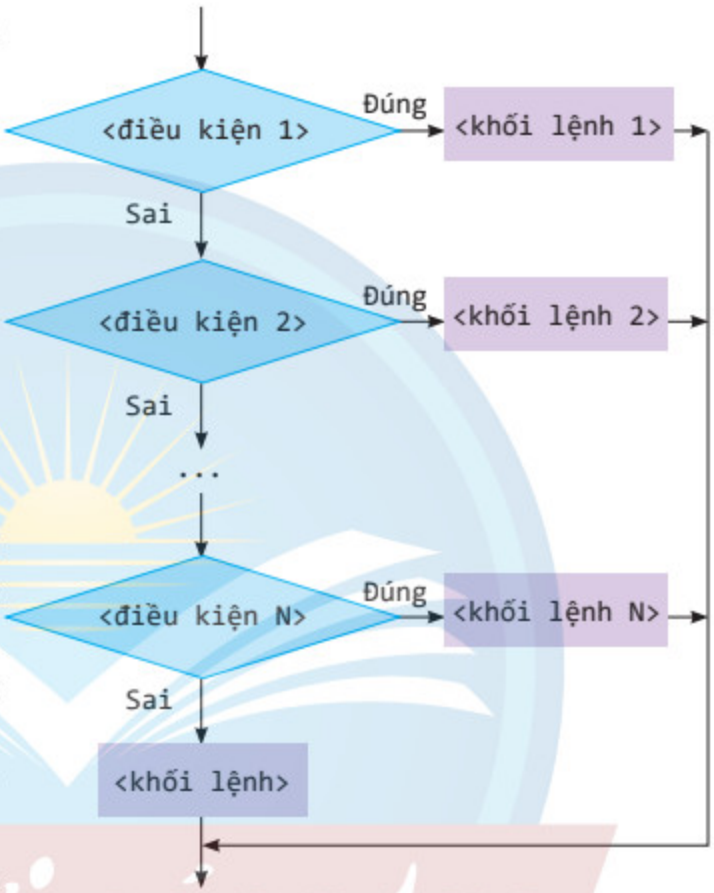
Một cách viết khác đơn giản hơn là:

```
<tên biến> = <biểu thức 1> if <điều kiện> else <biểu thức 2>
```

c) Lệnh if - elif - else

Cú pháp của lệnh `if - elif - else` như sau:

```
if <điều kiện 1>:
   <khối lệnh 1>
elif <điều kiện 2>:
   <khối lệnh 2>
...
elif <điều kiện N>:
   <khối lệnh N>
else:
   <khối lệnh>
```



Hình 5. Sơ đồ khối của lệnh if- elif - else

Trong đó: `<điều kiện 1>`, `<điều kiện 2>`, ..., `<điều kiện N>` là các biểu thức logic; `<khối lệnh 1>`, `<khối lệnh 2>`, ..., `<khối lệnh N>`, `<khối lệnh>` bao gồm một hoặc nhiều lệnh.

Cách thực hiện lệnh `if - elif - else` như sau:

– Nếu `<điều kiện 1>` là đúng thì thực hiện `<khối lệnh 1>`.

– Ngược lại nếu `<điều kiện 2>` là đúng thì thực hiện `<khối lệnh 2>`.

...

– Ngược lại nếu `<điều kiện N>` là đúng thì thực hiện `<khối lệnh N>`.

– Ngược lại, thực hiện `<khối lệnh>`.

Nếu không có `else: <khối lệnh>` , chương trình thực hiện lệnh kế tiếp (nếu có).

Hình 5 là sơ đồ khối của lệnh `if - elif - else`.

Ví dụ 3: Nhập số a. Nếu a < 100 thì hiển thị "Nhỏ hơn 100.". Nếu 100 ≤ a < 200 thì hiển thị "Lớn hơn hoặc bằng 100 và nhỏ hơn 200.". Ngược lại, hiển thị "Lớn hơn hoặc bằng 200.".

```
1 a = int(input("Nhập a: "))
2 if a < 100:
3    print("Nhỏ hơn 100.")
4 elif a < 200:
5    print("Lớn hơn hoặc bằng 100 và nhỏ hơn 200.")
6 else:
7    print("Lớn hơn hoặc bằng 200.")
```

Trong các hệ thống điều khiển cửa tự động, cảm biến được sử dụng để xác định khi nào cửa mở hoặc đóng bằng cách nhận các biến số học. Giả sử số chẵn biểu diễn cho trạng thái đóng cửa và số lẻ biểu diễn cho trạng thái cửa mở. Trong trường hợp này, kiểm tra tính chẵn lẻ của số được ghi nhận bởi cảm biến để đóng hoặc mở cửa:

– Nếu số là số chẵn (ví dụ, 2; 4; 6), trạng thái là đóng cửa.

– Nếu số là số lẻ (ví dụ, 1; 3; 5), trạng thái là mở cửa.

Việc kiểm tra tính chẵn lẻ giúp xác định và điều khiển trạng thái của cửa tự động một cách đơn giản và hiệu quả trong các ứng dụng thực tế.

Yêu cầu: Viết chương trình nhập số nguyên dương biểu diễn trạng thái của cửa tự động và hiển thị ra màn hình "Đóng cửa." nếu số nhập là số chẵn; ngược lại, hiển thị "Mở cửa.".

Dữ liệu vào: Số nguyên dương biểu diễn trạng thái của cửa tự động.

Dữ liệu ra: "Đóng cửa." hoặc "Mở cửa.".

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập trạng thái: 2`	`Đóng cửa.`
`Nhập trạng thái: 5`	`Mở cửa.`

- Cấu trúc điều kiện cho phép việc thực hiện của chương trình dựa vào việc kiểm tra một hoặc nhiều điều kiện. Các điều kiện này là các biểu thức logic có giá trị True (đúng) hoặc False (sai).
- Lệnh rẽ nhánh dùng để thực hiện cấu trúc điều kiện và có ba dạng lệnh là lệnh if, lệnh if - else và lệnh if - elif - else.

Chân trời sáng tạo

LUYỆN TẬP

Nhiệm vụ 1. Xét duyệt độ tuổi bầu cử

Khi xây dựng hệ thống bầu cử trực tuyến, tổ bầu cử cần xét duyệt độ tuổi khi tham gia bầu cử, tức là chỉ những người từ 18 tuổi trở lên mới có quyền bỏ phiếu.

Yêu cầu: Viết chương trình in ra màn hình thông báo "Bạn đủ điều kiện để bỏ phiếu." hoặc "Bạn không đủ điều kiện để bỏ phiếu." với số tuổi được nhập từ bàn phím.

Dữ liệu vào: Số nguyên dương là số tuổi của cử tri.

Dữ liệu ra: In ra màn hình là "Bạn đủ điều kiện để bỏ phiếu." nếu thoả mãn yêu cầu. Ngược lại, in ra màn hình là "Bạn không đủ điều kiện để bỏ phiếu." nếu không thoả mãn yêu cầu.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập tuổi của cư tri: 14`	`Bạn không đủ điều kiện để bỏ phiếu.`
`Nhập tuổi của cư tri: 18`	`Bạn đủ điều kiện để bỏ phiếu.`

Nhiệm vụ 2. Giảm giá sản phẩm

Tại một cửa hàng sử dụng ứng dụng đặt mua hàng trực tuyến, người chủ của cửa hàng có cài đặt ưu đãi giảm giá cho các đơn hàng có từ 5 sản phẩm trong giỏ hàng.

– Nếu có từ 5 sản phẩm trong giỏ hàng sẽ được giảm 10%.

– Nếu số lượng ít hơn 5 sẽ không áp dụng giảm giá.

Yêu cầu: Viết chương trình nhập số lượng sản phẩm và hiển thị "Được giảm 10%" ra màn hình nếu trong giỏ hàng có từ 5 sản phẩm trở lên; ngược lại, hiển thị "Không được giảm".

Dữ liệu vào: Số lượng sản phẩm có trong giỏ hàng.

Dữ liệu ra: "Được giảm 10%" hoặc "Không được giảm".

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập số sản phẩm: 17`	`Được giảm 10%`
`Nhập số sản phẩm: 3`	`Không được giảm`

VẬN DỤNG

Nhiệm vụ 3. Chấm điểm các sản phẩm

Trong một buổi triển lãm sản phẩm, ban tổ chức tiến hành chấm điểm các sản phẩm dự thi trên thang điểm 100 để xếp loại sản phẩm với *Bảng 6*.

Bảng 6. Xếp loại sản phẩm trong triển lãm

Thang điểm	Xếp loại
`Từ 90 điểm trở lên`	`A`
`75 – 89 điểm`	`B`
`60 – 74 điểm`	`C`
`Dưới 60 điểm`	`D`

Yêu cầu: Hãy viết chương trình xếp loại cho sản phẩm căn cứ vào số điểm chấm.

Dữ liệu vào: Một số nguyên là điểm của sản phẩm.

Dữ liệu ra: Loại của sản phẩm.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập điểm của sản phẩm: 80`	`Xếp loại B.`
`Nhập điểm của sản phẩm: 55`	`Xếp loại D.`

Hướng dẫn: Sử dụng dạng lệnh `if - elif - else` để kiểm tra điểm của sản phẩm thuộc thang điểm nào trong *Bảng 6* thì in ra loại sản phẩm tương ứng với thang điểm đó. Cụ thể như sau:

1. Nhập số nguyên `d`.
2. Nếu `d >= 90` thì in ra màn hình "Xếp loại A.".
   Ngược lại, nếu `d >= 75` thì in ra màn hình "Xếp loại B.".
   Ngược lại, nếu `d >= 60` thì in ra màn hình "Xếp loại C.".
   Ngược lại, in ra màn hình "Xếp loại D.".

BÀI F6 THỰC HÀNH LỆNH RẼ NHÁNH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Sử dụng được các dạng khác nhau của lệnh rẽ nhánh để giải quyết một số bài toán đơn giản.

KHỞI ĐỘNG

Em hãy quan sát chương trình sau đây và cho biết chương trình thực hiện nhiệm vụ gì? Nếu nhập vào giá trị điểm số của học sinh là 8 thì kết quả câu lệnh nào trong chương trình sẽ được thực hiện?

```
1 diem = float(input("Nhập điểm số của em: "))
2 if diem >= 8.0:
3     print("Xếp loại: Tốt.")
4 elif diem >= 6.5:
5     print("Xếp loại: Khá.")
6 elif diem >= 5.0:
7     print("Xếp loại: Đạt.")
8 else:
9     print("Xếp loại: Chưa đạt.")
```

THỰC HÀNH

Nhiệm vụ 1. Xác định tam giác

An được thầy giáo dạy môn Toán đề xuất viết chương trình xác định 3 số đo độ dài có phải là số đo 3 cạnh của tam giác hay không. Trong trường hợp 3 số đo độ dài là số đo 3 cạnh của tam giác, xác định loại của tam giác (đều, cân, vuông, vuông cân, thường) và tính diện tích tam giác.

Yêu cầu: Nhập 3 số đo độ dài, nhận xét có phải là số đo 3 cạnh của tam giác hay không. Trong trường hợp các độ dài này là số đo 3 cạnh của tam giác, in ra màn hình loại của tam giác này (đều, cân, vuông, vuông cân, thường) và diện tích của tam giác.

Dữ liệu vào: 3 số đo độ dài.

Dữ liệu ra: Các nhận xét "Không tạo thành tam giác.", "Tạo thành tam giác đều có diện tích S.", "Tạo thành tam giác cân có diện tích S.", "Tạo thành tam giác vuông có diện tích S.", "Tạo thành tam giác vuông cân có diện tích S.", "Tạo thành tam giác với diện tích S.".

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
Nhập số đo thứ nhất: 3 Nhập số đo thứ hai: 4 Nhập số đo thứ ba: 5	Tạo thành tam giác vuông có diện tích 6.0.

Hướng dẫn: Xét 3 độ dài cạnh. Nếu tổng của 2 độ dài cạnh bất kì luôn lớn hơn độ dài cạnh còn lại thì 3 cạnh này tạo thành một tam giác. Khi 3 cạnh tạo thành một tam giác, nếu 3 độ dài cạnh bằng nhau thì tạo thành tam giác đều, nếu 2 độ dài cạnh bằng nhau thì tạo thành tam giác cân, nếu tổng bình phương của 2 độ dài cạnh bằng bình phương của độ dài cạnh còn lại thì tạo thành tam giác vuông. Để tính diện tích tam giác khi biết độ dài 3 cạnh, em có thể sử dụng công thức Heron:

$$S = \sqrt{p \times (p-a) \times (p-b) \times (p-c)} \text{ với } p = \frac{1}{2} \times (a+b+c)$$

1. Nhập 3 số đo độ dài a, b, c là 3 số thực.
2. Nếu tổng của 2 số bất kì trong 3 số luôn lớn hơn số còn lại:

   Tính `p = (a+b+c)/2`

   Tính diện tích `S = round((p*(p-a)*(p-b)*(p-c))**0.5,2)` (Sử dụng hàm `round(<giá trị>,2)` để làm tròn đến hàng phần trăm).

   Nếu 3 số đo bằng nhau:

   In ra màn hình "Tạo thành tam giác đều có diện tích S.".

   Ngược lại, nếu chỉ có 2 số đo bằng nhau:

   Nếu tổng bình phương của 2 số đo bằng bình phương số đo còn lại:

   In ra màn hình "Tạo thành tam giác vuông cân có diện tích S.".

   Ngược lại:

   In ra màn hình "Tạo thành tam giác cân có diện tích S.".

   Ngược lại, nếu tổng bình phương của 2 số đo bằng bình phương số đo còn lại:

   In ra màn hình "Tạo thành tam giác vuông có diện tích S.".

   Ngược lại:

   In ra màn hình "Tạo thành tam giác có diện tích S.".

Ngược lại:

In ra màn hình "Không tạo thành tam giác".

Chương trình tham khảo:

```
a = float(input("Nhập số đo thứ nhất: "))
b = float(input("Nhập số đo thứ hai: "))
c = float(input("Nhập số đo thứ ba: "))
if a+b > c and b+c > a and c+a > b:          # Tạo thành tam giác
    p = (a+b+c)/2                            # Nửa chu vi
    S = round((p*(p-a)*(p-b)*(p-c))**0.5,2)# Diện tích
    if a == b == c:                          # Tam giác đều
        print(f'Tạo thành tam giác đều có diện tích {S}.')
    elif a == b or b == c or a == c:         # Tam giác cân
        if a**2+b**2 == c**2 or a**2+c**2 == b**2 or b**2+c**2 == a**2:
            print(f'Tạo thành tam giác vuông cân có diện tích {S}.')
        else:
            print(f'Tạo thành tam giác cân có diện tích {S}.')
    elif a**2+b**2 == c**2 or a**2+c**2 == b**2 or b**2+c**2 == a**2:
        print(f'Tạo thành tam giác vuông có diện tích {S}.')
    else:
        print(f'Tạo thành tam giác có diện tích {S}.')
else:
    print('Không tạo thành tam giác.')
```

Chân trời sáng tạo

Nhiệm vụ 2. Tính tiền nước mỗi tháng

Công ty cấp nước X lắp đặt một công tơ cho mỗi hộ gia đình và quy định bảng giá tính tiền nước theo số mét khối sử dụng cho từng tháng như sau:

Mức 1: 16 mét khối đầu thì tính 7 000 đ/mét khối.

Mức 2: Từ 17 đến 50 mét khối thì tính 8 500 đ/mét khối.

Mức 3: Từ 50 mét khối trở lên thì tính 10 000 đ/mét khối.

Yêu cầu: Viết chương trình nhập số mét khối sử dụng của một hộ gia đình và hiển thị số tiền nước ra màn hình mà hộ gia đình này phải trả theo bảng giá trên.

Dữ liệu vào: Số mét khối sử dụng.

Dữ liệu ra: Số tiền nước.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập số mét khối sử dụng: 15`	`Số tiền nước cần trả: 105000đ`

Hướng dẫn: Sử dụng dạng lệnh `if - elif - else`, từ số mét khối sử dụng, tính tiền theo số mét khối của từng mức từ mức 1 trở đi và số tiền nước mà gia đình phải trả bằng tổng tiền của các mức này.

1. Nhập số nguyên dương a là số mét khối sử dụng.
2. Nếu `a <= 16`:

   `kq = a*7000`

   Ngược lại nếu `a <= 50`:

   `kq = (16*7000)+((a-16)*8500)`

   Ngược lại:

   `kq = (16*7000)+(34*8500)+((a-50)*10000)`
3. Hiển thị `kq` ra màn hình.

Chương trình tham khảo:

```
a = int(input("Nhập số mét khối sử dụng: "))
if a <= 16:
    kq = a*7000
elif a <= 50:
    kq = 16*7000+(a-16)*8500
else:
    kq = 16*7000+34*8500+(a-50)*10000
print(f"Số tiền nước cần trả: {kq}đ")
```

LUYỆN TẬP

Hoàn thiện chương trình của Nhiệm vụ 1 ở phần THỰC HÀNH, bổ sung thêm các yêu cầu sau:

+ Kiểm tra 3 số đo đã nhập có lớn hơn 0 hay không? Nếu không phải thì in thông báo "Các số đo phải lớn hơn 0.", kết thúc chương trình.

+ Trong trường hợp 3 số đo tạo thành tam giác, làm tròn diện tích tìm được đến phần trăm (làm tròn 2 chữ số thập phân).

VẬN DỤNG

Nhiệm vụ 3. Kiểm tra số Armstrong

Lân được yêu cầu kiểm tra một số có tối đa 3 chữ số có phải là số Armstrong hay không. Biết rằng, số a được gọi là số Armstrong khi tổng luỹ thừa bậc n các chữ số của a bằng chính a, với n là số các chữ số của a. Ví dụ, mọi số có 1 chữ số đều là số Armstrong, số 153 là số Armstrong vì $1^3+5^3+3^3 = 153$.

Yêu cầu: Viết chương trình kiểm tra một số tự nhiên có tối đa 3 chữ số nhập từ bàn phím có phải là số Armstrong hay không.

Dữ liệu vào: Số tự nhiên a có tối đa 3 chữ số.

Dữ liệu ra: Thông báo "a là số Armstrong." hoặc "a không là số Amstrong.".

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập số a: 5`	`5 là số Armstrong.`
`Nhập số a: 11`	`11 không là số Armstrong.`

Hướng dẫn: Để kiểm tra số tự nhiên a là số Armstrong, sử dụng phép chia và phép lấy số dư của a với 10 để có được tất cả các chữ số của a và xác định được n là số chữ số của a. Nếu a bằng tổng của các luỹ thừa bậc n của tất cả các chữ số thì a là số Armstrong.

Nhiệm vụ 4. Giải phương trình bậc hai $ax^2 + bx + c = 0$

Quỳnh được yêu cầu viết chương trình giải phương trình bậc hai $ax^2 + bx + c = 0$ với các hệ số a, b, c đã cho là số thực và $a \neq 0$. Quỳnh được lưu ý rằng nếu hệ số a nhập vào bằng 0 thì phải thông báo "Dữ liệu không hợp lệ.".

Yêu cầu: Em hãy giúp Quỳnh viết chương trình giải phương trình bậc hai như trên.

Dữ liệu vào: Các số thực a, b, c.

Dữ liệu ra: Nếu nhập a = 0 thì thông báo ra màn hình "Dữ liệu không hợp lệ.". Ngược lại, giải phương trình và in kết quả ra màn hình.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập hệ số a, b, c: 0 1 2`	`Dữ liệu không hợp lệ.`
`Nhập hệ số a, b, c: 1 2 1`	`Phương trình có nghiệm kép: -1.0`
`Nhập hệ số a, b, c: 1 3 2`	`Phương trình có 2 nghiệm: -1.0 -2.0`
`Nhập hệ số a, b, c: 1 2 3`	`Phương trình vô nghiệm.`

Hướng dẫn:

Nhập các hệ số a, b, c.

Nếu a = 0 thì thông báo "Dữ liệu không hợp lệ.".

Ngược lại, giải phương trình: Tính $\Delta = b^2 - 4ac$. Nếu $\Delta < 0$ thì phương trình vô nghiệm. Nếu $\Delta = 0$ thì phương trình có nghiệm kép $x_1 = x_2 = -\frac{b}{2a}$. Nếu $\Delta > 0$ thì phương trình có 2 nghiệm: $x_1 = \frac{-b+\sqrt{\Delta}}{2a}$, $x_2 = \frac{-b-\sqrt{\Delta}}{2a}$.

BÀI F7 LỆNH LẶP

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết được cấu trúc lặp có số lần lặp xác định trước và lệnh lặp for.
- Biết được cấu trúc lặp có số lần lặp chưa xác định trước và lệnh lặp while.
- Sử dụng được cấu trúc lặp để thiết kế thuật toán và các lệnh lặp for và while để viết chương trình Python.

KHỞI ĐỘNG

Để tính giai thừa của số nguyên n > 1 (kí hiệu n!) theo công thức

$$n! = n \times (n-1) \times (n-2) \times \ldots \times 2 \times 1,$$

em phải thực hiện việc nhân lần lượt từng số nguyên từ 1 đến n. Để tìm ước số chung lớn nhất của hai số tự nhiên a, b, em có thể thay thế giá trị số lớn hơn trong hai số bởi hiệu của hai số, tiếp tục thực hiện việc thay thế cho đến khi được hai số bằng nhau cũng chính là giá trị ước chung lớn nhất của hai số tự nhiên a, b. Em hãy tìm một bài toán trong đó việc giải bài toán yêu cầu phải thực hiện một cách lặp đi lặp lại nhiều lần một thao tác.

KHÁM PHÁ

Các ngôn ngữ lập trình bậc cao đều cung cấp một số lệnh dùng để thực hiện các thao tác được lặp lại nhiều lần, tạo thành một cấu trúc lập trình và được gọi là cấu trúc lặp. Như vậy, cấu trúc lặp là một khối lệnh bao gồm một hoặc nhiều lệnh được thực hiện lặp đi lặp lại nhiều lần trong khi điều kiện xác định trước vẫn còn thoả mãn.

Có hai loại cấu trúc lặp: lặp có số lần biết trước và lặp có số lần chưa biết trước. Trong bài toán tính giai thừa của một số nguyên n > 1, câu lệnh thực hiện phép nhân được lặp đi lặp lại, trong mỗi bước lặp, thừa số nhân bổ sung được lấy trong dãy các số từ 1 đến n và kết thúc quá trình lặp khi không còn giá trị để lấy (tổng cộng lặp n lần). Trong trường hợp này, số lần lặp là biết trước. Trong bài toán tìm ước chung lớn nhất của hai số tự nhiên a, b, việc thay thế số lớn hơn trong hai số bởi hiệu của hai số được lặp đi lặp lại cho đến khi đạt được hai số bằng nhau. Trong trường hợp này, số lần lặp là chưa biết trước.

Như vậy, cấu trúc lặp với số lần biết trước sử dụng biến lặp để kiểm soát số lần lặp, giá trị của biến lặp ở mỗi bước lặp được lấy từ dãy các giá trị cho trước, quá trình lặp kết thúc khi biến lặp không còn giá trị để lấy. Cấu trúc lặp với số lần chưa biết trước sử dụng một biểu thức logic để thể hiện điều kiện lặp, khi biểu thức logic không còn được thoả mãn thì kết thúc quá trình lặp.

Ngôn ngữ lập trình Python sử dụng lệnh `for` và lệnh `while` để thể hiện cấu trúc lặp với số lần biết trước và lặp với số lần chưa biết trước.

1. Lệnh for

a) Lệnh for sử dụng hàm range()

Tích hợp sẵn trong Python, hàm `range()` được sử dụng để tạo một dãy các số nguyên trong phạm vi cho trước với cú pháp như sau:

```
range([<bắt đầu>,] <kết thúc>[, <bước nhảy>])
```

Trong đó, `<bắt đầu>`, `<kết thúc>`, `<bước nhảy>` là các biểu thức có giá trị nguyên; `<kết thúc>` là đối số bắt buộc phải có và không nằm trong dãy các số nguyên được tạo ra; `<bắt đầu>` có thể có hoặc không, nếu không thì mặc định là 0; `<bước nhảy>` có thể có hoặc không, nếu không thì mặc định là 1. Tuỳ theo số lượng đối số, hàm `range()` sẽ tự động chọn lựa dạng biểu thức và tạo ra vùng giá trị tương ứng.

Ví dụ,

+ `range(5)` tạo ra vùng giá trị là `0, 1, 2, 3, 4`. Vùng giá trị của hàm `range(5)` kết thúc tại `5` nhưng không bao gồm `5`, bắt đầu từ `0` và khoảng cách giữa hai số là `1`.

+ `range(1,5)` tạo ra vùng giá trị là `1, 2, 3, 4`. Vùng giá trị của hàm `range(1,5)` kết thúc tại `5` nhưng không bao gồm `5`, bắt đầu từ `1` và bước nhảy là `1`.

+ `range(5,1)` tạo ra vùng giá trị rỗng vì giá trị bắt đầu `5` lớn hơn giá trị kết thúc `1` với bước nhảy bằng `1`.

+ `range(5,1,-2)` tạo ra vùng giá trị là `5, 3`. Vùng giá trị của hàm `range(5,1,-2)` kết thúc tại `1` nhưng không bao gồm `1`, bắt đầu từ `5` và bước nhảy là `-2`.

Cú pháp của lệnh `for` sử dụng hàm `range()` như sau:

```
for <biến lặp> in range([<bắt đầu>,]<kết thúc>[,<bước nhảy>]):
    <khối lệnh>
```

Khi thực hiện lệnh lặp, ở mỗi vòng lặp, `<biến lặp>` sẽ lần lượt được gán các giá trị thuộc vùng giá trị được tạo ra bởi hàm `range()` và thực hiện `<khối lệnh>`. `<khối lệnh>`, còn được gọi là thân vòng lặp, bao gồm một hoặc nhiều câu lệnh. Vòng lặp kết thúc khi không còn giá trị thuộc vùng giá trị để gán cho `<biến lặp>`.

Ví dụ 1: Chương trình tính tổng các số tự nhiên từ 0 đến n.

```
1 n = int(input("Nhập số tự nhiên n: "))
2 tong = 0              # Khởi tạo biến tong và gán bằng 0
3 for i in range(n+1): # i nhận giá trị từ 0 đến n
4     tong = tong+i     # Cộng dồn i vào tong
5 print(f"Tổng các số từ 0 đến {n} là {tong}.")
```

Dòng lệnh 1 nhập giá trị của số tự nhiên n. Dòng lệnh 2 khởi tạo biến tong và gán giá trị bằng 0. Khi thực hiện lệnh lặp ở dòng lệnh 3, biến lặp i lần lượt được gán các giá trị thuộc vùng giá trị được tạo bởi hàm range(n+1) là các số tự nhiên từ 0 đến n, sau đó thực hiện lệnh cộng dồn vào tong ở dòng lệnh 4. Vòng lặp kết thúc khi biến lặp i đã được gán hết toàn bộ n giá trị thuộc vùng giá trị được tạo bởi hàm range(n+1), chuyển sang dòng lệnh 5 để in thông báo giá trị của tổng các số từ 0 đến n ra màn hình.

Ví dụ 2: Chương trình đếm các số tự nhiên lẻ nhỏ hơn n.

```
1 n = int(input("Nhập số tự nhiên n: "))
2 dem = 0                  # Khởi tạo biến dem và gán bằng 0
3 for i in range(1,n,2):   # i nhận các số lẻ < n, bắt đầu từ 1
4     dem = dem+1          # Tăng giá trị dem lên 1
5 print(f"Có {dem} số tự nhiên lẻ nhỏ hơn {n}.")
```

Dòng lệnh 1 nhập giá trị của số tự nhiên n. Dòng lệnh 2 khởi tạo biến dem và gán giá trị bằng 0. Khi thực hiện lệnh lặp ở dòng lệnh 3, biến lặp i lần lượt được gán các giá trị thuộc vùng giá trị được tạo bởi hàm range(1,n,2) là các số lẻ nhỏ hơn n, bắt đầu từ 1, cách nhau 2 đơn vị, sau đó thực hiện việc tăng giá trị của dem lên 1 đơn vị. Sau khi kết thúc vòng lặp, chuyển sang dòng lệnh 5 để in ra thông báo về số các số tự nhiên lẻ nhỏ hơn n.

Trong trường hợp vùng giá trị của hàm range() là rỗng, không thực hiện thân vòng lặp và chương trình chuyển qua thực hiện dòng lệnh kế tiếp. Ví dụ, trong hai chương trình ở Ví dụ 1 và Ví dụ 2, nếu n < 0, vùng giá trị của hàm range() tương ứng ở hai ví dụ đều là rỗng, lệnh for ở dòng lệnh 3 sẽ không thực hiện thân vòng lặp ở dòng lệnh 4, chương trình chuyển qua thực hiện dòng lệnh 5 (giá trị của các biến tong và dem đều bằng giá trị khởi tạo là 0).

Lưu ý: Trong thân vòng lặp for, không nên có câu lệnh làm thay đổi giá trị của biến lặp.

b) Lệnh for sử dụng bộ dữ liệu

Cú pháp của lệnh for sử dụng bộ dữ liệu như sau:

```
for <biến lặp> in (<phần tử 1>, <phần tử 2>,..., <phần tử N>):
    <khối lệnh>
```

Trong đó,

– <khối lệnh> bao gồm một hoặc nhiều lệnh.

– (<phần tử 1>, <phần tử 2>,..., <phần tử N>) là bộ dữ liệu bao gồm các phần tử thuộc bất kì kiểu dữ liệu nào và có thể trùng nhau.

– <biến lặp> lần lượt lấy các phần tử từ <phần tử 1> đến <phần tử N> cho mỗi bước lặp.

Ví dụ 3: Chương trình sau đây hiển thị các tên mùa trong năm.

```
1 for tenMua in ("Xuân", "Hạ", "Thu", "Đông"):
2     print(tenMua)
```

1. Viết chương trình nhập số nguyên dương n, hiển thị các ước số của n ra màn hình.

2. Viết chương trình nhập số nguyên dương n, hiển thị các số nhỏ hơn hoặc bằng n và chia hết cho 3 ra màn hình.

Biến lặp của vòng lặp `for` sẽ lần lượt lấy các giá trị từ hàm `range()` hoặc bộ dữ liệu.

2. Lệnh while

Cú pháp của lệnh while như sau:

```
while <điều kiện>:
    <khối lệnh>
```

Trong đó, `<điều kiện>` là một biểu thức logic, được gọi là điều kiện lặp của vòng lặp `while`. `<khối lệnh>` được gọi là thân vòng lặp, bao gồm một hoặc nhiều lệnh. Khi thực hiện lệnh `while`, Python sẽ kiểm tra `<điều kiện>`, chừng nào `<điều kiện>` còn nhận giá trị là True thì còn thực hiện `<khối lệnh>`. Nếu `<điều kiện>` nhận giá trị là False thì sẽ kết thúc lệnh `while`.

Ví dụ 4: Chương trình sau đây nhập số tự nhiên n và hiển thị các số Fibonacci nhỏ hơn hoặc bằng n. Biết rằng dãy số Fibonacci có tính chất $f_0 = 0$, $f_1 = 1$, $f_n = f_{n-1} + f_{n-2}$ với $n \geq 2$.

```
1 n = int(input("Nhập số tự nhiên n: "))
2 print(f"Các số Fibonacci nhỏ hơn hoặc bằng {n}:")
3 a = 0
4 b = 1
5 while a <= n:
6     print(a,end = " ")
7     c = a+b
8     a = b
9     b = c
```

Nhập số tự nhiên `n` (dòng lệnh 1). Gán `a = 0` và `b = 1` là hai số Fibonacci đầu tiên (các dòng lệnh 3, 4). Số Fibonacci đang xét là `a`. Vòng lặp while để kiểm tra và in số Fibonacci (các dòng lệnh 5 – 9). Nếu `a ≤ n` thì hiển thị `a` (dòng lệnh 6), tính `c` là số Fibonacci liền sau `b` (dòng lệnh 7), rồi gán `b` cho `a` và gán `c` cho `b` (các dòng lệnh 8, 9).

Ví dụ 5: Quan sát chương trình ở *Hình 1* và cho biết số vòng lặp được thực hiện.

```
1 i = 1
2 while i <= 5:
3     print(i)
```

Hình 1. Minh hoạ vòng lặp while

Dòng lệnh 1 khởi tạo biến `i` có giá trị bằng `1`. Khi đó, điều kiện `i ≤ 5` của vòng lặp `while` ở dòng lệnh 2 là True, nên thực hiện vòng lặp và in ra giá trị của biến `i` ở dòng lệnh 3, tiếp tục quay về dòng lệnh 2 để kiểm tra điều kiện `i ≤ 5`. Vì trong vòng lặp không có lệnh làm thay đổi giá trị của biến `i` (biến `i` luôn luôn bằng `1`) nên điều kiện `i ≤ 5` luôn luôn đúng và kết quả là vòng lặp bị lặp vô hạn.

Trong chương trình ở *Hình 2*, dòng lệnh 4 sẽ tăng giá trị của biến `i` lên `1` và quá trình lặp được tiếp tục cho đến khi giá trị của biến `i` là `6`. Khi đó, điều kiện của vòng lặp `i ≤ 5` là False và kết thúc quá trình lặp:

```
1 i = 1
2 while i <= 5:
3     print(i)
4     i = i + 1
```

Hình 2. Minh hoạ vòng lặp while

Lưu ý: Để tránh vòng lặp `while` bị lặp vô hạn, thân vòng lặp phải có một lệnh được thực hiện và lệnh này làm thay đổi giá trị của `<điều kiện>` từ True thành False để kết thúc quá trình lặp.

Viết câu lệnh nhập số tự nhiên `n` và hiển thị số chữ số của `n` ra màn hình. Ví dụ, số chữ số của 0; 9; 15; 152; 1 000 là 1; 1; 2; 3; 4 theo thứ tự tương ứng.

- Số lần lặp của lệnh `while` phụ thuộc vào điều kiện của lệnh.
- Để tránh vòng lặp vô hạn, thân vòng lặp phải có một lệnh được thực hiện và lệnh này làm thay đổi giá trị của điều kiện lặp từ True thành False để kết thúc quá trình lặp.

3. Lệnh break và continue

Lệnh `break` và `continue` là hai lệnh dùng để làm thay đổi việc thực hiện bình thường của vòng lặp `for` và vòng lặp `while`. Lệnh `break` dùng để thoát khỏi vòng lặp và lệnh `continue` dùng để bỏ qua các lệnh còn lại của vòng lặp hiện tại và quay trở lại kiểm tra điều kiện của vòng lặp.

a) Lệnh break

Lệnh `break` dùng để thoát khỏi vòng lặp ngay lập tức, nghĩa là *kết thúc quá trình lặp một cách bất thường*. Lệnh `break` ở trong thân vòng lặp của các vòng lặp `for` và `while`. Thông thường, lệnh `break` nằm trong lệnh `if` và có cú pháp như sau:

```
if <điều kiện>:
    break
```

Trong đó, `<điều kiện>` là một biểu thức logic dùng để thoát khỏi vòng lặp.

Ví dụ 6: Chương trình sau đây nhập số nguyên `n > 1` và hiển thị số lớn nhất trong các ước nhỏ hơn `n` ra màn hình.

```
1 n = int(input("Nhập số nguyên n > 1: "))
2 if n > 1:                 # n > 1
3     for k in range(n//2,0,-1): # Lặp k từ n // 2 giảm về 1
4         if n % k == 0: # n chia hết cho k
5             break      # Tìm thấy ULN, kết thúc vòng lặp
6     print(f"Số lớn nhất trong các ước nhỏ hơn {n} là {k}.")
7 else:                     # n <= 1
8     print("n phải lớn hơn 1.")
```

b) Lệnh continue

Lệnh `continue` dùng để bỏ qua các lệnh còn lại của vòng lặp hiện tại và quay trở lại kiểm tra điều kiện của vòng lặp. Thông thường, lệnh `continue` nằm trong lệnh `if` và có cú pháp như sau:

```
if <điều kiện>:
    continue
```

Trong đó, `<điều kiện>` là một biểu thức logic dùng để bỏ qua các lệnh còn lại của vòng lặp hiện tại và quay trở lại kiểm tra điều kiện của vòng lặp.

Ví dụ 7: Chương trình sau đây hiển thị các số từ 1 đến 5, ngoại trừ số 4.

```
1 for i in range(6):          # Biến lặp i từ 0 đến 5
2     if i == 4:              # i = 4
3         continue            # Không hiển thị khi i = 4
4     print(i)                # Hiển thị i
```

> Lệnh `break` dùng để thoát khỏi vòng lặp và lệnh `continue` dùng để bỏ qua các lệnh còn lại của vòng lặp hiện tại và quay trở lại kiểm tra điều kiện của vòng lặp. Thông thường, các lệnh `break` và `continue` nằm trong lệnh `if` và ở trong thân vòng lặp của các vòng lặp `for` và `while`.

LUYỆN TẬP

1. Em hãy viết lại chương trình trong Ví dụ 5, sử dụng cấu trúc lặp `for`.
2. Viết chương trình theo hai cách (dùng vòng lặp `for` và vòng lặp `while`) để nhập số nguyên dương n, tính và hiển thị tổng số S như sau:

$$S = 1^2 + 2^2 + 3^2 + \ldots + n^2.$$

3. Viết chương trình nhập hai số nguyên dương n và k với $k < n$. Tìm số nguyên m với điều kiện $k < m \leq n$ và m chia hết cho k. Nếu tìm thấy thì hiển thị m ra màn hình; ngược lại, hiển thị "Không tìm thấy.".

VẬN DỤNG

1. Viết chương trình nhập các số nguyên, tính và in ra màn hình tổng các số nguyên dương. Chương trình kết thúc khi số đã nhập là số nguyên âm.
2. Viết chương trình nhập các số nguyên n và hiển thị "Là số nguyên tố." ra màn hình nếu n là số nguyên tố; ngược lại, hiển thị "Không là số nguyên tố.". Chương trình kết thúc khi số đã nhập nhỏ hơn hoặc bằng 1.
3. Viết chương trình nhập số nguyên $n > 1$, in ra màn hình số nguyên tố nhỏ nhất lớn hơn hoặc bằng n.

Chân trời sáng tạo

BÀI F8 THỰC HÀNH LỆNH LẶP

MỤC TIÊU **Sau bài học này, em sẽ:**

- Sử dụng được các lệnh lặp để viết chương trình giải quyết một số bài toán đơn giản.

KHỞI ĐỘNG

Em hãy phân tích cấu trúc lặp của các tình huống sau, xác định là cấu trúc lặp có số lần lặp xác định trước hoặc chưa xác định trước và chỉ ra phần công việc lặp đi lặp lại của bài toán.

Tình huống 1: Nhập số nguyên dương n, in ra màn hình giá trị tổng bình phương n số nguyên lẻ đầu tiên.

Tình huống 2: Nhập số nguyên dương n, in ra màn hình số nguyên tố đầu tiên lớn hơn n.

THỰC HÀNH

Nhiệm vụ 1. Tính tổng bình phương của n số nguyên dương lẻ đầu tiên

Vinh nói với Khôi rằng tổng bình phương của n số nguyên dương lẻ đầu tiên được tính theo công thức: $1^2+3^2+\ldots+(2n-1)^2 = \frac{1}{3}n(4n^2-1)$. Khôi muốn kiểm tra xem công thức trên là đúng hay sai nên sẽ viết chương trình tính tổng bình phương của n số nguyên dương lẻ đầu tiên theo hai cách: tính tổng bình phương từng số nguyên lẻ dùng vòng lặp (vế trái) và tính tổng bình phương theo công thức (vế phải).

Yêu cầu: Viết chương trình tính tổng bình phương của n số nguyên dương lẻ đầu tiên theo hai cách: tính tổng bình phương từng số nguyên dương lẻ dùng vòng lặp (vế trái) và tính tổng bình phương theo công thức (vế phải).

Dữ liệu vào: Số nguyên dương n.

Dữ liệu ra: Tổng bình phương của n số nguyên dương lẻ đầu tiên, được tính theo hai cách.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị

`Nhập số nguyên dương n: 12`	`Tổng bình phương 12 số nguyên lẻ đầu tiên:`
	`Tính theo công thức vế trái: 2300`
	`Tính theo công thức vế phải: 2300.0`

Hướng dẫn: Sử dụng vòng lặp `for` với biến lặp lấy từ vùng giá trị gồm `n` số lẻ đầu tiên (bắt đầu từ 1) và tính tổng bình phương của các số lẻ này. Cụ thể như sau:

1. Nhập số nguyên dương `n`.
2. Khởi tạo `s = 0` là tổng cần tìm.
3. Tính tổng bình phương theo vế trái.

   Cho `i` từ `1` đến `n`:

   Cộng dồn `(2*i-1)**2` vào `s`

   In `s` ra màn hình.

4 Tính tổng bình phương theo vế phải.

In ra màn hình giá trị biểu thức `n*(4*n**2-1)/3`

Chương trình tham khảo:

```
n = int(input("Nhập số nguyên dương n: "))
s = 0
for i in range(1,n+1):
    s += (2*i-1)**2
print(f"Tổng bình phương {n} số nguyên lẻ đầu tiên:")
print(f" Tính theo công thức vế trái: {s}")
print(f" Tính theo công thức vế phải: {n*(4*n**2-1)/3}")
```

Nhiệm vụ 2. Nhập số nguyên dương n, in ra màn hình toàn bộ các số chính phương nhỏ hơn n

An đố Bình làm cách nào có thể liệt kê được hết tất cả các số chính phương nhỏ hơn số nguyên dương n cho trước. Bình viết chương trình in ra màn hình toàn bộ các số chính phương nhỏ hơn số nguyên dương n được nhập từ bàn phím để giải đáp lời thách đố của An.

Yêu cầu: Viết chương trình in ra màn hình toàn bộ các số chính phương nhỏ hơn số nguyên dương n được nhập từ bàn phím.

Dữ liệu vào: Số nguyên dương n.

Dữ liệu ra: Tất cả các số chính phương nhỏ hơn n.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
Nhập số nguyên dương n: 10	Các số chính phương nhỏ hơn 10 là: 0 1 4 9

Hướng dẫn: Sử dụng vòng lặp `while` để xét biến lặp `i` với `0 ≤ i < n`. Nếu $i^2 < n$ thì in giá trị bình phương của biến `i` ra màn hình; ngược lại, kết thúc quá trình lặp. Cụ thể như sau:

1 Nhập số nguyên dương `n`.

2 Khởi tạo `i = 0` là giá trị kiểm tra đầu tiên.

3 Tìm các số chính phương nhỏ hơn `n` và in ra màn hình.

Trong khi `i**2 < n` thì:

In `i**2` ra màn hình

Tăng `i` lên `1` đơn vị

Chương trình tham khảo:

```
n = int(input("Nhập số nguyên dương n: "))
i = 0            # Giá trị kiểm tra đầu tiên
print(f"Các số chính phương nhỏ hơn {n} là: ")
while i**2 < n:
    print(i**2,end = " ")
    i += 1       # Tăng i lên 1 đơn vị
```

Nhiệm vụ 3. Tìm ước chung lớn nhất của hai số tự nhiên

Yêu cầu: Viết chương trình in ra màn hình ước chung lớn nhất của hai số tự nhiên được nhập từ bàn phím, ứng dụng thuật toán Euclid.

Dữ liệu vào: Hai số tự nhiên a, b.

Dữ liệu ra: Ước chung lớn nhất của hai số tự nhiên a, b.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
Nhập số nguyên dương a: 198 Nhập số nguyên dương b: 54	Ước chung lớn nhất của 198 và 54 là 18

Chân trời sáng tạo

Ví dụ: Thuật toán Euclid để tìm ước chung lớn nhất của hai số tự nhiên a = 198, b = 54 được thực hiện như *Bảng 1*.

Thuật toán Euclid được mô tả như sau:

Bước 1: Khi b = 0, thì UCLN = a. Kết thúc.

Bước 2: Khi b > 0, tính số dư r của a chia cho b, tức là r = a%b.

Bước 3: Thay a = b và b = r. Quay lại Bước 1.

Bảng 1.

a	b	r = a%b
198	54	36
54	36	18
36	18	0
18	0 Vì b = 0 nên UCLN là a = 18	

Hướng dẫn: Sử dụng vòng lặp `while` để xét hai số `a` và `b`. Nếu `b = 0` thì UCLN là `a` và kết thúc quá trình lặp. Nếu `b > 0` thì tính số dư `r` của `a` chia cho `b`, rồi gán `b` cho `a`, gán `r` cho `b` và lặp lại vòng lặp. Cụ thể như sau:

1. Nhập vào hai số dương `a`, `b`.
2. Thực hiện thuật toán Euclid.

   Trong khi `b > 0` thì:

   Gán số dư của `a` chia `b` cho `r`.

   Gán `b` cho `a`.

   Gán `r` cho `b`.
2. In `a` (là ước chung lớn nhất của hai số `a`, `b` ban đầu) ra màn hình.

Chương trình tham khảo:

```
a = int(input("Nhập số nguyên dương a: "))
b = int(input("Nhập số nguyên dương b: "))
print(f"Ước chung lớn nhất của {a} và {b} là", end = " ")
while b > 0:
    r = a%b
    a = b
    b = r
print(a)
```

LUYỆN TẬP

1. Hoàn chỉnh chương trình của Nhiệm vụ 1 ở phần THỰC HÀNH, bổ sung thêm các yêu cầu sau:

+ Kiểm tra giá trị nhập, nếu không phải là số dương thì in ra thông báo "Giá trị nhập phải là số nguyên dương." và không thực hiện phần tính tổng.

+ Khi giá trị là số dương, in ra màn hình tổng bình phương các số nguyên lẻ với đầy đủ các số hạng. Ví dụ, với giá trị n nhập từ bàn phím là 5, chương trình sẽ in ra màn hình nội dung "1^2+3^2+5^2+7^2+9^2=165".

2. Hoàn chỉnh chương trình của Nhiệm vụ 2 ở phần THỰC HÀNH, bổ sung thêm các yêu cầu sau:

+ Kiểm tra giá trị nhập, nếu không phải là số dương thì in ra màn hình thông báo "Giá trị nhập phải là số nguyên dương." và không thực hiện phần tìm số chính phương.

+ Khi giá trị nhập là số dương, đếm số chính phương tìm được và bổ sung trong thông báo in ra màn hình. Ví dụ, với giá trị n được nhập từ bàn phím là 5, in ra màn hình thông báo "Có 3 số chính phương nhỏ hơn 5 là: 0 1 4".

Chân trời sáng tạo

3. Hoàn chỉnh chương trình của Nhiệm vụ 3 ở phần THỰC HÀNH, bổ sung thêm các yêu cầu sau:

+ Kiểm tra giá trị nhập, nếu `a < 0` hoặc `b < 0` hoặc `a == b == 0` thì in ra màn hình thông báo "Số đã nhập không hợp lệ.", không thực hiện phần tìm ước chung lớn nhất.

+ Khi giá trị nhập là hợp lệ, nếu ước chung lớn nhất là 1 thì in thêm thông báo "Hai số nguyên tố cùng nhau.".

VẬN DỤNG

1. Viết chương trình tính giai thừa của số tự nhiên n được nhập từ bàn phím.

Yêu cầu: Nhập số tự nhiên n từ bàn phím, in ra màn hình công thức tính giai thừa của số tự nhiên n và giá trị của giai thừa (lưu ý, 0! = 1).

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập số tự nhiên n: 5`	`5! = 1*2*3*4*5 = 120`

2. Viết chương trình tìm tất cả các số Amrstrong có số chữ số n được nhập từ bàn phím ($n \le 6$). Biết rằng số tự nhiên a có n chữ số là số Armstrong nếu a bằng tổng luỹ thừa bậc n các chữ số của a.

Yêu cầu: Nhập số nguyên dương $n \le 6$ từ bàn phím, in ra màn hình tất cả các số Armstrong có n chữ số.

Dữ liệu vào: Số nguyên dương $n \le 6$.

Dữ liệu ra: Số các số Armstrong có n chữ số tìm được và liệt kê tất cả các số Armstrong này (nếu có).

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập số tự nhiên n: 3`	`Có 4 số Armstrong có 3 chữ số:` `153 370 371 407`

Gợi ý: Sử dụng vòng lặp `for` với biến lặp `a` lấy trong vùng giá trị từ `10**(n-1)` đến `10**n-1`. Để kiểm tra `a` là số Armstrong, sử dụng phép chia và phép lấy số dư của `a` với `10` để có được tất cả chữ số của `a`. Nếu `a` bằng tổng của các luỹ thừa bậc `n` của tất cả chữ số thì `a` là số Armstrong và tăng biến đếm lên 1 (ban đầu, biến đếm bằng 0).

3. Viết chương trình giải mã số bí mật có 3 chữ số.

Yêu cầu: Nhập số nguyên dương n có 3 chữ số, in ra màn hình thông báo so sánh giá trị của n với số bí mật (được đặt sẵn từ đầu trong chương trình). Nếu trong 10 lần nhập tìm ra được số bí mật của chương trình thì in ra màn hình thông báo "Chúc mừng, bạn đã giải mã được số bí mật." và kết thúc chương trình. Ngược lại thì in ra màn hình thông báo "Bạn đã thất bại trong việc giải mã số bí mật." và kết thúc chương trình.

Dữ liệu vào: Số nguyên dương n.

Dữ liệu ra: Thông báo so sánh giá trị của n với số bí mật. Nếu trong 10 lần nhập, n bằng với số bí mật thì thông báo chúc mừng. Nếu không nhập được giá trị n bằng với số bí mật trong 10 lần nhập thì thông báo thất bại.

BÀI F9 XÂU KÍ TỰ

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết được xâu kí tự và cách biểu diễn xâu kí tự trong Python.
- Sử dụng được một số hàm xử lí xâu kí tự trong Python.

KHỞI ĐỘNG

Quan sát câu lệnh dưới đây, em hãy cho biết xâu a có bao nhiêu kí tự? Có cách nào để đưa ra số lượng kí tự của xâu a mà không cần phải đếm?

```
>>> a = "Tin học 10"
```

KHÁM PHÁ

1. Xâu kí tự

Xâu kí tự (được gọi tắt là xâu) là một dãy các kí tự liên tiếp có thể là chữ cái, chữ số, kí tự đặc biệt. Ví dụ, "Chào bạn, bạn đang học sách Tin học 10, Bộ sách Chân trời sáng tạo" là một xâu kí tự gồm các chữ cái, chữ số và kí tự đặc biệt (khoảng trắng, dấu phẩy).

Chiều dài của xâu là số kí tự của xâu. Xâu rỗng (xâu không có kí tự nào) có chiều dài bằng 0. Xâu kí tự trong Python là dãy các kí tự Unicode và được đặt trong cặp nháy kép ("…") hoặc cặp nháy đơn ('…'). Trong chương trình ở hoạt động KHỞI ĐỘNG có một biến a thuộc kiểu xâu kí tự.

Lưu ý: Ngôn ngữ Python cho phép một xâu kí tự được ghi trên nhiều dòng lệnh, được đặt trong cặp ba nháy kép ("""…""") hoặc cặp ba nháy đơn ('''…''').

Ví dụ dưới đây tạo biến c là một xâu kí tự được ghi trên nhiều dòng.

```
c = """Một số ngôn ngữ lập trình thông dụng:
- JavaScript
- Python
- Java
- C#
..."""
print(c)
```

Trong Python, hàm `len(s)` trả về chiều dài của xâu `s`. Có 2 cách đánh chỉ số cho một xâu `s` có chiều dài `n`:

+ Chỉ số dương: đánh từ trái qua phải từ `0` đến `n - 1`.

+ Chỉ số âm: đánh từ phải qua trái từ `-1` về `-n`.

Kí tự có chỉ số `i` của xâu `s` được kí hiệu là `s[i]`.

Quan sát *Hình 1*, xâu `s = "Tin-học"` có chiều dài 7 được đánh chỉ số theo hai cách (sử dụng chỉ số dương hoặc sử dụng chỉ số âm). Kí tự `s[0]` và `s[-7]` đều là `"T"`, kí tự `s[6]` và `s[-1]` đều là `"c"`.

Xâu s	T	i	n	-	h	ọ	c
Chỉ số dương	0	1	2	3	4	5	6
Chỉ số âm	-7	-6	-5	-4	-3	-2	-1

Hình 1. Biểu diễn xâu s = "Tin-học"

Em hãy quan sát chương trình sau và cho biết kết quả in ra màn hình là gì?

```
s = "Chân Trời Sáng Tạo"
print(f"Số kí tự của xâu '{s}' là: {len(s)}")
for i in range(int(len(s)/2)):
    print(s[i],end=" ")
```

- Xâu kí tự là một dãy các kí tự liên tiếp nhau. Các kí tự có thể là chữ cái, chữ số và kí tự đặc biệt.
- Trong ngôn ngữ Python, các kí tự của xâu được đặt trong cặp nháy kép("...") hoặc cặp nháy đơn ('...').
- Các kí tự của xâu `s` được truy cập theo chỉ số. Kí tự có chỉ số `i` của xâu `s` được kí hiệu là `s[i]`.

2. Xử lí xâu kí tự trong Python

Ngôn ngữ lập trình Python cung cấp một số phép toán và hàm để thực hiện các thao tác về xâu kí tự như: kết nối các xâu thành một xâu mới, tạo xâu con từ một xâu, tìm kiếm xâu con trong một xâu, đếm số lần xuất hiện xâu con trong một xâu,...

a) Kết nối xâu

Để kết nối các xâu với nhau trong Python, ta có thể sử dụng toán tử +.

```
>>> a = "Công nghệ"
>>> b = "thông tin"
>>> a+" "+b
'Công nghệ thông tin'
```

Trong các lệnh trên, xâu `a` kết nối xâu " " rồi kết nối xâu `b` bằng toán tử + và cho ra kết quả là xâu "Công nghệ thông tin".

b) Tạo xâu con từ một xâu

Xâu con của một xâu gồm các kí tự liên tiếp hoặc không liên tiếp của xâu đó. Trong Python, lát cắt xâu với cú pháp `a[m:n]` dùng để tạo xâu con từ xâu `a`, gồm các kí tự từ chỉ số `m` đến chỉ số `n-1`. Nếu không có `m` thì `m` là `0`, nếu không có `n` thì `n` là số kí tự của xâu `a`.

Ví dụ 1: Cho xâu `a = "Tin học 10"` (*Hình 2*). Sau đây là các ví dụ về lát cắt của xâu `a`:

```
>>> a = "Tin học 10"
>>> a[:]
'Tin học 10'
>>> a[:3]
'Tin'
>>> a[4:]
'học 10'
>>> a[4:7]
'học'
```

Xâu a	T	i	n		h	ọ	c		1	0
Chỉ số	0	1	2	3	4	5	6	7	8	9

Hình 2. Biểu diễn xâu a = "Tin học 10"

Chân trời sáng tạo

+ Lát cắt `a[:]` tạo ra xâu con bằng toàn bộ xâu `a`.

+ Lát cắt `a[:3]` tạo ra xâu con gồm 3 kí tự đầu tiên của xâu `a` (từ chỉ số 0 đến trước chỉ số 3).

+ Lát cắt `a[4:]` tạo ra xâu con gồm các kí tự của xâu `a` tính từ chỉ số 4.

+ Lát cắt `a[4:7]` tạo ra xâu con gồm các kí tự của xâu `a` tính từ chỉ số 4 đến trước chỉ số 7.

Trong Python, lát cắt xâu với cú pháp `a[m:n:p]` dùng để tạo xâu con từ xâu `a` bằng cách chọn lựa các kí tự trong xâu `a` có chỉ số nhỏ hơn `n`, cách nhau một bước nhảy `p` (với `p` là số nguyên khác `0`), bắt đầu từ chỉ số `m`. Nếu không có `m` thì `m` là `0`, nếu không có `n` thì `n` là số kí tự của xâu `a`.

Ví dụ 2: Cho xâu `a = "Chân Trời Sáng Tạo"` (*Hình 3*).

Xâu a	C	h	â	n		T	r	ờ	i		S	á	n	g		T	ạ	o
Chỉ số	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Hình 3. Biểu diễn xâu a = "Chân Trời Sáng Tạo"

Các lệnh bên là các ví dụ về lát cắt của xâu `a`:

```
>>> a = "Chân Trời Sáng Tạo"
>>> a[::5]
'CTST'
>>> a[:10:5]
'CT'
>>> a[10::5]
'ST'
```

– Lát cắt `a[::5]` tạo ra một xâu bao gồm các kí tự trong xâu `a` có chỉ số cách nhau `5`, bắt đầu từ chỉ số `0` (các chỉ số `0`, `5`, `10`, `15` tương ứng với các kí tự `"C"`, `"T"`, `"S"`, `"T"`).

– Lát cắt `a[:10:5]` tạo ra một xâu bao gồm các kí tự trong xâu `a` có chỉ số cách nhau `5` và nhỏ hơn `10`, bắt đầu từ chỉ số `0` (các chỉ số `0`, `5` tương ứng với các kí tự `"C"`, `"T"`).

– Lát cắt `a[10::5]` tạo ra một xâu bao gồm các kí tự trong xâu `a` có chỉ số cách nhau `5`, bắt đầu từ chỉ số `10` (các chỉ số `10`, `15` tương ứng với các kí tự `"S"`, `"T"`).

Bên cạnh lát cắt, trong Python, ta có thể tách một xâu thành nhiều xâu con bằng cách sử dụng lệnh `split()`. Mặc định lệnh `split()` sẽ dùng các khoảng trắng để làm căn cứ tách xâu thành các xâu con.

Ví dụ 3: Cho xâu `s = 'Tin học - Lớp 10'`. Hàm `s.split()` tách xâu `s` thành các xâu con phân cách nhau bởi khoảng trắng gồm `'Tin'`, `'học'`, `'-'`, `'Lớp'`, `'10'`. Hàm `s.split('-')` tách xâu `s` thành các xâu con phân cách nhau bởi kí tự `'-'` gồm `'Tin học '`, `' Lớp 10'`.

```
>>> s = 'Tin học - Lớp 10'
>>> s.split()
['Tin', 'học', '-', 'Lớp', '10']
>>> a,b = s.split('-')
>>> a
'Tin học '
>>> b
' Lớp 10'
```

c) Tìm kiếm xâu con trong xâu

Trong Python, để kiểm tra sự xuất hiện của một xâu con trong một xâu, có thể sử dụng phép toán `in` trả về giá trị True nếu tìm thấy, False nếu không tìm thấy và phép toán `not in` trả về giá trị True nếu không tìm thấy, False nếu tìm thấy.

Ví dụ 4: Trong các lệnh bên, vì Python phân biệt chữ hoa và chữ thường, kết quả trả về của `a in c` là True (nghĩa là tìm thấy xâu `a` trong xâu `c`), kết quả trả về của `b not in c` là True (nghĩa là không tìm thấy xâu `b` trong xâu `c`).

```
>>> a = "Khoa học"
>>> b = "khoa học"
>>> c = "Khoa học máy tính"
>>> a in c
True
>>> b not in c
True
```

Ngoài ra, Python cung cấp hàm `find()` trả về vị trí (chỉ số) xuất hiện đầu tiên của một xâu trong một xâu khác.

Ví dụ 5: Cho xâu `a = "Thứ"`, `b = "Thứ hai, Thứ ba, Thứ"` (*Hình 4*).

Xâu b	T	h	ứ		h	a	i	,		T	h	ứ		b	a	,		T	h	ứ
Chỉ số	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Hình 4. Biểu diễn xâu b = "Thứ hai, Thứ ba, Thứ"

```
>>> a = "Thứ"
>>> b = "Thứ hai, Thứ ba, Thứ"
>>> a in b
True
>>> b.find(a)
0
>>> b.find("A")
-1
>>> b.find(a,3)
9
>>> b.find(a,3,10)
-1
```

– Lệnh `b.find(a)` trả về vị trí xuất hiện đầu tiên của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số `0`, kết quả là `0`.

– Lệnh `b.find("A")` trả về vị trí xuất hiện đầu tiên của xâu `"A"` trong xâu `b`, bắt đầu tìm từ chỉ số `0`, kết quả là `-1` vì không tìm thấy.

– Lệnh `b.find(a,3)` trả về vị trí xuất hiện đầu tiên của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số `3`, kết quả là `9`.

– Lệnh `b.find(a,3,10)` trả về vị trí xuất hiện đầu tiên của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số `3` đến trước chỉ số `10`, kết quả là `-1` vì không tìm thấy (xâu `a` không xuất hiện trọn vẹn trong khu vực từ chỉ số `3` đến trước chỉ số `10` của xâu `b`).

d) Đếm số lần xuất hiện xâu con

Trong Python, để đếm số lần xâu con xuất hiện trong một xâu, ta có thể sử dụng hàm `count()`.

Ví dụ 6: Với 2 xâu `a`, `b` được nhắc đến ở Ví dụ 5.

```
>>> a = "Thứ"
>>> b = "Thứ hai, Thứ ba, Thứ"
>>> b.count(a)
3
>>> b.count(a,15)
1
>>> b.count(a,5,15)
1
```

– Lệnh `b.count(a)` trả về số lần xuất hiện của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số 0, kết quả là 3 (xâu `a` xuất hiện 3 lần trong xâu `b`).

– Lệnh `b.count(a,15)` trả về số lần xuất hiện của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số 15, kết quả là 1 (xâu `a` xuất hiện 1 lần từ chỉ số 15 trong xâu `b`).

– Lệnh `b.count(a,5,15)` trả về số lần xuất hiện của xâu `a` trong xâu `b`, bắt đầu tìm từ chỉ số 5 đến trước chỉ số 15, kết quả là 1 (xâu `a` xuất hiện 1 lần từ chỉ số 5 đến trước chỉ số 15 trong xâu `b`).

1. Em hãy quan sát chương trình sau và cho biết kết quả in ra màn hình là gì?

```
1 a = "Tin Học Ứng Dụng"
2 b = "Tin Học 10"
3 print(b,a)
4 print(b+a)
5 print(b,"-",a[8:])
6 print(b,"-",a[::4])
```

2. Cho hai xâu `a = "Công Nghệ Thông Tin"`, `b = "Tin học"`. Em hãy cho biết kết quả của các câu lệnh sau:

a) `a.count(b[:3])`

b) `b.find(a[-3:])`

Một số công cụ xử lí xâu thường được sử dụng trong Python:

- Phép toán + để kết nối các xâu.
- Lát cắt `a[m:n:p]` để tạo một xâu từ xâu `a`.
- Phép toán `in` và `not in` để kiểm tra sự xuất hiện của xâu con trong một xâu.
- Hàm `find()` trả về vị trí xuất hiện đầu tiên của xâu con trong một xâu.
- Hàm `count()` trả về số lần xuất hiện của xâu con trong một xâu.

LUYỆN TẬP

Em hãy viết chương trình nhập xâu "Em học sách Tin học 10, Định hướng Tin học ứng dụng", gán cho biến `a`, sau đó thực hiện các yêu cầu sau:

a) Đếm số từ trong xâu `a` với nhận xét rằng số từ trong xâu `a` nhiều hơn số khoảng trắng " " là 1.

b) Tạo xâu con `b` có nội dung "Tin học" được cắt từ xâu `a`.

c) Đếm số lần xuất hiện của xâu `b` trong xâu `a` và tìm kiếm tất cả các vị trí xuất hiện của xâu `b` trong xâu `a`.

VẬN DỤNG

Em hãy viết chương trình nhập họ và tên của một học sinh bao gồm các chữ cái và khoảng trắng (không có khoảng trắng ở đầu và cuối xâu được nhập), in ra màn hình thông tin về tên, họ và tên lót của học sinh. Biết rằng tên là từ cuối cùng, họ và tên lót là phần còn lại của họ và tên sau khi bỏ đi tên.

Dữ liệu vào: Xâu `hoVaTen` của học sinh, không có khoảng trắng ở đầu và cuối xâu.

Dữ liệu ra: Xâu `ten` và xâu `hoVaLot`.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập họ và tên:` `Nguyễn Trần Anh Thư`	`Tên: Thư` `Họ và tên lót: Nguyễn Trần Anh`

Hướng dẫn: Sử dụng vòng lặp để kiểm tra từng kí tự của xâu `hoVaTen` được nhập từ bàn phím, duyệt theo vị trí từ cuối xâu về đầu xâu. Kí tự khoảng trắng đầu tiên tìm được trong xâu `hoVaTen` đánh dấu vị trí phân cách giữa xâu `ten` với xâu `hoVaLot`.

BÀI F10 THỰC HÀNH XỬ LÍ XÂU

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết và thực hiện được một số lệnh thường dùng với xâu kí tự trong Python.

KHỞI ĐỘNG

Xâu kí tự là một phần quan trọng trong cuộc sống hằng ngày, là cách để chúng ta lưu trữ và biểu diễn thông tin. Em hãy trao đổi với các bạn về một vài nhiệm vụ xử lí xâu phổ biến mà em đã được học từ trước đến giờ và xem xét khả năng xử lí nhiệm vụ đó trong Python. Ví dụ: tách tên với họ và chữ lót của tên học sinh, tách các kí tự trong mã định danh sản phẩm để thực hiện phân luồng, ghép các kí tự được trích từ những xâu kí tự khác nhau để tạo thành mã hỗn hợp phục vụ cho việc quản lí công việc,...

THỰC HÀNH

Nhiệm vụ 1. Đếm số kí tự của một xâu thoả mãn điều kiện cho trước

Yêu cầu: Viết chương trình nhập họ và tên của học sinh (bao gồm các chữ cái và khoảng trắng) và in ra màn hình số kí tự không kể khoảng trắng.

Dữ liệu vào: Xâu họ và tên của học sinh.

Dữ liệu ra: Số kí tự khác khoảng trắng trong xâu họ và tên của học sinh.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập họ và tên của bạn: Nguyễn Văn An`	`Số kí tự khác khoảng trắng là: 11.`

Hướng dẫn: Ban đầu cho biến đếm bằng 0. Sử dụng vòng lặp để xét từng kí tự của xâu họ và tên: nếu kí tự đang xét khác khoảng trắng thì tăng biến đếm lên 1. Cuối cùng, giá trị của biến đếm là số kí tự khác khoảng trắng. Cụ thể như sau:

1. Nhập xâu `hoTen` từ bàn phím.
2. `n` = chiều dài của xâu `hoTen`
3. `kiTu = 0`
4. Duyệt từng kí tự `hoTen[i]` từ vị trí `0` đến vị trí `n-1`:
   Nếu `hoTen[i]` khác khoảng trắng:
   Tăng `kiTu` lên 1
5. In `kiTu` ra màn hình.

Chương trình tham khảo:

```
hoTen = input("Nhập họ và tên của bạn: ")
n = len(hoTen)
kiTu = 0
```

```
4 for i in range(0,n):
5     if hoTen[i] != " ":
6         kiTu += 1
7 print(f"Số kí tự khác khoảng trắng là: {kiTu}.")
```

Nhiệm vụ 2. Tách một xâu thành nhiều từ

Yêu cầu: Viết chương trình nhập xâu a (chỉ chứa chữ cái và khoảng trắng), in ra màn hình toàn bộ các từ trong xâu a, mỗi từ in trên 1 dòng. Một từ là dãy các chữ cái liên tiếp nhau, được phân cách bằng khoảng trắng.

Dữ liệu vào: Xâu a.

Dữ liệu ra: Các từ trong xâu a, mỗi từ được in trên 1 dòng.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị

`Nhập xâu cần xử lí: Tin  học 10`	`Tin`
	`học`
	`10`

Hướng dẫn: Duyệt xâu a để lấy mỗi từ trong xâu này. Sử dụng biến xâu tu để chứa một từ; ban đầu, xâu tu là rỗng. Sau đó, dùng vòng lặp để xét từng kí tự của xâu a: nếu kí tự đang xét khác khoảng trắng thì thêm kí tự này vào cuối xâu tu; ngược lại, kí tự đang xét là khoảng trắng và xâu tu khác rỗng thì in xâu tu ra màn hình (là một từ trong xâu a) và cho xâu tu là rỗng để tìm một từ kế tiếp trong xâu a. Khi xét xong xâu a, nếu xâu tu khác rỗng thì in xâu tu ra màn hình (là từ cuối cùng trong xâu a). Cụ thể như sau:

1. Nhập xâu a từ bàn phím.
2. n = chiều dài của xâu a.
3. tu = xâu rỗng.
4. Duyệt từng kí tự a[i] từ vị trí 0 đến vị trí n-1:
   Nếu a[i] khác khoảng trắng:
      Ghép a[i] vào phía sau của xâu tu
   Nếu a[i] bằng khoảng trắng và xâu tu khác rỗng.
      In xâu tu ra màn hình.
      tu = xâu rỗng.
5. Kết thúc duyệt, nếu xâu tu khác rỗng thì in xâu tu (đây là từ cuối cùng của xâu a).

Chương trình tham khảo:

```
1 a = input("Nhập xâu cần xử lí: ")
2 n = len(a)
3 tu = ""                    # Lưu trữ một từ của xâu a
4 for i in range(0,n):
5     if a[i] != " ":
6         tu += a[i]         # Ghép a[i] vào tu
7     elif tu != "":         # Xâu tu khác rỗng
8         print(tu)
9         tu = ""            # Bắt đầu từ mới
10 if tu != "":              # Từ cuối cùng
11     print(tu)
```

Nhiệm vụ 3. Kiểm tra xâu đối xứng (xâu Palindrome)

Yêu cầu: Viết chương trình kiểm tra xem một xâu có phải là đối xứng hay không (xâu viết xuôi hay viết ngược đều giống nhau, ví dụ 121, 1331, tết,...).

Dữ liệu vào: Xâu `a` cần kiểm tra tính đối xứng.

Dữ liệu ra: Thông báo về tính đối xứng của xâu `a`.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập xâu cần kiểm tra tính đối xứng: tết`	`tết là xâu đối xứng.`

Hướng dẫn: Sử dụng lát cắt xâu `a[::-1]` để tạo xâu mới là xâu đảo ngược của xâu `a`. Nếu xâu mới giống với xâu `a` thì xâu `a` là xâu đối xứng.

1. Nhập xâu `a` từ bàn phím.
2. So sánh giữa xâu `a` và xâu đảo ngược `a[::-1]`

   Nếu hai xâu trùng nhau:

   In ra màn hình thông báo xâu đối xứng.

   Ngược lại:

   In ra màn hình thông báo xâu không đối xứng.

Chương trình tham khảo

```
kTra = input("Nhập xâu cần kiểm tra tính đối xứng: ")
if kTra == kTra[::-1]:
    print(f"{kTra} là xâu đối xứng.")
else:
    print(f"{kTra} không là xâu đối xứng.")
```

LUYỆN TẬP

Hoàn thiện chương trình của Nhiệm vụ 1 của phần THỰC HÀNH, in ra màn hình số khoảng trắng của xâu `hoTen` và số từ trong xâu `hoTen`.

Gợi ý: Một từ được bắt đầu bằng kí tự khác khoảng trắng và nếu có kí tự liền trước thì kí tự này là khoảng trắng. Trong vòng lặp ở bước 4, thêm lệnh `if` để nhận biết một từ và đếm số từ.

VẬN DỤNG

Viết chương trình chuẩn hoá xâu họ tên bằng cách xoá bỏ khoảng trắng ở đầu và cuối xâu; kí tự đầu tiên của mỗi từ là chữ hoa, các kí tự còn lại của mỗi từ là chữ thường; các từ chỉ cách nhau một khoảng trắng.

Dữ liệu vào: Xâu `hoTen`.

Dữ liệu ra: Xâu `hoTenMoi` với nội dung là xâu `hoTen` được chỉnh dạng.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập xâu cần xử lí: nguyễn trần anh vy`	`Nguyễn Trần Anh Vy`

Gợi ý: Bên cạnh việc xử lí duyệt từng kí tự như đã thực hiện trong Nhiệm vụ 1, Nhiệm vụ 2 ở phần THỰC HÀNH, em có thể sử dụng các hàm `lower()`, `upper()`, `capitalize()`, `title()` để chỉnh dạng xâu theo yêu cầu; hàm `strip()` để trả về một xâu đã loại bỏ các khoảng trắng ở đầu và cuối xâu.

Chân trời sáng tạo

BÀI F11 DANH SÁCH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết được cấu trúc dữ liệu mảng.
- Biết được danh sách và cách biểu diễn danh sách trong Python, biểu diễn mảng bằng danh sách.
- Sử dụng được một số phép toán về danh sách của Python.

KHỞI ĐỘNG

Cô giáo cần tìm học sinh có điểm cao nhất trong bảng điểm môn Tin học của lớp 10A1 và bạn An muốn viết chương trình để giải quyết yêu cầu này của cô giáo. Việc lưu trữ từng điểm kiểm tra và tên của học sinh vào các biến khác nhau để tìm giá trị lớn nhất sẽ gây khó khăn cho việc viết chương trình vì số lượng biến quá nhiều. Bạn An cần sử dụng kiểu dữ liệu danh sách để lưu trữ điểm kiểm tra và tên của học sinh để xử lí yêu cầu tìm học sinh có điểm cao nhất. Em hãy nêu một vài trường hợp cần sử dụng kiểu dữ liệu danh sách để lưu trữ.

KHÁM PHÁ

1. Danh sách trong Python

Trong Python, danh sách là một dãy các phần tử có thể có kiểu dữ liệu bất kì và được khởi tạo theo cú pháp như sau:

```
<tên danh sách> = [<phần tử 0>,<phần tử 1>,…,<phần tử n>]
```

Trong đó, các `<phần tử 0>, <phần tử 1>,… , <phần tử n>` là các giá trị có thể có cùng kiểu dữ liệu hoặc có kiểu dữ liệu khác nhau (số nguyên, số thực, xâu kí tự, giá trị logic True False, giá trị rỗng None,…). Có thể truy cập từng phần tử của danh sách thông qua chỉ số. Để truy cập phần tử có chỉ số `i` của danh sách `A`, sử dụng cú pháp `A[i]`.

Ví dụ 1: Quan sát các lệnh bên, biến `A` là danh sách các số nguyên có 4 phần tử, biến `B` là danh sách có 6 phần tử thuộc nhiều kiểu dữ liệu khác nhau.

```
>>> A = [1,3,4,2]
>>> A[0]
1
>>> A[3]
2
>>> A[-1]
2
>>> B = [1,2,"Tin học 10","Python",True,None]
>>> B[2]
'Tin học 10'
>>> B[-2]
True
```

`A[0]` là phần tử có chỉ số `0` (phần tử đầu tiên) của danh sách `A`, `A[3]` là phần tử có chỉ số `3`

của danh sách A, A[-1] là phần tử có chỉ số -1 của danh sách A. Tương tự B[2] và B[-2] là các phần tử có chỉ số 2 và -2 của danh sách B. Các câu lệnh trong Ví dụ 1 cho thấy cách truy cập các phần tử của danh sách thông qua chỉ số.

Trong Python, chỉ số của các phần tử có thể được đánh bằng chỉ số dương (từ trái sang phải bắt đầu từ chỉ số 0) hoặc chỉ số âm (từ phải sang trái bắt đầu từ chỉ số -1). Quan sát *Hình 1*, danh sách A = [2,6,5,1,3] có n = 5 phần tử. Áp dụng cách đánh chỉ số dương, chỉ số của phần tử đầu tiên là 0, chỉ số của phần tử cuối cùng là 4. Áp dụng cách đánh chỉ số âm, chỉ số của phần tử cuối cùng là -1, chỉ số của phần tử đầu tiên là -5.

Danh sách A	2	6	5	1	3
Chỉ số dương	0	1	2	3	4
Chỉ số âm	-5	-4	-3	-2	-1

Hình 1. Danh sách A = [2,6,5,1,3] được đánh chỉ số theo 2 cách

Việc chỉ số hoá từng phần tử của danh sách cho thấy có thể sử dụng danh sách để biểu diễn một mảng trong nhiều ngôn ngữ lập trình bậc cao khác, lưu ý rằng các phần tử trong mảng bắt buộc phải thuộc cùng một kiểu dữ liệu.

Chiều dài của danh sách là số phần tử của danh sách. Danh sách rỗng là danh sách không có phần tử nào, có chiều dài bằng 0 và được kí hiệu là []. Trong Python, hàm len(A) trả về chiều dài của danh sách A.

Danh sách A có thể có phần tử là một danh sách và được gọi là danh sách con. Khi danh sách A có phần tử là danh sách con thì danh sách A được gọi là danh sách lồng. Sử dụng lệnh type() để kiểm tra kiểu dữ liệu của các phần tử trong danh sách.

Ví dụ 2: Danh sách A = [3,[5,6,7],True] (*Hình 2*) là danh sách lồng với phần tử A[1] = [5,6,7] là danh sách con. Sử dụng cú pháp A[1][1] để truy xuất giá trị phần tử vị trí 1 của danh sách con A[1].

Danh sách A	3	[5,6,7]	True
Chỉ số dương	0	1	2

Hình 2. Danh sách A = [3,[5,6,7],True]

Quan sát các lệnh bên để tìm hiểu về cách truy xuất các giá trị trong danh sách A, cập nhật giá trị bằng lệnh gán và kiểm tra kiểu dữ liệu của các phần tử.

```
>>>A = [3,[5,6,7],True]
>>>A[1]
   [5, 6, 7]
>>>A[1][1]
   6
>>>type(A[1])
   <class 'list'>
>>>type(A[1][1])
   <class 'int'>
>>>A[1][2] = 8
>>>A[2] = False
>>>A
   [3, [5, 6, 8], False]
```

1. Em hãy viết lệnh tạo danh sách A = [1,2,3,4,5], sau đó tìm kiếm phần tử có giá trị 0, 3, 6 trong danh sách A.

2. Em hãy viết lệnh tạo danh sách B có 5 phần tử cùng mang giá trị 1, sau đó kết nối với danh sách A ở câu trên tạo thành danh sách C = A+B, in danh sách C ra màn hình.

Trong Python:

- *Danh sách* là một dãy các phần tử có thể có kiểu dữ liệu bất kì.
- *Chiều dài* của danh sách là số phần tử của danh sách. *Danh sách rỗng* có chiều dài bằng 0.
- Phần tử chỉ số `i` của danh sách `A` được kí hiệu là `A[i]`. Chỉ số của các phần tử có thể được đánh bằng chỉ số dương (từ trái sang phải bắt đầu từ chỉ số `0`) hoặc chỉ số âm (từ phải sang trái bắt đầu từ chỉ số `-1`).

2. Một số phép toán về danh sách

a) Phép duyệt danh sách

Có thể dùng lệnh `for` để duyệt các phần tử của một danh sách; có hai dạng lệnh: `for - range()` và `for - danh sách`. Quy định về hàm `range()` được trình bày ở mục 1a trong KHÁM PHÁ của Bài F7.

Ví dụ 3: Chương trình sau đây hiển thị các phần tử của danh sách `A`.

```
A = [1,6,2]                 # Tạo danh sách A
for i in range(len(A)):     # Dùng lệnh for - range()
    print(A[i],end = ' ')   # Hiển thị phần tử A[i]
print()
for x in A:                 # Dùng lệnh for - danh sách
    print(x,end = ' ')      # Hiển thị phần tử x
```

b) Phép tìm kiếm trong danh sách

Phép *nằm trong* (toán tử `in`) và phép *không nằm trong* (toán tử `not in`) dùng để tìm kiếm phần tử trong một danh sách và có cú pháp như sau:

```
<phần tử> in <danh sách>
<phần tử> not in <danh sách>
```

– Phép *nằm trong* (toán tử `in`) dùng để tìm kiếm `<phần tử>` nằm trong danh sách `<danh sách>` và trả về giá trị True nếu tìm thấy hoặc False nếu không tìm thấy.

– Phép *không nằm trong* (toán tử `not in`) dùng để tìm kiếm `<phần tử>` không nằm trong danh sách `<danh sách>` và trả về giá trị True nếu không tìm thấy hoặc False nếu tìm thấy.

Ví dụ 4: Các lệnh bên minh hoạ phép tìm kiếm phần tử trong danh sách. Trong đó:

– Danh sách `A` là danh sách lồng với 3 phần tử `A[0] = 1`, `A[1] = [2,3]`, `A[2] = 4`.

– Giá trị 2 không thuộc danh sách A mà thuộc danh sách con `A[1]`, do đó kết quả của lệnh `2 in A` là False.

```
>>> A = [1,[2,3],4]
>>> B = [2,3]
>>> 2 in A
False
>>> B in A
True
>>> 1 not in A
False
```

– Danh sách B chính là phần tử A[1] nên kết quả của lệnh B in A là True.

– Giá trị 1 là phần tử thuộc danh sách A (phần tử A[0]) nên kết quả của lệnh 1 not in A là False.

c) Phép kết nối danh sách

Phép kết nối danh sách (toán tử +) dùng để tạo một danh sách mới bằng cách kết nối hai hoặc nhiều danh sách lại với nhau và có cú pháp như sau:

<danh sách 1>+<danh sách 2>+…+<danh sách n>

Danh sách mới chứa tất cả các phần tử từ <danh sách 1>, sau đó là tất cả các phần tử từ <danh sách 2>,… , cuối cùng là tất cả phần tử từ <danh sách n>.

Ví dụ 5: Các lệnh bên dưới minh hoạ phép kết nối hai danh sách A, B và gán cho danh sách C.

```
>>>A = [1,6,2] # Tạo danh sách A
>>>B = [4,5]   # Tạo danh sách B
>>>C = A+B     # C = [1,6,2,4,5]
>>>print(C)
   [1, 6, 2, 4, 5]
```

d) Phép nhân bản danh sách

Phép nhân bản danh sách (toán tử *) dùng để tạo một danh sách mới bằng cách nhân bản một danh sách nhiều lần. Phép nhân bản danh sách có cú pháp như sau:

<danh sách gốc>*<số lần>

Danh sách mới chứa tất cả các phần tử từ <danh sách gốc> kết nối với nhau với số lần lặp là <số lần> (là số nguyên). Nếu <số lần> ≤ 0 thì danh sách mới là danh sách rỗng.

Ví dụ 6: Các lệnh bên dưới minh hoạ cho phép nhân bản danh sách.

```
>>>A = [1,6,2] # Tạo danh sách A
>>>B = A*2     # B = [1,6,2,1,6,2]
>>>print(B)
   [1, 6, 2, 1, 6, 2]
>>>C = A*-1    # C = [] vì số lần nhân bản là số âm
>>>print(C)
   []
>>>D = [0]*3   # D = [0,0,0]
>>>print(D)
   [0, 0, 0]
```

Trong các lệnh trên, danh sách A được nhân bản 2 lần và gán cho danh sách B. Với số lần nhân bản là -1 (số âm), danh sách C là rỗng. Danh sách [0] được nhân bản 3 lần và gán cho danh sách D.

- Dùng lệnh `for - range()` và `for - danh sách` để duyệt các phần tử của danh sách.
- Phép *nằm trong* (toán tử `in`) dùng để tìm kiếm `<phần tử>` nằm trong danh sách `<danh sách>` và trả về giá trị True nếu tìm thấy hoặc False nếu không tìm thấy. Ngược lại, phép *không nằm trong* (toán tử `not in`) sẽ trả về giá trị False nếu tìm thấy hoặc True nếu không tìm thấy.
- Phép kết nối danh sách dùng để kết nối các danh sách với nhau và trả về một danh sách.
- Phép nhân bản danh sách dùng để nhân bản danh sách nhiều lần và trả về một danh sách.

3. Gán các phần tử của danh sách cho các biến

Có thể gán lần lượt các phần tử của danh sách cho các biến và có cú pháp như sau:

`<tên biến 1>...<tên biến n> = <danh sách có độ dài n>`

Trong đó, `<tên biến 1>...<tên biến n>` là các tên biến được gán các phần tử tương ứng trong `<danh sách có độ dài n>`. Điều kiện gán là số tên biến phải bằng số phần tử của `<danh sách có độ dài n>`.

Ví dụ 7: Trong các lệnh sau đây, các phần tử của danh sách `A` được gán lần lượt cho các biến `b`, `c`, `d` và các phần tử của danh sách `[5,"An"]` được gán cho hai biến `e`, `f`.

```
>>>A = [1,[2,3],4] # Tạo danh sách A
>>>b,c,d = A       # b = 1;c = [2,3];d = 4
>>>e,f = [5,"An"]  # e = 5;f = "An"
>>>print(b,c,d,e,f,sep=";")
  1;[2, 3];4;5;An
```

Với `A = [1,2,[3,4]]`, lệnh gán `b,c,d = A`. Giá trị của `c` là:

A. 1 B. 2 C. 3 D. 4

Gán lần lượt các phần tử của danh sách cho các biến theo cú pháp:

`<tên biến 1>...<tên biến n> = <danh sách có độ dài n>`.

4. Tạo danh sách

Có thể tạo danh sách dùng `for - range()` mà các phần tử của danh sách này có được dựa vào giá trị trả về của hàm `range()`.

Ví dụ,

+ với `A = [0 for i in range(3)]` thì `A = [0,0,0]`, danh sách `A` có 3 phần tử cùng mang giá trị 0.

+ với `B = [2*i for i in range(0,3)]` thì `B = [0,2,4]`, danh sách `B` có 3 phần tử mang giá trị `2*i` với i lần lượt bằng 0, 1, 2.

+ với `C = [[i,2*i,0] for i in range(0,5,2)]` thì `C = [[0,0,0],[2,4,0],[4,8,0]]`, danh sách `C` có 3 phần tử là các danh sách mang giá trị `[i,2*i,0]` với i lần lượt bằng 0, 2, 4.

Trong Python, khi gán danh sách A cho danh sách B thì cả hai danh sách này sử dụng chung một vùng nhớ, mọi thay đổi ở danh sách A cũng là thay đổi ở danh sách B.

Ví dụ 8: Trong các lệnh dưới đây, gán danh sách A cho danh sách B, sau đó thực hiện thay đổi giá trị của A[0] thì giá trị của B[0] cũng bị thay đổi theo.

```
>>>A = [1,6,2] # Tạo danh sách A
>>>B = A       # Danh sách B là danh sách A
>>>A[0] = 5    # Thay đổi A[0] = 5, kéo theo thay đổi B[0] = 5
>>>print(A)    # A = [5,6,2]
   [5, 6, 2]
>>>print(B)    # B = [5,6,2]
   [5, 6, 2]
```

Tuy nhiên, sau khi gán danh sách A cho danh sách B, nếu tiếp tục gán một danh sách khác cho danh sách A thì vùng nhớ của 2 danh sách A và B là khác nhau, mọi thay đổi ở danh sách A không gây ảnh hưởng đến danh sách B.

Ví dụ 9: Trong các lệnh dưới đây, gán danh sách A cho danh sách B, sau đó gán danh sách [4,7] cho danh sách A. Lúc này vùng nhớ của danh sách A và danh sách B là khác nhau, thay đổi ở danh sách A không gây ảnh hưởng đến danh sách B.

```
>>>A = [1,6,2] # Tạo danh sách A
>>>B = A       # B = [1,6,2]
>>>A = [4,7]   # Gán danh sách mới: A = [4,7]
>>>A[0] = 5    # Thay đổi phần tử A[0], A = [5,6,2]
>>>print(A)    # A = [5,6,2]
   [5, 7]
>>>print(B)    # B = [1,6,2]
   [1, 6, 2]
```

Ngoài ra, trong Python, có thể tạo danh sách B giống danh sách A nhưng hai danh sách sử dụng vùng nhớ khác nhau. Thao tác này được gọi là *lệnh gán từng phần tử* của danh sách A cho danh sách B và được thực hiện như sau:

1. Tạo danh sách B chứa các phần tử None, có số phần tử bằng số phần tử của danh sách A.
2. Gán từng phần tử của danh sách A cho từng phần tử tương ứng của danh sách B.

Ví dụ 10: Trong các lệnh dưới đây, danh sách B được tạo mới với 3 phần tử None, sau đó thực hiện gán từng phần tử của danh sách A cho danh sách B.

```
>>>A = [1,6,2]
>>>B = [None]*len(A)          # Tạo B chứa các phần tử None
>>>for i in range(len(A)):    # Với mỗi phần tử A[i]
...     B[i] = A[i]           # Gán A[i] cho B[i]
...
...
>>>A[0] = 5
>>>print(A)
   [5, 6, 2]
>>>print(B)
   [1, 6, 2]
```

Em hãy quan sát các lệnh dưới đây và cho biết kết quả nhận được là gì.

```
A = [1,2,3,4,5]
B = A
C = [None]*len(A)
for i in range(len(A))
    C[i] = A[i]
A[3] = 3
print(A)
print(B)
print(C)
```

- Có thể tạo danh sách bằng cách sử dụng lệnh `for - range()`.
- Khi gán danh sách `A` cho danh sách `B` bằng lệnh gán `B = A`, hai danh sách này sử dụng chung vùng nhớ, mọi thay đổi các phần tử ở danh sách `A` cũng thay đổi các phần tử ở danh sách `B`.
- Lệnh gán từng phần tử của danh sách `A` cho danh sách `B` tạo ra danh sách `B` giống danh sách `A` nhưng hai danh sách sử dụng vùng nhớ khác nhau.

LUYỆN TẬP

Viết chương trình nhập danh sách `A` gồm các số nguyên và in ra màn hình danh sách `B` là các số nguyên khác nhau trong danh sách `A`.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
`Nhập danh sách A: 1 1 1 2 2 2 3 4 4 5 5 3 3 2 3 4 1`	`Danh sách B: [1, 2, 3, 4, 5]`

Gợi ý: Duyệt từng phần tử trong danh sách `A`, sử dụng phép không nằm trong (`not in`) và phép kết nối danh sách (+) để tạo danh sách `B` gồm các số nguyên khác nhau trong danh sách `A`.

VẬN DỤNG

Viết chương trình nhập danh sách các số nguyên và thông báo về số lượng số lẻ và số chẵn trong danh sách.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập danh sách các số nguyên: 1 1 1 2 2 3 4 4 5 5 3 3 2 3 4 1`	`Danh sách có 10 số lẻ, 6 số chẵn.`
`Nhập danh sách các số nguyên: 1 3 5 7 9 1 3 1 1`	`Danh sách có 9 số lẻ, 0 số chẵn.`
`Nhập danh sách các số nguyên: 0 4 8 10 6 4 2 0 2 4 6`	`Danh sách có 0 số lẻ, 11 số chẵn.`

Gợi ý: Duyệt từng phần tử trong danh sách, sử dụng phép chia lấy số dư (%) để đếm số lượng số lẻ và số chẵn có trong danh sách.

BÀI F12 MỘT SỐ HÀM TRÊN DANH SÁCH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết được một số hàm thư viện trên danh sách của Python.
- Sử dụng được một số hàm thư viện trên danh sách để giải quyết một số bài toán liên quan đến danh sách.

KHỞI ĐỘNG

Bạn Hải nhập danh sách `A = [7,1,3,6,2]`. Để thêm một phần tử giá trị 5 vào cuối danh sách `A`, bạn Hải có thể sử dụng toán tử + để kết nối danh sách `A` với danh sách `B = [5]`. Em hãy đề xuất cách để bạn Hải thêm phần tử 5 vào đầu danh sách. Nếu bạn Hải được yêu cầu thêm phần tử giá trị 5 vào vị trí 3 trong danh sách `A`, thì bạn Hải cần phải làm thế nào?

KHÁM PHÁ

Python cung cấp một số hàm thư viện trên danh sách để viết chương trình giải quyết nhiều bài toán liên quan đến danh sách một cách dễ dàng và nhanh chóng.

Cho danh sách `A = [7,1,3,6,2]`. *Hình 1* biểu diễn danh sách `A` với các chỉ số dương và chỉ số âm của các phần tử.

Danh sách A	7	1	3	6	2
Chỉ số dương	0	1	2	3	4
Chỉ số âm	-5	-4	-3	-2	-1

Hình 1. Biểu diễn danh sách `A = [7,1,3,6,2]`

1. Các hàm chèn phần tử vào danh sách

Có thể chèn phần tử vào đầu, vào giữa hoặc vào cuối một danh sách bằng cách sử dụng các hàm thư viện của Python. Cụ thể:

– Hàm `<A>.append(<k>)` chèn phần tử có giá trị `<k>` vào cuối danh sách `<A>`. Ví dụ, ở các lệnh bên, lệnh `A.append(4)` chèn phần tử có giá trị 4 vào cuối danh sách `A`.

```
>>> A = [7,1,3,6,2]
>>> A.append(4)
>>> A
[7, 1, 3, 6, 2, 4]
```

– Hàm `<A>.insert(<i>,<k>)` chèn phần tử có giá trị `<k>` vào vị trí `<i>` của danh sách `<A>`. Nếu `<i>` ≥ `len(<A>)`: chèn phần tử có giá trị `<k>` vào cuối danh sách `<A>`. Nếu `<i>` ≤ `-len(<A>)`: chèn phần tử có giá trị `<k>` vào đầu danh sách `<A>`. Ví dụ, quan sát các lệnh bên, danh sách `A = [1,2,3]` có 3 phần tử, lệnh `A.insert(1,4)` chèn giá trị 4 vào vị trí 1, danh sách `A = [1,4,2,3]` có 4 phần tử với `A[1] = 4`. Lệnh `A.insert(6,5)` chèn 5 vào vị trí 6 (lớn hơn `len(A)`) nên Python sẽ chèn 5 vào vị trí cuối danh sách, có 5 phần tử với `A[4] = 5`. Lệnh `A.insert(-7,0)` chèn 0 vào vị trí -7 (nhỏ hơn `-len(A)`) nên Python sẽ chèn 0 vào vị trí đầu danh sách, `A = [0,1,4,2,3,5]` có 6 phần tử với `A[0] = A[-6] = 0`.

```
>>> A = [1,2,3]
>>> A.insert(1,4)
>>> A
    [1, 4, 2, 3]
>>> A.insert(6,5)
>>> A
    [1, 4, 2, 3, 5]
>>> A.insert(-7,0)
>>> A
    [0, 1, 4, 2, 3, 5]
```

Cho danh sách `A = [7,3,6,2,4,5]`. Em hãy viết các câu lệnh để được danh sách `A = [7,8,8,3,6,2,8,4,5,8]`.

Các hàm chèn phần tử vào danh sách: `append()`, `insert()`.

2. Các hàm xoá phần tử của danh sách

Có thể xoá một phần tử của danh sách ở đầu, ở giữa hoặc ở cuối một danh sách và xoá rỗng danh sách bằng cách sử dụng các hàm thư viện của Python. Cụ thể:

– Hàm `<A>.remove(<k>)` xoá phần tử có giá trị `<k>` của lần xuất hiện đầu tiên trong danh sách `<A>`. Nếu không tồn tại giá trị `<k>` trong danh sách `<A>` thì thông báo lỗi `ValueError`; để tránh lỗi này, nên kiểm tra điều kiện `k in A` là `True` trước khi gọi hàm `remove()`.

```
>>> A = [7,6,3,6,2]
>>> A.remove(6)
>>> A
    [7, 3, 6, 2]
>>> A.clear()
>>> A
    []
```

– Hàm `<A>.clear()` xoá rỗng danh sách `<A>`.

Ví dụ, quan sát các lệnh ở trên, lệnh `A.remove(6)` sẽ xoá giá trị 6 trong danh sách `A`, lệnh `A.clear()` sẽ xoá rỗng danh sách `A`.

Cho danh sách `A = [7,8,8,3,6,2,8,4,5,8]`. Em hãy viết các lệnh để được danh sách `A = [7,3,6,2,4,9,5]`.

Các hàm xoá phần tử của danh sách: `remove()`, `clear()`.

LUYỆN TẬP

Cho chương trình sau:

```
A = [7,1,1,3,6,2,1,4,1,5]
B = []
C = []
for x in A:
    if x % 2 == 0:
        if x not in B:
            B.append(x)
    else:
        if x not in C:
            C.append(x)
print(B)
print(C)
```

Theo em, lệnh `append()` trong chương trình này thực hiện công việc gì? Và kết quả của chương trình là gì?

THỰC HÀNH

Nhiệm vụ 1. Tìm kiếm trong danh sách

Yêu cầu: Viết chương trình nhập danh sách các số nguyên và số nguyên n. Sau đó, tìm số nguyên n trong danh sách này và hiển thị chỉ số của lần xuất hiện đầu tiên nếu tìm thấy, nếu không tìm thấy thì in ra thông báo không tìm thấy n.

Gợi ý: Sử dụng câu lệnh `list(map(int,input("Nhập danh sách số nguyên: ").split()))` để tạo ra danh sách các số nguyên được tách từ xâu đã nhập. Sau đó, dùng vòng lặp để xét từng số nguyên trong danh sách: nếu số đang xét là n thì hiển thị chỉ số của số này. Nếu đã xét hết danh sách mà không tìm thấy n thì hiển thị thông báo không tìm thấy.

Chương trình tham khảo:

```
# Tạo danh sách A bằng cách nhập xâu (tách, chuyển kiểu)
A = list(map(int,input("Nhập danh sách số nguyên: ").split()))
n = int(input("Nhập số nguyên n: "))
chiSo = -1
for i in range(len(A)):      # Xét từng phần tử A[i]
    if A[i] == n:
        chiSo = i            # Chỉ số của lần xuất hiện đầu tiên
        break
if chiSo == -1:
    print(f"Không tìm thấy {n}.")
else:
    print(f"Chỉ số lần xuất hiện đầu tiên của {n} là {chiSo}.")
```

Nhiệm vụ 2. Nhập bảng điểm của môn học

Yêu cầu: Viết chương trình lặp lại nhiều lần công việc nhập dữ liệu của học sinh: nhập mã số, nếu mã số nhỏ hơn hoặc bằng 0 thì kết thúc việc nhập, ngược lại, tiếp tục nhập họ tên, điểm số và chèn dữ liệu của học sinh này vào cuối bảng điểm (ban đầu, bảng điểm là rỗng). Cuối cùng, hiển thị bảng điểm của môn học ra màn hình, mỗi dòng gồm mã số, họ tên và điểm số của học sinh.

Hướng dẫn: Sử dụng hàm `append()` để chèn dữ liệu của học sinh gồm mã số, họ tên và điểm số vào cuối bảng điểm của môn học. Cụ thể như sau:

1. Tạo bảng điểm `bangDiem` rỗng.
2. Còn tiếp tục nhập.
3. Vòng lặp trong khi còn tiếp tục nhập:

   Nhập mã số và gán cho biến `maSo`

   Nếu `maSo <= 0` thì:

   Kết thúc nhập.

   Ngược lại:

   Nhập họ tên và gán cho biến `hoTen`.

   Nhập điểm số và gán cho biến `diemSo`.

   Chèn học sinh `[maSo, hoTen, diemSo]` vào cuối bảng điểm `bangDiem`.
4. Vòng lặp i từ 0 đến (số học sinh – 1): # Hiển thị bảng điểm của môn học.

   Lấy mã số, họ tên, điểm số của học sinh [i].

   Hiển thị mã số, họ tên, điểm số.

Chương trình tham khảo:

```
1 bangDiem = []
2 tiepTuc = True
3 while tiepTuc:
4     maSo = int(input("Nhập mã số (kết thúc = 0): "))
5     if maSo <= 0:
6         tiepTuc = False
7     else:
8         hoTen = input("Nhập họ tên: ")
9         diemSo = int(input("Nhập điểm số: "))
10        bangDiem.append([maSo,hoTen,diemSo])
11 print("Bảng điểm của môn học:")
12 for i in range(len(bangDiem)):
13     maSo,hoTen,diemSo = bangDiem[i]
14     print(f"Học sinh {i+1}: {maSo} - {hoTen}: {diemSo}")
```

Giáo dục sáng tạo

VẬN DỤNG

1. Tiếp tục chương trình nhập của Nhiệm vụ 2, viết thêm đoạn mã để tìm học sinh có điểm cao nhất nhằm giải quyết yêu cầu ở KHỞI ĐỘNG của Bài F11.

Hướng dẫn: Tiếp tục hướng dẫn của Nhiệm vụ 2. Duyệt bảng điểm để tìm điểm cao nhất. Sau đó, duyệt bảng điểm lần thứ hai để hiển thị các học sinh có điểm cao nhất này. Cụ thể như sau:

5. Gán `diemCaoNhat = -1`

6 Vòng lặp i từ 0 đến (số học sinh – 1):

Lấy `diemSo` của học sinh [i]

Nếu `diemSo` > `diemCaoNhat` thì:

`diemCaoNhat` = `diemSo`

7 Vòng lặp i từ 1 đến (số học sinh – 1):

Lấy `maSo`, `hoTen`, `diemSo` của học sinh [i]

Nếu `diemSo` = `diemCaoNhat` thì:

Hiển thị `maSo`, `hoTen`, `diemSo`.

2. Viết chương trình nhập danh sách `A` bao gồm các số nguyên và số nguyên `x`; xoá bỏ tất cả các phần tử có giá trị bằng `x` trong danh sách `A`; in danh sách `A` ra màn hình. Ví dụ, với danh sách `A = [7,1,1,3,6,2,1,4,1,5]` và `x = 1`. Danh sách `A` sau khi xoá bỏ các phần tử có giá trị bằng 1 là `A = [7,3,6,2,4,5]`.

Hướng dẫn: Sử dụng hàm `A.count(x)` để đếm số lần xuất hiện của giá trị `x` trong danh sách `A` rồi gán cho biến `n`. Sau đó, thực hiện `n` lần việc xoá phần tử có giá trị `x` trong danh sách `A` bằng cách sử dụng hàm `A.remove(x)`. Cụ thể như sau:

1 Nhập danh sách số nguyên `A`.

2 Nhập số nguyên `x`.

3 Hiển thị danh sách `A` ban đầu.

4 `n` = số lần xuất hiện của số `x` trong danh sách `A`.

5 Vòng lặp `i` từ `0` đến `n-1`:

Xoá số `x` trong danh sách `A`.

6 Hiển thị danh sách `A` sau khi xoá số `x`.

3. Viết chương trình nhập các số nguyên, kết thúc việc nhập bằng phím **Enter**. In ra màn hình danh sách các số vừa nhập được sắp xếp theo thứ tự tăng dần. Ví dụ, nhập các số 1, 5, 2, 6, 4, 7, 3, 1. In ra màn hình danh sách `[1,1,2,3,4,5,6,7]`.

Gợi ý: Tạo `A` là danh sách chứa các số nguyên đã nhập, được sắp xếp theo thứ tự tăng dần; ban đầu, danh sách `A` rỗng. Khi nhập số nguyên `x`, tìm vị trí `i` thích hợp từ đầu danh sách `A` để chèn số nguyên này sao cho danh sách `A` vẫn có thứ tự tăng dần bằng cách sử dụng hàm `A.insert(i,x)` và `A.append(x)`.

Chân trời sáng tạo

BÀI F13 HÀM TRONG PYTHON

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết được một số hàm trong thư viện chuẩn của Python.
- Tạo được hàm tự định nghĩa.
- Sử dụng các hàm thư viện chuẩn và các hàm tự định nghĩa để viết chương trình giải quyết một bài toán đơn giản.

KHỞI ĐỘNG

Em hãy quan sát và cho nhận xét về các cách viết chương trình tìm ước số chung lớn nhất của hai số nguyên dương a, b cho trước sau đây:

Chương trình 1	Chương trình 2

`a = int(input("Nhập a = "))`
`b = int(input("Nhập b = "))`
`while b > 0:`
`  r = a%b`
`  a = b`
`  b = r`
`print(a)`
	`import math`
	`a = int(input("Nhập a = "))`
	`b = int(input("Nhập b = "))`
	`print(math.gcd(a,b))`

KHÁM PHÁ

1. Giới thiệu về hàm trong Python

Một bài toán ban đầu phức tạp có thể được chia thành các bài toán con đơn giản hơn. Khi viết chương trình giải quyết bài toán ban đầu thì chương trình sẽ bao gồm nhiều khối lệnh, mỗi khối lệnh giải quyết một bài toán con, làm cho chương trình có rất nhiều câu lệnh, khó hiểu, …

Để chương trình đơn giản và dễ hiểu hơn, mỗi khối lệnh giải quyết một bài toán con sẽ được viết thành một chương trình con được đặt tên. Chương trình con này được gọi là hàm, tên của chương trình con được gọi là tên hàm. Sau đó, chương trình giải quyết bài toán ban đầu sẽ thực hiện các bài toán con bằng cách gọi thực hiện các hàm này thông qua lời gọi hàm.

Chân trời sáng tạo

Ví dụ 1: Trong Python, chương trình giải phương trình ax + b = 0 có hai cách viết như sau:

Cách 1: Chương trình bao gồm các lệnh nhập giá trị của a, b (các dòng lệnh 2, 3) và các lệnh giải phương trình (các dòng lệnh 5 – 12).

```
1 # Nhập a và b
2 a = float(input("Nhập a: "))
3 b = float(input("Nhập b: "))
4 # Giải phương trình ax + b = 0
5 if a == 0:
6     if b == 0:
7         print("Phương trình có vô số nghiệm.")
8     else:
9         print("Phương trình vô nghiệm.")
10 else:
11     x = -b/a
12     print(f"Nghiệm x = {x}")
```

Cách 2: Đoạn lệnh giải phương trình được viết thành chương trình con (hàm) có tên là giaiPT (các dòng lệnh 2 – 10). Chương trình bao gồm các lệnh nhập giá trị của a, b (các dòng lệnh 12, 13) và lời gọi hàm giaiPT(a,b) để giải phương trình (dòng lệnh 15).

```
1 # Định nghĩa hàm giaiPT(a,b) để giải phương trình ax + b = 0
2 def giaiPT(a,b):
3     if a == 0:
4         if b == 0:
5             print("Phương trình có vô số nghiệm.")
6         else:
7             print("Phương trình vô nghiệm.")
8     else:
9         x = -b/a
10        print(f"Nghiệm x = {x}")
11 # Nhập a và b
12 a = float(input("Nhập a: "))
13 b = float(input("Nhập b: "))
14 # Lời gọi hàm giaiPT()
15 giaiPT(a,b)
```

Một hàm có thể được gọi từ các hàm khác, điều này tránh được một đoạn mã lệnh bị trùng lặp nhiều lần với các dữ liệu khác nhau.

Ví dụ 2: Tính diện tích của hình chữ nhật có các độ dài a, b và hình chữ nhật có các độ dài c, d.

Cách 1: Công thức tính diện tích được viết hai lần (bị lặp lại) ở hai dòng lệnh.

```
1 dienTich1 = a*b # Tính diện tích hình chữ nhật, độ dài a,b
2 dienTich2 = c*d # Tính diện tích hình chữ nhật, độ dài c,d
```

Cách 2: Hàm dienTichChuNhat(x,y) để tính diện tích hình chữ nhật có các độ dài x, y dưới đây, công thức tính diện tích hình chữ nhật chỉ được viết một lần trong toàn bộ chương trình.

```
1 # Hàm tính diện tích hình chữ nhật có các độ dài x,y
2 def dienTichChuNhat(x,y):
3     dienTich = x*y    # Tính diện tích hình chữ nhật, độ dài x,y
4     return dienTich
```

Sau đó, gọi hàm này để tính diện tích của hai hình chữ nhật ở trên:

```
5 # Tính diện tích hình chữ nhật, độ dài a,b
6 dienTich1 = dienTichChuNhat(a,b)
7 # Tính diện tích hình chữ nhật, độ dài c,d
8 dienTich2 = dienTichChuNhat(c,d)
```

Quan sát đoạn lệnh sau và cho biết kết quả in ra màn hình.

```
1 # Các hàm tính CV
2 def tinhCV1(a,b):
3     cV = (a+b)*2
4     return cV
5
6 def tinhCV2(a):
7     cV = a*4
8     return cV
9
10 # Chương trình chính
11 print(tinhCV1(5,10))
12 print(tinhCV2(5))
```

- Hàm là một chương trình con gồm một khối lệnh để thực hiện một công việc.
- Lời gọi hàm dùng để gọi thực hiện một hàm.

2. Hàm có sẵn và một số hàm trong thư viện chuẩn của Python

Trong lập trình, hàm có sẵn là hàm được tích hợp sẵn trong một ngôn ngữ lập trình. Lập trình viên có thể sử dụng các hàm này thông qua lời gọi hàm.

Ví dụ 3: Đoạn lệnh Python dưới đây nhập một số nguyên từ bàn phím bằng cách dùng hàm `input()` để nhập một xâu kí tự từ bàn phím và gán cho biến `s` (dòng lệnh 1), dùng hàm `int()` để chuyển đổi giá trị của biến `s` sang số nguyên và gán cho biến `n` (dòng lệnh 2). Hàm `input()` và `int()` là các hàm có sẵn trong Python.

```
1 s = input("Nhập số nguyên: ")
2 n = int(s)
```

Trong Python, các hàm có sẵn có thể được tập hợp lại trong tệp .py hoặc thư viện lập trình. Thư viện `math` trong Python cung cấp các hàm và hằng số toán học cần thiết để thực hiện các phép tính. Ví dụ, hàm `sqrt(x)` để tính căn bậc hai của `x`, hàm `pow(x,y)` để tính luỹ thừa bậc `y` của `x`, hàm `gcd(m,n)` để tính ước chung lớn nhất của hai số nguyên `m` và `n`,... Để có thể sử dụng được các hàm trong thư viện `math`, dùng lệnh `import math` hoặc `from math import *`.

Ví dụ 4: Trong Python, đoạn mã lệnh dưới đây khai báo và sử dụng thư viện `math` (dòng lệnh 1), nhập hai số nguyên `m` và `n` từ bàn phím (các dòng lệnh 2, 3); sau đó, sử dụng hàm có sẵn `gcd(m,n)` trong thư viện `math` để tính ước chung lớn nhất của hai số nguyên `m` và `n` (dòng lệnh 4).

Cách 1: Khai báo sử dụng thư viện `math` dùng lệnh `import math`.

```
1 import math
2 m = int(input("Nhập số nguyên m: "))
3 n = int(input("Nhập số nguyên n: "))
4 ucln = math.gcd(m,n)
5 print(f"Ước chung lớn nhất của {m} và {n} là {ucln}.")
```

Cách 2: Khai báo sử dụng thư viện `math` dùng lệnh `from math import *`.

```
1 from math import *
2 m = int(input("Nhập số nguyên m: "))
3 n = int(input("Nhập số nguyên n: "))
4 ucln = gcd(m,n)
5 print(f"Ước chung lớn nhất của {m} và {n} là {ucln}.")
```

Em hãy cho biết kết quả của các hàm có sẵn trong thư viện `math`: `pow(2,3)`, `gcd(8,16)`, `sqrt(16)`.

- Hàm có sẵn là hàm được tích hợp sẵn trong một ngôn ngữ lập trình.
- Python cung cấp một số hàm có sẵn và các thư viện. Lệnh `import` dùng để khai báo sử dụng các hàm trong thư viện. Thư viện `math` trong Python cung cấp các hàm có sẵn cần thiết của Toán học để thực hiện các phép tính.

3. Hàm tự định nghĩa

Trong lập trình, hàm tự định nghĩa là một hàm do lập trình viên tự viết để thực hiện một công việc.

Trong Python, hàm tự định nghĩa có dạng như sau:

```
def <tên hàm> ([<danh sách tham số>]):
    <khối lệnh>
```

Lệnh `def` dùng để định nghĩa hàm có tên là `<tên hàm>` và `<danh sách tham số>` của hàm (gồm các tham số phân cách nhau bởi dấu phẩy, có thể không có danh sách tham số), mỗi tham số là một biến.

Lời gọi hàm gồm `<tên hàm>` và `<danh sách đối số>` (gồm các đối số phân cách nhau bởi dấu phẩy), mỗi đối số có thể là một hằng, một biến hoặc một biểu thức. Lời gọi hàm có dạng như sau:

```
<tên hàm> ([<danh sách đối số>])
```

Chân trời sáng tạo

Nếu lệnh định nghĩa hàm không có <danh sách tham số> thì lời gọi hàm cũng không có <danh sách đối số>. Nếu lệnh định nghĩa hàm có <danh sách tham số> thì lời gọi hàm phải có <danh sách đối số>, hai danh sách này tương ứng với nhau về số lượng và thứ tự xuất hiện trong danh sách. Khi gọi hàm, giá trị của các đối số sẽ được gán cho các tham số tương ứng theo thứ tự xuất hiện trong danh sách.

<khối lệnh> gồm một hoặc nhiều lệnh, được gọi là thân hàm, dùng để thực hiện một công việc. Nếu thân hàm không có lệnh return thì quá trình thực thi của hàm sẽ kết thúc khi thực hiện xong lệnh ở dòng cuối cùng của thân hàm và hàm trả về None (giá trị rỗng) cho lệnh gọi hàm. Nếu thân hàm có lệnh return thì quá trình thực thi của hàm sẽ kết thúc khi thực hiện lệnh return.

Hàm trả về danh sách giá trị cho lời gọi hàm, gồm các giá trị phân cách nhau dấu phẩy, mỗi giá trị có thể một hằng, một biến hoặc một biểu thức. Lệnh return của hàm có dạng như sau:

```
return <danh sách giá trị>
```

Ví dụ 5: Quan sát chương trình sau đây, hàm trả về None vì lệnh return không có danh sách giá trị:

```
1 # Hàm in câu thông báo là xâu s
2 def inThongBao(s):
3     print(f"Câu thông báo: {s}.")
4     return
5 loiNgoaiLe = "Ngoại lệ SyntaxError"
6 inThongBao(loiNgoaiLe)      # Lời gọi hàm
7 cauTB = "Nhập số nguyên"
8 inThongBao(cauTB)           # Lời gọi hàm
```

Dòng lệnh 2 là lệnh định nghĩa hàm inThongBao(s) với tham số s. Các dòng lệnh 3, 4 là thân hàm. Dòng lệnh 6 là lời gọi hàm inThongBao(loiNgoaiLe), gán giá trị của đối số loiNgoaiLe cho tham số s và dòng lệnh 3 sẽ in giá trị của s ra màn hình. Dòng lệnh 4 là lệnh return trở về lời gọi hàm ở dòng lệnh 6, tiếp tục thực hiện dòng lệnh 7. Dòng lệnh 8 là lời gọi hàm inThongBao(cauTB), gán giá trị của đối số cauTB cho tham số s và dòng lệnh 3 sẽ in giá trị của s ra màn hình. Tiếp tục thực hiện dòng lệnh 4 là lệnh return trở về lời gọi hàm ở dòng 8 và kết thúc thực thi chương trình.

Ví dụ 6: Quan sát chương trình sau đây, hàm dienTichChuNhat() trả về một giá trị:

```
1 # Hàm tính diện tích hình chữ nhật có các độ dài x,y
2 def dienTichChuNhat(x,y):
3     dienTich = x*y      # Tính diện tích hình chữ nhật, độ dài x,y
4     return dienTich
5 dtcn = dienTichChuNhat(2,3)
```

Dòng lệnh 2 là lệnh định nghĩa hàm dienTichChuNhat(x,y) với x, y là các tham số. Các dòng lệnh 3, 4 là thân hàm. Dòng lệnh 4 là lệnh return và hàm trả về giá trị của biến

dienTich. Dòng lệnh 5 là lời gọi hàm dienTichChuNhat(2,3), gán giá trị 2 cho x và giá trị 3 cho y. Dòng lệnh 3 tính diện tích x*y và gán cho biến dienTich. Dòng lệnh 4 là lệnh return trả về giá trị của biến dienTich và gán cho biến dtcn.

Ví dụ 7: Quan sát chương trình sau đây, hàm nhapDuLieu() không có tham số dùng để nhập hai hệ số của phương trình bậc nhất và trả về hai hệ số này:

```
1 # Hàm nhập hai hệ số của phương trình bậc nhất
2 def nhapDuLieu():
3     x = float(input("Nhập hệ số thứ nhất: "))
4     y = float(input("Nhập hệ số thứ hai: "))
5     return x,y
6 a,b = nhapDuLieu()
7 print(f"Phương trình {a}x + {b} = 0")
```

Dòng lệnh 2 là lệnh định nghĩa hàm nhapDuLieu(). Các dòng lệnh 3 - 5 là thân hàm. Dòng lệnh 5 là lệnh return trả về hai giá trị của hai biến x và y. Dòng lệnh 6 là lời gọi hàm nhapDuLieu(). Dòng lệnh 3 - 4 sẽ nhập hai số thực, lần lượt gán cho hai biến x và y. Dòng lệnh 5 là lệnh return trả về giá trị của biến x và gán cho biến a, trả về giá trị của biến y và gán cho biến b. Sau đó, tiếp tục thực hiện dòng lệnh 7.

Quan sát hàm tinhTongSo() và hàm tinhTichSo() dưới đây, em hãy giải thích mỗi câu lệnh của hàm sẽ thực hiện công việc gì và ý nghĩa giá trị trả về của mỗi hàm.

```
def tinhTongSo(n):
    tongSo = 0
    m = len(str(n))
    for i in range(n):
        if i%2 == 0 and len(str(i)) == m:
            tongSo = tongSo+i
    return tongSo
def tinhTichSo(n):
    tichSo = 1
    s = str(n)
    for kt in s:
        tichSo = tichSo*int(kt)
    return tichSo
```

- Hàm tự định nghĩa là một hàm do lập trình viên tự viết để thực hiện một công việc.
- Lệnh def dùng để định nghĩa hàm và có thể có danh sách tham số.
- Lệnh return dùng để kết thúc quá trình thực thi của hàm và có thể trả về các giá trị cho lời gọi hàm.
- Lời gọi hàm có thể có các đối số, dùng để gọi thực hiện một hàm và có thể nhận các giá trị trả về của hàm để gán cho các biến.

Chân trời sáng tạo

LUYỆN TẬP

Viết chương trình nhập ba số nguyên dương từ bàn phím, tìm ước chung lớn nhất của ba số nguyên dương này và in kết quả ra màn hình.

Dữ liệu vào: Ba số nguyên dương.

Dữ liệu ra: Số là ước chung lớn nhất của ba số nguyên dương.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập ba số nguyên dương: 12 21 18`	`Ước chung lớn nhất là: 6`

Gợi ý: Sử dụng hàm `gcd()` trong thư viện `math` để tìm ước chung lớn nhất của hai số nguyên. Nhập vào ba số nguyên dương `a`, `b` và `c`. Tìm ước chung lớn nhất của `a`, `b` và gọi là `d`. Cuối cùng, tìm ước chung lớn nhất của `c`, `d` và in kết quả ra màn hình.

VẬN DỤNG

Để tính diện tích của một khu đất có hình dạng bất kì, ta chia nhỏ khu đất này thành các hình tam giác. Sử dụng công thức Heron để tính diện tích từng tam giác dựa trên độ dài các cạnh. Cuối cùng, tổng diện tích của các hình tam giác này là diện tích của khu đất. Dưới đây là công thức Heron để tính diện tích S của hình tam giác với độ dài của ba cạnh a, b, c và p là nửa chu vi của hình tam giác: $S = \sqrt{p(p-a)(p-b)(p-c)}$.

Yêu cầu: Viết chương trình tính diện tích của hình tam tam giác khi biết độ dài của ba cạnh.

Dữ liệu vào: Ba số thực dương là độ dài của ba cạnh.

Dữ liệu ra: Diện tích của hình tam giác.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị

`Nhập độ dài cạnh a: 3`
`Nhập độ dài cạnh b: 4`
`Nhập độ dài cạnh c: 5`
	`Diện tích của tam giác là: 6.00`

Gợi ý: Sử dụng hàm `sqrt()` trong thư viện math để tính căn bậc 2 của các số thực dương. Để `a`, `b`, `c` là ba cạnh của tam giác thì `a > 0, b > 0, c > 0` và tổng 2 cạnh bất kì phải lớn hơn cạnh còn lại. Áp dụng công thức Heron để tính diện tích tam giác và in kết quả ra màn hình.

Chân trời sáng tạo

BÀI F14 THAM SỐ CỦA HÀM VÀ PHẠM VI CỦA BIẾN

MỤC TIÊU **Sau bài học này, em sẽ:**

- Viết và sử dụng được hàm, hàm có tham số, đối số, biến cục bộ, biến toàn cục.

KHỞI ĐỘNG

Em hãy quan sát chương trình bên và cho biết kết quả hiển thị trên màn hình là gì?

```
1 def tong(a,b):
2     s = a+b if a > b else a-b
3     return s
4 a,b,s = 10,5,0
5 print(s)
6 print(tong(a,b))
7 print(s)
```

KHÁM PHÁ

1. Tham số, đối số, biến cục bộ, biến toàn cục

Tham số (parameter) là các biến được khai báo trong phát biểu định nghĩa hàm và sử dụng để chứa giá trị được truyền vào khi hàm được gọi. Tham số được sử dụng để xác định các dữ liệu cần thiết để hàm hoạt động.

Ví dụ 1: Khai báo và sử dụng tham số của hàm.

```
1 def nguyenTo(n): # Hàm kiểm tra n là số nguyên tố
2     kt = True
3     if n <= 1:
4         kt = False
5     else:
6         for i in range(2,int(n**0.5)+1):
7             if n%i == 0:
8                 kt = False
9     return kt
10 # Chương trình chính
11 k = int(input("Nhập số tự nhiên: "))
12 nt = nguyenTo(k) # Lệnh gọi hàm nguyenTo()
13 if nt:
14     print(f"{k} là số nguyên tố.")
15 else:
16     print(f"{k} không là số nguyên tố.")
```

Trong đó, dòng 1 có n được khai báo giữa cặp dấu ngoặc tròn () gọi là tham số. Nếu có nhiều hơn một tham số thì các tham số sẽ cách nhau bằng dấu phẩy.

Đối số (argument) là các giá trị được ghi trong lệnh gọi hàm và được gán cho các tham số tương ứng khi hàm được gọi. Đối số phản ánh các giá trị thực tế được chuyển cho hàm khi

Chân trời sáng tạo

được gọi và được gán cho các tham số tương ứng. Danh sách các đối số phải được cung cấp theo thứ tự như các tham số được liệt kê. Trong Ví dụ 1, giá trị `k` ở dòng 12 được gọi là đối số.

Theo phạm vi hoạt động, biến trong Python gồm biến cục bộ và biến toàn cục.

Biến cục bộ được tạo trong một hàm và có phạm vi trong hàm này từ khi được tạo, không thể truy xuất ở bên ngoài hàm này. Ở Ví dụ 1, biến `kt` ở dòng 2 là biến cục bộ của hàm `nguyenTo()` và có phạm vi trong hàm `nguyenTo()`.

Biến toàn cục được tạo ở bên ngoài mọi hàm hoặc được khai báo ở bên trong các hàm với từ khoá `global` và có phạm vi là toàn bộ chương trình từ khi được tạo. Ở Ví dụ 1, biến `k` ở dòng 11 được tạo ở bên ngoài mọi hàm, được gọi là biến toàn cục và có phạm vi trong toàn bộ chương trình.

Ví dụ 2: Xét hai chương trình sau:

Chương trình 1	Chương trình 2
1 `def chuNhat():` 2 `    s = a*b` 3 `    p = (a+b)*2` 4 `#Các lệnh gọi hàm` 5 `a,b = map(int,input().split())` 6 `chuNhat()` 7 `print(s,p)`	1 `def chuNhat():` 2 `    s = a*b` 3 `    p = (a+b)*2` 4 `    print(s,p)` 5 `#Chương trình chính` 6 `a,b = map(int,input().split())` 7 `chuNhat()`

Ở Ví dụ 2, chương trình 1 sẽ báo lỗi ở dòng 7 vì `s`, `p` là biến cục bộ nên chỉ có phạm vi sử dụng trong hàm `chuNhat()`. Biến a, b là biến toàn cục nên có thể được sử dụng ở chương trình chính và hàm `chuNhat()`. Tuy nhiên, nếu trong hàm `chuNhat()` có lệnh gán giá trị cho biến `a` thì biến `a` này là biến cục bộ và không phải là biến toàn cục `a`, mặc dù có cùng tên `a`. Sau khi kết thúc thực hiện hàm `chuNhat()` và trở về lệnh gọi thì giá trị của biến toàn cục `a` vẫn không thay đổi.

Khi sử dụng biến toàn cục bên trong hàm và muốn thay đổi giá trị của nó trong quá trình thực hiện hàm, thì phải khai báo biến toàn cục theo cú pháp: `global <danh sách biến toàn cục>`.

Ví dụ 3: Chương trình dùng biến toàn cục có sự thay đổi giá trị:

```
def kiemTra():
    global s            # Khai báo s là biến toàn cục
    if n%2 == 0:
        s = "Số chẵn" # Gán giá trị mới cho biến s
    else:
        s = "Số lẻ"   # Gán giá trị mới cho biến s
# Chương trình chính
n = int(input("Nhập số nguyên: "))
s = ""                  # Biến s là xâu rỗng
kiemTra()
print(s)                # Biến s khác xâu rỗng
```

1. Hãy cho biết sự khác biệt khi dùng từ khoá global cho biến toàn cục.

2. Hãy cho biết kết quả in ra màn hình của chương trình sau:

```
1 def f(x,y):
2     n = 3
3     a = n+2*x+y
4     print(n)
5     return a
6 n = 7
7 print(f(2,3))
8 print(n)
```

- Tham số là các biến được khai báo trong phát biểu định nghĩa hàm và được sử dụng để chứa giá trị được truyền vào khi hàm được gọi.
- Đối số là các giá trị được ghi trong lệnh gọi hàm và được gán cho các tham số tương ứng khi hàm được gọi.
- Biến cục bộ được tạo trong một hàm và có phạm vi trong hàm này từ khi được tạo.
- Biến toàn cục được tạo ở bên ngoài mọi hàm hoặc được khai báo ở bên trong các hàm với từ khoá `global` và có phạm vi là toàn bộ chương trình từ khi được tạo.

2. Cách sử dụng hàm

Trong ngôn ngữ lập trình, viết hàm giúp chương trình trở nên dễ đọc, dễ quản lí hơn, tăng hiệu suất, dễ dàng bảo trì và nâng cấp chương trình. Một số lợi ích như cho phép tái sử dụng hàm nhiều lần trong chương trình, tổ chức chương trình thành các phần nhỏ, dễ quản lí hơn.

Hàm trong Python là một khối lệnh thực hiện một nhiệm vụ cụ thể và có thể được sử dụng ở nhiều vị trí khác nhau trong chương trình thông qua lệnh gọi hàm.

Bài toán 1. Kiểm tra số đẹp

Giờ giải lao, Dương giới thiệu một định nghĩa về số đẹp như sau: "Số đẹp là số nguyên tố có tổng các chữ số là số nguyên tố và các chữ số cũng là số nguyên tố.". Chẳng hạn, số 2 357 là số đẹp vì số 2 357 là số nguyên tố; tổng của các số 2, 3, 5, 7 là 17 cũng là số nguyên tố và các chữ số 2, 3, 5, 7 là các số nguyên tố. Sau đó, Dương đố các bạn là các số ghi trên bảng có phải "số đẹp" hay không?

Yêu cầu: Hãy viết chương trình giúp các bạn kiểm tra số ghi trên bảng có phải số đẹp hay không.

Dữ liệu vào: Số nguyên dương n.

Dữ liệu ra: Nếu n là số đẹp thì in ra màn hình chữ "Có" và tổng các chữ số của n. Nếu n không phải là số đẹp thì in ra màn hình chữ "Không" và giá trị làm cho n không phải là số đẹp theo thứ tự ưu tiên: số n không nguyên tố, tổng các chữ số của n không nguyên tố, từng chữ số của n theo thứ tự xuất hiện không nguyên tố.

Bảng bên minh hoạ cho kết quả chạy chương trình.

Thông tin nhập	Kết quả hiển thị
23753	Không 20

Hướng dẫn: Định nghĩa hàm `nguyenTo(n)` để kiểm tra `n` là số nguyên tố. Kiểm tra lần lượt các điều kiện số đẹp theo thứ tự ưu tiên: số `n` là số nguyên tố, tổng các chữ số của `n` là số nguyên tố, các chữ số của `n` cũng là số nguyên tố. Cụ thể như sau:

1. Định nghĩa hàm `nguyenTo(n)`: # Dùng để kiểm tra số nguyên tố

   Nếu `n == 1` thì trả về False

   Duyệt `i` từ `2` đến `int`(căn bậc hai của `n`) `+ 1`:

   Nếu `n%i == 0` thì

   Trả về False

   Trả về True

2. Định nghĩa hàm `tongChuSo(n)`: # Dùng để tính tổng các chữ số của n

   `s = 0`

   Cho mỗi chữ số `i` trong `n`:

   `s += int(i)`

   Trả về s

3. Định nghĩa hàm `soDep(n)`: # Dùng để kiểm tra một số có phải số đẹp không

   # Kiểm tra xem n có phải là số nguyên tố không

   Nếu không phải `nguyenTo(int(n))` thì

   Trả về "Không", `n`

   # Kiểm tra xem tổng có phải là số nguyên tố không

   Nếu không phải `nguyenTo(tongChuSo(n))` thì

   Trả về "Không", `tongChuSo(n)`

   # Kiểm tra xem từng chữ số có phải là số nguyên tố không

   `cacSoNguyenTo = ['2','3','5','7']`

   Cho mỗi kí tự i trong n:

   Nếu `i` không thuộc `cacSoNguyenTo` thì

   Trả về "Không", `i`

   # Nếu là số đẹp

   Trả về "Có", `tongChuSo(n)`

4. Chương trình chính

   Nhập `n`

   In ra màn hình `soDep(n)`

Chân trời sáng tạo

Chương trình tham khảo:

```
1 def nguyenTo(n):
2     if n == 1:
3         return False
4     for i in range(2,int(n**0.5)+1):
5         if n%i == 0:
6             return False
7     return True
8
9 def tongChuSo(n):
10     s = 0
11     for i in n:
12         s += int(i)
13     return s
14
15 def soDep(n):
16     # Kiểm tra xem n có phải là số nguyên tố không
17     if not nguyenTo(int(n)):
18         return "Không",n
19     # Kiểm tra xem tổng có phải là số nguyên tố không
20     if not nguyenTo(tongChuSo(n)):
21         return "Không",tongChuSo(n)
22     # Kiểm tra xem từng chữ số có phải là số nguyên tố không
23     cacSoNguyenTo = ['2','3','5','7']
24     for i in n:
25         if i not in cacSoNguyenTo:
26             return "Không",i
27     # Nếu là số đẹp
28     return "Có",tongChuSo(n)
29
30 n = input() #Nhập số n
31 ketQua,liDo = soDep(n)
32 print(ketQua,liDo)
```

Bài toán 2: Một cửa hàng tổ chức chương trình tri ân cho các khách hàng thân thiết. Mỗi người nhận được một phiếu bốc thăm có mã phiếu gồm các kí tự chữ và số. Căn cứ vào các số trong mã phiếu (không xét các kí tự chữ) để xác định thứ tự ưu tiên của khách hàng lên nhận thưởng: sắp thứ tự tăng dần theo số trong mã phiếu.

Yêu cầu: Hãy viết chương trình xác định trình tự của các khách hàng bốc thăm theo phiếu họ nhận được.

Dữ liệu vào: Một số nguyên dương là số lượng các khách hàng được bốc thăm và các kí tự trên phiếu bốc thăm của các khách hàng nhận được.

Dữ liệu ra: Các phiếu được xếp theo thứ tự bốc thăm của các khách hàng.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
Nhập số lượng phiếu: 4 Nhập mã phiếu: EB34FO Nhập mã phiếu: ACT14AVC Nhập mã phiếu: 1546KHA Nhập mã phiếu: TGA567	Trình tự của các khách hàng bốc thăm: ACT14AVC EB34FO TGA567 1546KHA

Hướng dẫn: Số phiếu của mã phiếu là số gồm các chữ số có trong mã phiếu theo thứ tự xuất hiện. Sử dụng một danh sách chứa số phiếu và mã phiếu đã nhập. Sau đó, sắp xếp danh sách này theo thứ tăng dần các số phiếu. Cuối cùng, hiển thị các mã phiếu đã được sắp xếp của danh sách này. Cụ thể như sau:

1. Định nghĩa hàm `catSo(maPhieu)`:

   Khởi tạo xâu `soPhieu` bằng rỗng.

   Duyệt `kiTu` trong xâu `maPhieu`:

   Nếu `kiTu` là số thì:

   Thêm `kiTu` vào cuối xâu `soPhieu`.

   Trả về số nguyên của xâu `soPhieu`.

2. Chương trình chính

   Nhập số lượng phiếu `n`.

   Khởi tạo danh sách `dsPhieu` bằng rỗng.

   Lặp `n` lần:

   Nhập mã phiếu `maPhieu`.

   Tính giá trị `soPhieu` bằng cách gọi `catSo(maPhieu)`.

   Thêm `[soPhieu, maPhieu]` vào cuối danh sách `dsPhieu`.

   Sắp xếp danh sách `dsPhieu`.

   Với mỗi mã phiếu `maPhieu` trong danh sách `dsPhieu`:

   Hiển thị mã phiếu `maPhieu` ra màn hình.

Chương trình tham khảo:

```
def catSo(maPhieu):
    soPhieu = ""
    for kiTu in maPhieu:
        if kiTu.isdigit():
            soPhieu = soPhieu+kiTu
    return int(soPhieu)

n = int(input("Nhập số lượng phiếu: "))
dsPhieu = []
for i in range(n):
    maPhieu = input("Nhập mã phiếu: ")
    soPhieu = catSo(maPhieu)
    dsPhieu.append([soPhieu,maPhieu])
dsPhieu.sort()                # Sắp xếp danh sách dsPhieu
print("Trình tự của các khách hàng bốc thăm:")
for phieu in dsPhieu:
    soPhieu,maPhieu = phieu
    print(maPhieu)
```

Hãy nêu ví dụ về một bài toán trong thực tế có sử dụng hàm để viết chương trình.

Trong ngôn ngữ lập trình, viết hàm giúp chương trình trở nên dễ đọc, dễ quản lí, tăng hiệu suất, dễ dàng bảo trì và nâng cấp chương trình. Ngoài ra, sử dụng hàm có thể giúp phân chia bài toán lớn thành các bài toán con nhỏ hơn.

LUYỆN TẬP

1. Biến được khai báo trong hàm gọi là:

A. Tham số. B. Đối số.

C. Biến cục bộ. D. Biến toàn cục.

2. Giá trị được truyền cho hàm khi gọi hàm được gọi là:

A. Tham số. B. Đối số.

C. Biến cục bộ. D. Biến toàn cục.

3. Hãy xác định biến toàn cục, biến cục bộ, tham số, đối số của chương trình sau:

```
1 def giaiThua(n):
2     gt = 1
3     for i in range(2,n+1):
4         gt *= i
5     return gt
6 n = int(input("Nhập số n: "))
7 print(f'Giai thừa của {n} là {giaiThua(n)}.')
```

VẬN DỤNG

Sau khi đã được học về biến toàn cục và biến cục bộ, Khánh muốn sử dụng biến toàn cục trong chương trình tính tổng các số nguyên tố nhỏ hơn hoặc bằng n, với n là số tự nhiên lớn hơn 2. Ví dụ, với n = 10, các số nguyên tố nhỏ hơn hoặc bằng 10 là 2, 3, 5, 7. Tổng các số nguyên tố này là 17.

Yêu cầu: Viết chương trình tính tổng các số nguyên tố nhỏ hoặc bằng n, với n là số tự nhiên lớn hơn 2, có sử dụng biến toàn cục.

Dữ liệu vào: Số tự nhiên n > 2.

Dữ liệu ra: Tổng S của các số nguyên tố nhỏ hơn hoặc bằng n.

BÀI F15 THỰC HÀNH HÀM TRONG PYTHON

MỤC TIÊU **Sau bài học này, em sẽ:**

- Viết được chương trình có sử dụng chương trình con đã tạo.

KHỞI ĐỘNG

Hãy chỉ ra tham số, đối số của hàm `tinh()` trong chương trình dưới đây:

```
def tinh(n,m):
    while n != m:
        if n > m:
            n = n-m
        else:
            m = m-n
    return n,m
a,b = map(int,input("Nhập 2 số nguyên a,b: ").split())
n,m = tinh(a,b)
print(f"n = {n},m = {m}")
```

THỰC HÀNH

Nhiệm vụ 1. Dùng hàm để tính biểu thức

Yêu cầu: Viết chương trình tính biểu thức S.

$$S = 1^2 + 1^2 \times 2^2 + 1^2 \times 2^2 \times 3^2 + 1^2 \times 2^2 \times 3^2 \times 4^2 + \ldots + 1^2 \times 2^2 \times \ldots \times (n-1)^2 \times n^2.$$

Dữ liệu vào: Số nguyên dương n.

Dữ liệu ra: Giá trị của biểu thức S.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập số nguyên dương n: 5`	`Giá trị của biểu thức S: 15017`

Hướng dẫn: Trong công thức tính `S`, ta nhận thấy số hạng đầu tiên là `1`, số hạng thứ `i` (với `2 ≤ i ≤ n`) bằng số hạng liền trước nhân với `i` bình phương. Cụ thể như sau:

1. Định nghĩa hàm `tinhBieuThucS(n)`:

```
S = 0
tich = 1
Vòng lặp i từ 1 đến n:
    tich = tich*i**2
    S = S+tich
Trả về S
```

❷ Chương trình chính

Nhập số nguyên dương `n`

Hiển thị giá trị `tinhBieuThucS(n)`

Chương trình tham khảo:

```
def tinhBieuThucS(n):
    S = 0
    tich = 1
    for i in range(1,n+1):
        tich = tich*i**2
        S = S+tich
    return S
# Chương trình chính
n = int(input("Nhập số nguyên dương n: "))
print("Giá trị của biểu thức S:",tinhBieuThucS(n))
```

Nhiệm vụ 2. Tìm độ dài lớn nhất của danh sách bằng phẳng

Một danh sách bằng phẳng có tất cả các phần tử đều bằng nhau. Danh sách có một phần tử được gọi là danh sách bằng phẳng có độ dài bằng 1.

Yêu cầu: Cho danh sách A gồm các số nguyên. Hãy tìm độ dài lớn nhất của các danh sách bằng phẳng (các phần tử không cần phải liên tục) có trong danh sách A.

Dữ liệu vào: Danh sách A gồm các số nguyên.

Dữ liệu ra: Độ dài lớn nhất của các danh sách bằng phẳng.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập danh sách A: 1 1 3 2 2 3 3 3 2`	`Độ dài lớn nhất: 4`

Hướng dẫn: Độ dài của một danh sách bằng phẳng có giá trị x là số lần xuất hiện x trong danh sách A. Với mỗi giá trị x trong danh sách A, tìm độ dài của danh sách bằng phẳng có giá trị này. Từ đó, ta tìm được độ dài lớn nhất.

❶ Định nghĩa hàm `demPhanTu(A)`:

Khởi tạo biến `doDaiLonNhat = 0`

Khởi tạo danh sách `demA` = danh sách rỗng

Duyệt từng phần tử `x` thuộc danh sách A:

  Nếu `x` không thuộc danh sách `demA`:

    Thêm `x` vào danh sách `demA`

    `dem` = số lần xuất hiện của `x` trong danh sách `A`

    Nếu `doDaiLonNhat < dem`:

      `doDaiLonNhat = dem`

Trả về `doDaiLonNhat`

❷ **Chương trình chính**

Nhập danh sách `A` từ bàn phím.

Hiển thị giá trị `demPhanTu(A)` ra màn hình.

Chân trời sáng tạo

Chương trình tham khảo:

```
def demPhanTu(A):
    doDaiLonNhat = 0
    demA = []                    # demA lưu các phần tử đã đếm
    for x in A:
        if x not in demA:
            demA.append(x)       # Thêm x vào demA
            dem = A.count(x) # Đếm số lần xuất hiện của x trong A
            if doDaiLonNhat < dem:
                doDaiLonNhat = dem  # Cập nhật doDaiLonNhat
    return doDaiLonNhat

# Chương trình chính
A = list(map(int,input("Nhập danh sách A: ").split()))
print(f"Độ dài lớn nhất: {demPhanTu(A)}")
```

Nhiệm vụ 3. Đếm số lượng số nguyên tố Fibonacci trong đoạn [a,b]

Dãy Fibonacci là dãy vô hạn các số tự nhiên bắt đầu bằng 0 và 1, các số sau đó được xác định theo quy tắc mỗi số bằng tổng hai số liền trước. Dãy số Fibonacci được định nghĩa như sau:

$$f_0 = 0;$$
$$f_1 = 1;$$
$$f_n = f_{n-1} + f_{n-2} \text{ với } n \geq 2.$$

Các số đầu tiên của dãy Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144,...

Một số Fibonacci, đồng thời cũng là số nguyên tố, được gọi là số nguyên tố Fibonacci. Ví dụ, 2, 3, 5, 13, 89, 233,...

Yêu cầu: Viết chương trình in ra màn hình số lượng số nguyên tố Fibonacci trong đoạn `[a,b]`.

Dữ liệu vào: Hai số tự nhiên `a, b (a ≤ b)`.

Dữ liệu ra: Số lượng số nguyên tố Fibonacci trong đoạn `[a,b]`.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập 2 số nguyên dương a <= b:` `2 8`	`Số lượng số nguyên tố Fibonacci trong đoạn [2,8]: 3.`

Hướng dẫn: Ban đầu, biến đếm bằng `0`. Xét các số Fibonacci f_i trong đoạn `[a,b]` với `i ≥ 0`, nếu f_i là số nguyên tố thì tăng biến đếm lên `1`. Cuối cùng, in giá trị của biến đếm ra màn hình.

1. Hàm `nguyenTo(n)`: # Kiểm tra n là số nguyên tố
   Nếu `n <= 1` thì:
      Trả về False
   Vòng lặp `i` từ `2` đến (căn bậc `2` của `n`)+1:
      Nếu `n` chia hết cho `i` thì:
         Trả về False
   Trả về True

2 Hàm `demFiboNguyenTo(a,b)`: # Đếm số nguyên tố Fibonacci

```
f0 = 0
f1 = 1
f2 = f1+f0
c = 0
```

Vòng lặp trong khi `f2 <= b`:

Nếu `a <= f2 <= b` và `isPrime(f2)` thì:

```
    c = c+1
f0 = f1
f1 = f2
f2 = f0+f1
```

Trả về c

3 Chương trình chính.

Nhập 2 số nguyên dương và gán cho a và b.

```
n = demFiboNguyenTo(a,b)
```

Hiển thị số lượng số nguyên tố Fibonacci là n.

Chương trình tham khảo:

```
def nguyenTo(n):
    if n <= 1:
        return False
    for i in range(2,int(n**0.5)+1):
        if n%i == 0:
            return False
    return True
def demFiboNguyenTo(a,b):
    f0 = 0
    f1 = 1
    f2 = f1+f0
    c = 0
    while f2 <= b:
        if a <= f2 <= b and nguyenTo(f2):
            c = c+1
        f0 = f1
        f1 = f2
        f2 = f0+f1
    return c
# Chương trình chính
a,b = map(int,input("Nhập 2 số nguyên dương a <= b: ").split())
n = demFiboNguyenTo(a,b)
print(f"Số lượng số nguyên tố Fibonacci trong đoạn [{a},{b}]: {n}.")
```

Chân trời sáng tạo

LUYỆN TẬP

Nhiệm vụ 4. In tất cả các số Lucas trong đoạn [a,b]

Dãy số Lucas là dãy vô hạn các số tự nhiên bắt đầu bằng 2 và 1, các số sau đó được xác định theo quy tắc mỗi số bằng tổng hai số liền trước. Dãy số Lucas được định nghĩa như sau:

$$L_0 = 2; L_1 = 1; L_n = L_{n-1}+L_{n-2} \text{ với } n \geq 2.$$

Yêu cầu: Viết chương trình in ra màn hình tất cả các số Lucas trong đoạn `[a,b]`.

Dữ liệu vào: Hai số nguyên dương `a`, `b`.

Dữ liệu ra: Các số Lucas trong đoạn `[a,b]` cách nhau một khoảng trắng.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập 2 số nguyên dương a <= b:` `3 18`	`Các số Lucas trong đoạn [3, 18]:` `3, 4, 7, 11, 18`

VẬN DỤNG

Nhiệm vụ 5. Tính tổng bình phương các ước thực sự của một số nguyên dương

Yêu cầu: Viết chương trình tính tổng bình phương các ước thực sự của một số nguyên dương.

Dữ liệu vào: Số nguyên dương.

Dữ liệu ra: Tổng bình phương các ước thực sự của số nguyên dương.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập số nguyên dương: 8`	`Tổng bình phương các ước thực sự của 8 là: 21`

Nhiệm vụ 6. Tìm độ dài lớn nhất của danh sách con các phần tử liên tục và không giảm trong danh sách A

Yêu cầu: Cho danh sách `A` gồm các số nguyên. Hãy tìm độ dài lớn nhất của danh sách con các phần tử liên tục trong danh sách `A` có giá trị không giảm.

Dữ liệu vào: Danh sách các số nguyên `A`.

Dữ liệu ra: Độ dài lớn nhất của danh sách con `B` các phần tử liên tục và không giảm trong danh sách `A`.

Ví dụ minh hoạ cho kết quả chạy chương trình:

Thông tin nhập	Kết quả hiển thị
`Nhập danh sách A: 1 1 3 2 2 3 3 3 2`	`Độ dài lớn nhất: 5`

Gợi ý: Khởi tạo biến `doDaiLonNhat = 0`. Khởi tạo danh sách `B` và thêm phần tử đầu tiên của danh sách `A` vào danh sách `B`. Với mỗi `A[i]` là phần tử của danh sách `A`, nếu `A[i]` không nhỏ hơn phần tử `B[-1]` (phần tử cuối cùng của danh sách `B`) thì thêm `A[i]` vào `B`. Ngược lại, nếu `A[i]` nhỏ hơn phần tử `B[-1]`, so sánh `len(B)` với `doDaiLonNhat` để cập nhật `doDaiLonNhat` (nếu `doDaiLonNhat < len(B)`), xoá danh sách `B` và thêm phần tử `A[i]` vào danh sách `B`.

Chân trời sáng tạo

BÀI F16 KIỂM THỬ VÀ GỠ LỖI CHƯƠNG TRÌNH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Biết và phân loại được một số lỗi thường gặp trong chương trình.
- Biết được một số cách đơn giản để kiểm thử và gỡ lỗi chương trình.
- Sử dụng được các chức năng kiểm thử và gỡ lỗi chương trình trong môi trường lập trình Python.

KHỞI ĐỘNG

Cho chương trình tìm giá trị lớn nhất trong mảng A:

```
def timGTLonNhat(A):
    gtLonNhat = 0
    for i in range(len(A)):
        if gtLonNhat < A[i]:
            gtLonNhat = A[i]
    return gtLonNhat
A = list(map(int,input("Nhập mảng A: ").split()))
print(f"Giá trị lớn nhất là {timGTLonNhat(A)}.")
```

a) Theo em, chương trình trên có lỗi không?

b) Cho `A = [12,17]` hay `A = []`. Em hãy cho biết kết quả được in trên màn hình có đúng với yêu cầu của bài toán không.

KHÁM PHÁ

1. Một số lỗi thường gặp trong chương trình

a) Phân loại lỗi của chương trình

Trong lập trình, không thể tránh khỏi việc phát sinh một số lỗi khi viết các chương trình. Thông thường, các loại lỗi thường gặp có thể được phân loại như sau:

– **Lỗi cú pháp** là lỗi sai do viết câu lệnh bị sai với cú pháp của ngôn ngữ lập trình. Các lỗi sai này được thông báo khi biên dịch lệnh hoặc chương trình. Ví dụ: Trong Python, lệnh `s = "Tin học'` bị lỗi SyntaxError vì cặp dấu bắt đầu và kết thúc xâu kí tự khác nhau.

– **Lỗi thời gian chạy** là lỗi sai xảy ra khi một lệnh không thể được thực hiện trong thời gian chạy của chương trình. Ví dụ: Trong Python, khi thực hiện chương trình đến lệnh `a = 1 / b`, nếu `b` bằng `0` thì lệnh này bị lỗi ZeroDivisionError vì không thể chia cho 0.

– **Lỗi logic** là lỗi sai trong chương trình làm cho chương trình tạo ra kết quả không mong đợi hoặc chương trình bị lặp vòng vô tận. Thông thường, khó tìm ra lỗi logic ở các lệnh nào của chương trình. Ví dụ: Trong chương trình giải phương trình $ax^2 + bx + c = 0$, lệnh `delta = b**2+4*a*c` gây ra lỗi sai logic vì lệnh đúng là `delta = b**2-4*a*c`.

b) Một số thông báo lỗi thường gặp

Trong lúc lập trình, khi gặp các lỗi cú pháp hoặc lỗi thời gian chạy, máy tính sẽ dừng lại và thông báo lỗi để người viết chương trình có thể kịp thời phát hiện và sửa chữa. Dưới đây là một số thông báo lỗi thường gặp:

– **SyntaxError** là lỗi do viết lệnh bị sai với cú pháp. Ví dụ, lệnh `a = 1.o` bị lỗi SyntaxError vì một giá trị số không được có chữ cái và sửa lại đúng là `a = 1.0`.

```
>>> a = 1.o
SyntaxError: invalid decimal literal
```

– **NameError** là lỗi do viết một tên mà tên này không có. Ví dụ, lệnh `prin("Chào bạn")` bị lỗi NameError vì không có tên hàm `prin` và sửa lại đúng là `print("Chào bạn")`.

```
>>> prin("Chào bạn")
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    prin("Chào bạn")
NameError: name 'prin' is not defined. Did you mean: 'print'?
```

– **IndexError** là lỗi do chỉ số nằm ngoài miền giá trị. Ví dụ, `A = [6,7,8]`; `print(A[4])`, lệnh print bị lỗi IndexError vì không có chỉ số 4 trong mảng A.

```
>>> A=[6,7,8]
>>> print(A[4])
Traceback (most recent call last):
  File "<pyshell#3>", line 1, in <module>
    print(A[4])
IndexError: list index out of range
```

– **TypeError** là lỗi do sai quy định về kiểu dữ liệu. Ví dụ, lệnh `a = 1 + "234"` bị lỗi TypeError vì không thể cộng một giá trị số với một xâu kí tự.

```
>>> a = 1+"234"
Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    a = 1+"234"
TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

– **ValueError** là lỗi do giá trị không hợp lệ. Ví dụ, lệnh `a = int("12o")` bị lỗi ValueError vì không thể chuyển một xâu kí tự có chữ cái thành một số nguyên và sửa lại đúng là `a = int("120")`.

```
>>> a = int("12o")
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    a = int("12o")
ValueError: invalid literal for int() with base 10: '12o'
```

Chân trời sáng tạo

– **IndentationError** là lỗi do thụt đầu dòng không đúng quy định. Ví dụ, trong đoạn mã lệnh dưới đây, lệnh print ở dòng lệnh 2 bị lỗi IndentationError vì lệnh này phải trong vòng lặp for và phải thụt đầu dòng vào bên trong vòng lặp, nghĩa là thụt đầu dòng qua phải một số khoảng trắng.

```
>>> for i in range(5):
...print(i)
    SyntaxError: expected an indented block after 'for' statement on line 1
```

1. Theo em, khi biên dịch các đoạn chương trình sau, sẽ xuất hiện các lỗi nào?

A. `a = 2.x`

B. `a = 0`
`b = 1`
`c = b // a`

C. `a = 1.0 + "2.0"`

D. `printf("Xin chao")`

2. Em hãy trình bày và cho ví dụ về sự khác biệt giữa lỗi cú pháp và lỗi thời gian chạy.

3. Theo em, giữa lỗi thời gian chạy và lỗi logic, em hãy cho biết lỗi nào khó phát hiện hơn. Cho ví dụ minh hoạ.

4. Cho A là một danh sách bất kì, câu lệnh `print(sum(A))` có thể gặp thông báo lỗi nào? Cho ví dụ minh hoạ.

Một số lỗi có thể gặp trong quá trình lập trình: Lỗi cú pháp xảy ra khi viết câu lệnh sai cú pháp của ngôn ngữ lập trình; Lỗi thời gian chạy xảy ra khi một lệnh không thể được thực hiện trong thời gian chạy chương trình; Lỗi logic xảy ra khi chương trình tạo ra kết quả không mong đợi và là loại lỗi khó phát hiện nhất.

2. Kiểm thử chương trình và gỡ lỗi

Các lỗi cú pháp, hay lỗi thời gian chạy làm cho chương trình dừng lại và báo lỗi, vì vậy có thể phát hiện và sửa lệnh đã gây ra lỗi. Tuy nhiên, đối với lỗi logic, thì chương trình không dừng lại nhưng tạo ra kết quả không mong đợi. Vì vậy, sau khi viết xong chương trình, cần phải chạy với các bộ dữ liệu đầu vào khác nhau để kiểm tra và phát hiện lỗi logic. Quá trình kiểm tra này được gọi là *kiểm thử chương trình*. Mỗi bộ dữ liệu đầu vào dùng cho một lần chạy chương trình được gọi là *trường hợp kiểm thử*. Tập hợp các trường hợp kiểm thử được gọi là *bộ kiểm thử* cho chương trình.

Tuỳ theo các trường hợp kiểm thử cụ thể mà có thể phát hiện một số lỗi trong chương trình. Nếu phát hiện lỗi trong chương trình thì cần phải tìm ra lệnh bị lỗi, lệnh gây ra lỗi và sửa lỗi để chương trình cho ra kết quả đúng. Quá trình này được gọi là *gỡ lỗi chương trình*.

Ví dụ 1: Xét chương trình tìm nghiệm của phương trình ax + b = 0 như sau:

```
1 a = float(input('Nhập a: '))
2 b = float(input('Nhập b: '))
3 x = b//a
4 print('Nghiệm x = ',x)
```

Chương trình trên gây ra lỗi nếu nhập các giá trị xâu kí tự không phải dạng số như `'ab'` cho các biến `a`, `b` hay giá trị 0 cho biến `a`. Nếu nhập giá trị khác 0 cho `a` thì chương trình này không gây ra lỗi, nhưng xảy ra lỗi logic ở dòng lệnh 3 (phép tính không đúng và sử dụng phép chia nguyên), vì lệnh đúng phải là $x = -\frac{b}{a}$. Chương trình đúng như sau:

```
1 a = float(input('Nhập a: '))
2 b = float(input('Nhập b: '))
3 x = -b/a
4 print('Nghiệm x = ',x)
```

a) Kiểm thử chương trình

Để kiểm thử một chương trình, cần phải phân loại các trường hợp kiểm thử sao cho bao quát các giá trị đầu vào thông thường và các giá trị nhập ở ngưỡng để kiểm tra các giới hạn của chương trình. Các tổ hợp đặc biệt của dữ liệu nhập để kiểm tra chương trình được dùng để thực hiện việc kiểm tra các khía cạnh riêng biệt của bài toán cần giải quyết.

Bộ kiểm thử phải bao gồm các trường hợp: thông thường (ví dụ kiểm thử với hai giá trị số thực a, b nhỏ và khác 0 trong Ví dụ 1 để kiểm tra nghiệm đúng); trường hợp cận biên như giá trị lớn nhất, nhỏ nhất của dữ liệu nhập (ví dụ như kiểm thử với giá trị nhập cho hai biến `a` và `b` rất lớn trong Ví dụ 1); trường hợp đặc biệt (ví dụ như danh sách rỗng hay chỉ có một phần tử).

Ví dụ 2: Đối với chương trình ở KHỞI ĐỘNG. Xét các bộ kiểm thử sau đây:

– Bộ kiểm thử 1 bao gồm các trường hợp kiểm thử là TC1: `A = [5]`; TC2: `A = [2,5,4]`. Chương trình này đều cho kết quả đúng. Bộ kiểm thử này không phát hiện lỗi trong chương trình.

– Bộ kiểm thử 2 bao gồm các trường hợp kiểm thử là TC3: `A = [2,5]`; TC4: `A = []`. Với trường hợp kiểm thử TC4, chương trình này cho kết quả là 0 và bị lỗi logic vì không thể có giá trị lớn nhất trong mảng rỗng; phải thông báo "Mảng rỗng" với lỗi ValueError và kết thúc chương trình.

– Bộ kiểm thử 3 bao gồm các trường hợp kiểm thử là TC5: `A = [2,5,4,3]`; TC6: `A = [-2]`. Với trường hợp kiểm thử TC6, chương trình này cho kết quả là 0 và bị lỗi logic vì kết quả đúng là -2.

b) Gỡ lỗi chương trình

Để gỡ lỗi chương trình, cần phải gỡ lỗi cho từng trường hợp kiểm thử phát hiện lỗi. Chẳng hạn, Ví dụ 3, 4 minh hoạ quá trình gỡ lỗi chương trình trong KHỞI ĐỘNG cho các trường hợp kiểm thử TC4 và TC6.

Ví dụ 3: Với trường hợp kiểm thử TC4: `A = []`, cần phải kiểm tra mảng rỗng và thông báo lỗi trước khi tìm giá trị lớn nhất bằng lệnh if và lệnh raise. Chương trình được viết lại như sau:

```
1 def timGTLonNhat(A):
2     if A == []:                                  # Gỡ lỗi cho TC4
3         raise ValueError("Mảng rỗng.")           # Gỡ lỗi cho TC4
4     gtLonNhat = 0
5     for i in range(len(A)):
6         if gtLonNhat < A[i]:
7             gtLonNhat = A[i]
8     return gtLonNhat
9 A = list(map(int,input("Nhập mảng A: ").split()))
10 print(f"Giá trị lớn nhất là {timGTLonNhat(A)}.")
```

Với trường hợp kiểm thử TC6: `A = [-2]`, lệnh gây ra lỗi là lệnh `gtLonNhat = 0` vì mảng A chỉ có một giá trị âm và lệnh đúng là `gtLonNhat = A[0]`. Chương trình được viết lại như sau:

```
1 def timGTLonNhat(A):
2     if A == []:                                  # Gỡ lỗi cho TC4
3         raise ValueError("Mảng rỗng.")           # Gỡ lỗi cho TC4
4     gtLonNhat = A[0]                             # Gỡ lỗi cho TC6
5     for i in range(len(A)):
6         if gtLonNhat < A[i]:
7             gtLonNhat = A[i]
8     return gtLonNhat
9 A = list(map(int,input("Nhập mảng A: ").split()))
10 print(f"Giá trị lớn nhất là {timGTLonNhat(A)}.")
```

Vì sao cần có nhiều bộ dữ liệu khác nhau để kiểm thử chương trình?

Kiểm thử chương trình là chạy chương trình với các bộ dữ liệu đầu vào khác nhau để kiểm tra và phát hiện lỗi logic. Bộ dữ liệu kiểm thử cần phải bao gồm các trường hợp: thông thường, đặc biệt và cận biên.

3. Cách tìm lỗi trong chương trình

Phần này trình bày hai cách tìm lỗi trong chương trình Python: thêm các lệnh `print()` vào chương trình để in các câu thông báo, giá trị của các biến trong chương trình,... và sử dụng công cụ trình gỡ lỗi (debugger) trong môi trường lập trình Python.

a) Thêm các lệnh print vào chương trình

Việc chèn các lệnh `print()` vào mã nguồn cho phép theo vết được quá trình thay đổi giá trị của các biến, từ đó có thể xác định được lỗi logic khi thực hiện chương trình. Cách làm này giúp dự đoán và khoanh vùng phần chương trình chứa các lệnh tạo ra kết quả sai.

Ví dụ 4: Quay lại chương trình ở KHỞI ĐỘNG, với trường hợp kiểm thử TC6: `A = [-2]`, để theo dõi giá trị của biến `gtLonNhat` bị thay đổi như thế nào thì thêm các lệnh `print()` vào trong chương trình như sau:

```
1 def timGTLonNhat(A):
2     gtLonNhat = 0
3     for i in range(len(A)):
4         # Tìm lỗi
5         print(f'gtLonNhat = {gtLonNhat}, A[{i}] = {A[i]}',end = ' -> ')
6         if gtLonNhat < A[i]:
7             gtLonNhat = A[i]
8             # Tìm lỗi
9             print(f'gtLonNhat = {gtLonNhat}')
10    return gtLonNhat
11 A = list(map(int,input("Nhập mảng A: ").split()))
12 print(f"Giá trị lớn nhất là {timGTLonNhat(A)}.")
```

Kết quả hiển thị là:

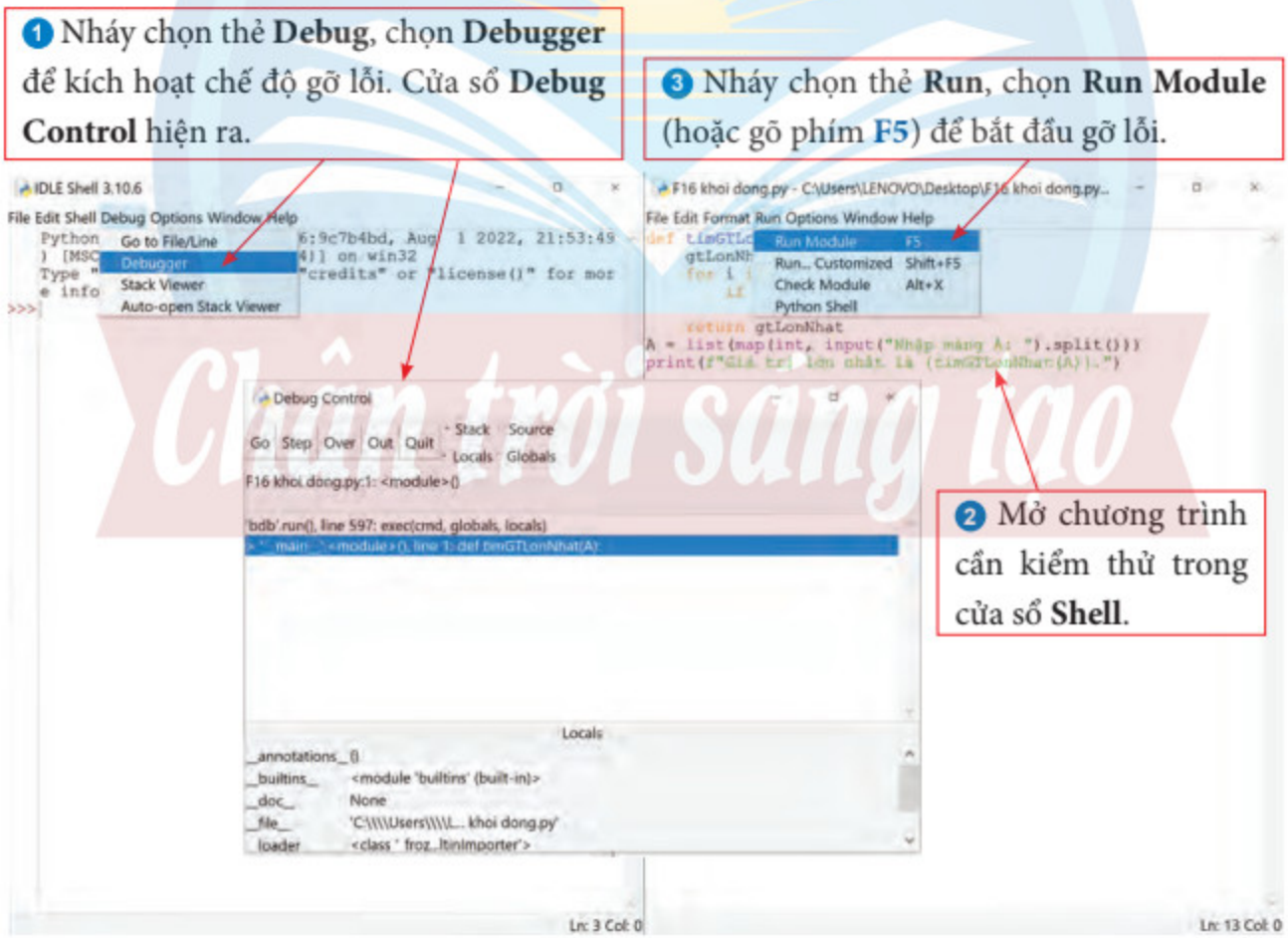
```
Nhập mảng A: -2
gtLonNhat = 0, A[0] = -2 -> Giá trị lớn nhất là 0.
```

Từ kết quả này có thể suy đoán lệnh gây ra lỗi ở dòng lệnh 2 và lệnh đúng là

```
gtLonNhat = A[0].
```

b) Sử dụng công cụ trình gỡ lỗi trong Python

Việc kiểm thử bằng cách chèn các lệnh print đòi hỏi người kiểm thử phải chỉnh sửa mã nguồn. Vì vậy, IDLE Shell cung cấp công cụ **Debugger** để kiểm thử chương trình. Để thực hiện kiểm thử và gỡ lỗi chương trình, thực hiện các thao tác sau:

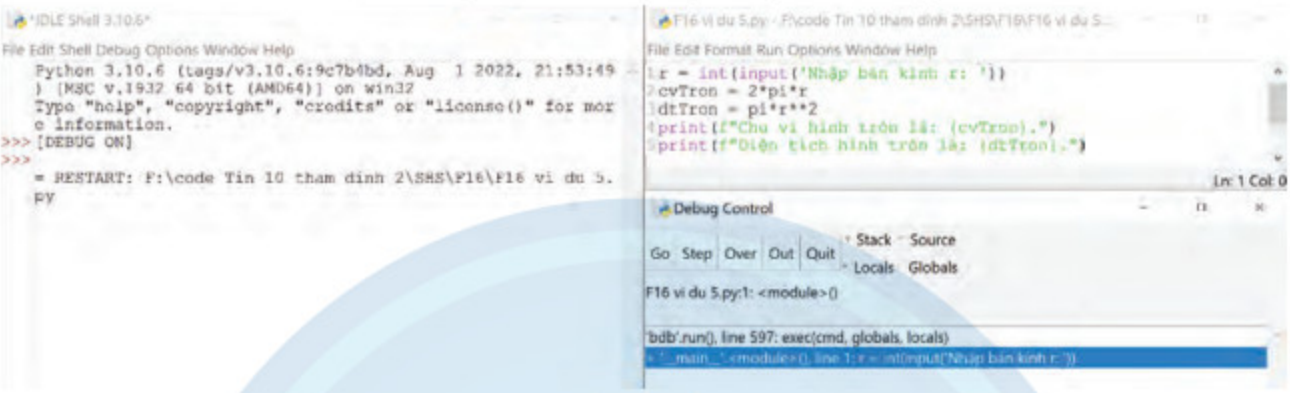


Hình 1. Minh hoạ các bước thực hiện gỡ lỗi chương trình

Ví dụ 5: Gỡ lỗi chương trình tính chu vi và diện tích hình chữ nhật, hình tròn lưu trong tệp dt.py như sau:

```
1 r = int(input('Nhập bán kính r: '))
2 cvTron = 2*pi*r
3 dtTron = pi*r**2
4 print(f"Chu vi hình tròn là: {cvTron}.")
5 print(f"Diện tích hình tròn là: {dtTron}.")
```

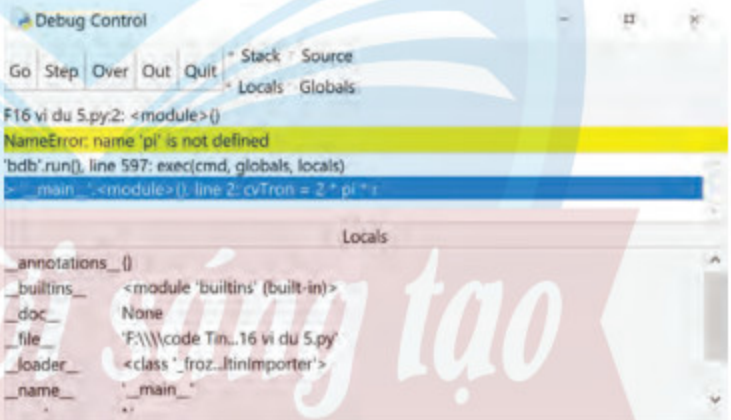
Ta có thể tìm kiếm lỗi chương trình trên bằng công cụ **Debugger** như *Hình 2*.



*Hình 2. Minh hoạ kiểm thử chương trình dt.py trong cửa sổ **Debug Control***

Trong cửa sổ **Debug Control**, nháy chọn **Step** để thực hiện chạy từng lệnh của chương trình và quan sát sự thay đổi của các biến. Đối với các lệnh vào-ra, thì **Step** bị ẩn để quay lại của sổ **Shell** chờ nhập giá trị. Khi thực hiện dòng lệnh 2 thì chương trình báo lỗi như *Hình 3* vì lỗi "name 'pi' is not defined".

Sau khi xác định được lỗi do dòng lệnh 2 gây ra, có thể nháy chọn **Out** để kết thúc gỡ lỗi chương trình hoặc nháy chọn **Over** để kết thúc hàm. Sau đó, để gỡ lỗi cần thêm dòng lệnh `from math import pi` vào dòng đầu tiên của Ví dụ 5.



Hình 3. Minh hoạ thông báo lỗi khi thực hiện kiểm thử

1. Có nên giữ các lệnh `print()` được chèn thêm sau khi hoàn thành kiểm thử chương trình hay không?

2. Em hãy cho biết ưu điểm khi sử dụng công cụ **Debugger** để kiểm thử và gỡ lỗi.

Chèn các lệnh `print()` vào chương trình để in các biến trung gian, hoặc sử dụng các chức năng **Step**, **Out**, **Over** của IDLE Shell để kiểm thử và gỡ lỗi chương trình.

LUYỆN TẬP

Cho chương trình xoá tất cả các phần tử có giá trị âm trong danh sách A như sau:

```
A = list(map(int,input().split())
i = 0
for i in A:
if i < 0:
    A.remove(i)
print(a)
```

Chương trình này đang gặp lỗi SyntaxError, IndentationError và NameError. Em hãy tìm và khắc phục các lỗi trên để chương trình có thể thực thi được.

THỰC HÀNH

Nhiệm vụ 1. Thực hành kiểm thử và gỡ lỗi chương trình

Yêu cầu: Tiếp tục làm việc với chương trình trong Bài tập của phần LUYỆN TẬP, thực hiện các yêu cầu dưới đây để gỡ lỗi chương trình, chạy ra kết quả đúng.

Sau khi sửa các lỗi SyntaxError, chạy thử chương trình với các bộ kiểm thử sau:

+ Bộ kiểm thử `TC1 = [-1,2,-3,4,-5]`, chương trình in ra màn hình kết quả `[2,4]`. Chương trình cho kết quả đúng.

+ Bộ kiểm thử `TC2 = [-1,-2,-3,-4,-5]`, chương trình in ra màn hình kết quả `[-2,-4]`. Chương trình cho kết quả sai.

a) Thêm lệnh `print()` vào các dòng thích hợp để tìm lỗi chương trình.

b) Gỡ lỗi và chỉnh sửa chương trình để thực hiện được yêu cầu đặt ra: xoá tất cả các phần tử có giá trị âm trong danh sách A.

Gợi ý: Tạo danh sách B là bản sao gán từng thành phần của danh sách A, duyệt từng phần tử trong danh sách B và sử dụng hàm `remove()` để xoá các phần tử này trong danh sách A.

Nhiệm vụ 2. Thực hành sử dụng công cụ Debugger

Yêu cầu: Cho chương trình ở *Hình 4*.

a) Sửa các lỗi SyntaxError, IndentationError và TypeError để thực thi được chương trình.

b) Sử dụng công cụ **Debugger** để kiểm thử và gỡ lỗi chương trình.

```
def tinhTong(a,b):
    tong = a+b
    print("Tổng là: " tong)

x = 10
y = "20"
ketQua = tinhTong(x,y)
  print(ketQua)
```

Hình 4. Chương trình tính tổng

VẬN DỤNG

1. Tìm hiểu và sử dụng chức năng đặt điểm ngắt (**Breakpoint**) trong **Debug Control** để bỏ qua các lệnh không cần kiểm tra và thực hiện nhanh đến dòng lệnh quan tâm.
2. Em hãy tìm hiểu và thay đổi giá trị của các biến trong vùng cửa sổ **Locals** hay **Globals** của **Debug Control** trong quá trình kiểm thử.

BÀI F17 THỰC HÀNH TỔNG HỢP

MỤC TIÊU **Sau bài học này, em sẽ:**

- Tạo được các hàm tự định nghĩa và thiết kế được thuật toán có sử dụng các cấu trúc điều khiển của Python.
- Sử dụng được các hàm thư viện của Python để viết chương trình giải quyết một số bài toán đơn giản.

KHỞI ĐỘNG

Quay lại KHỞI ĐỘNG ở Bài F11, tới kì thi tuyển học sinh giỏi, cô giáo cần tìm năm em học sinh có điểm cao nhất trong bảng điểm môn Tin học của lớp 10A1 để chọn vào đội tuyển học sinh giỏi nên đã yêu cầu bạn An sửa lại chương trình đã viết để giải quyết yêu cầu này của cô giáo. Em hãy tìm hiểu và hướng dẫn An giải quyết vấn đề trên.

THỰC HÀNH

Nhiệm vụ 1. Liệt kê các số đối xứng đầu tiên

Số đối xứng là số nguyên dương mà khi đảo ngược các chữ số của số nguyên này thì giá trị vẫn không thay đổi. Ví dụ: Các số 1, 33, 121, 12 321 là các số đối xứng; các số 12, 123, 12 312 là các số không đối xứng.

Yêu cầu: Viết chương trình nhập số nguyên dương n và hiển thị n số nguyên đầu tiên là số đối xứng.

Hướng dẫn: Kiểm tra số nguyên `i` là số đối xứng bằng cách dùng hàm `str(i)` để chuyển đổi số nguyên `i` thành xâu `s`. Sau đó, nếu xâu `s` bằng xâu đảo ngược của xâu `s` thì số nguyên `i` là số đối xứng; ngược lại, số nguyên `i` không là số đối xứng. Cụ thể như sau:

1. Định nghĩa hàm `soDoiXung(i)`: # Hàm kiểm tra số nguyên i là số đối xứng
   Chuyển đổi số nguyên i thành xâu s
   Trả về (xâu s bằng xâu đảo ngược của s)
2. Định nghĩa hàm `inSoDoiXung(n)`: # Hàm hiển thị n số đối xứng đầu tiên
   Hiển thị "các số đối xứng đầu tiên"
   `soDem = 0` # Đếm số lượng số đối xứng
   `i = 1` # Bắt đầu từ số 1

Vòng lặp trong khi `soDem < n`:

Nếu `soDoiXung(i)` thì: # i là số đối xứng

Tăng `soDem` lên 1

Hiển thị số đối xứng i

Tăng i lên 1

3 Chương trình chính

Nhập số nguyên dương n

`inSoDoiXung(n)` # Hiển thị n số đối xứng đầu tiên

Chương trình tham khảo:

```
1 # Hàm kiểm tra số nguyên i là số đối xứng
2 def soDoiXung(i):
3     s = str(i)
4     return s == s[::-1]
5 # Hàm hiển thị n số đối xứng đầu tiên
6 def inSoDoiXung(n):
7     print(f"{n} số đối xứng đầu tiên:")
8     soDem = 0
9     i = 1
10    while soDem < n:
11        if soDoiXung(i):
12            soDem = soDem+1
13            print(i,end = ' ')
14        i = i+1
15 # Chương trình chính
16 n = int(input("Nhập số dương n: "))
17 inSoDoiXung(n)
```

Nhiệm vụ 2. Sắp xếp bảng điểm của môn học

Yêu cầu: Viết chương trình lặp lại nhiều lần công việc nhập dữ liệu của học sinh: nhập mã số, nếu mã số nhỏ hơn hoặc bằng 0 thì kết thúc việc nhập; ngược lại, tiếp tục nhập họ tên, nhập điểm số và chèn dữ liệu của học sinh này vào cuối bảng điểm (ban đầu, bảng điểm là rỗng). Nếu trùng mã số thì phải nhập lại mã số khác. Cuối cùng, hiển thị bảng điểm của môn học ra màn hình theo thứ tự điểm số tăng dần, mỗi dòng gồm mã số, họ tên và điểm số của một học sinh. Tham khảo Nhiệm vụ 2 của Bài F12.

Hướng dẫn: Bảng điểm `bangDiem` có 3 cột là mã số (cột 0), họ tên (cột 1), điểm số (cột 2). Cột mã số là một danh sách `[row[0] for row in bangDiem]`. Mã số `maSo` bị trùng khi `maSo in [row[0] for row in bangDiem]`. Sắp xếp bảng điểm `bangDiem` theo thứ tự điểm số (cột 2) tăng dần bằng lệnh `bangDiem.sort(key = lambda bgDiem: bgDiem[2])`, có nghĩa là với mỗi hàng `bgDiem` thì lấy giá trị của `bdDiem[2]` (là điểm số ở cột 2) để sắp xếp. Cụ thể như sau:

1 Định nghĩa hàm `nhapBangDiem()`:# Nhập bảng điểm của môn học

Tạo bảng điểm `bangDiem` rỗng

`tiepTuc = True` # Còn tiếp tục nhập

Vòng lặp trong khi `tiepTuc`:

Nhập mã số (kết thúc = 0) và gán cho biến `maSo`

Nếu mã số <= 0 thì:

`tiepTuc = False` # Kết thúc nhập

Ngược lại nếu mã số đã có trong bảng điểm: # Trùng mã số

Hiển thị trùng mã số

Ngược lại: # Mã số > 0

Nhập họ tên và gán cho biến `hoTen`

Nhập điểm số và gán cho biến `diemSo`

Chèn học sinh `[maSo, hoTen, diemSo]` cuối bảng điểm `bangDiem`

Trả về bảng điểm `bangDiem`

2 Định nghĩa hàm `inBangDiem(bangDiem)`: # In bảng điểm của môn học

Hiển thị "Bảng điểm theo thứ tự điểm số tăng dần:"

Vòng lặp i từ 0 đến (số học sinh - 1):

Lấy mã số, họ tên, điểm số của học sinh là `bangDiem[i]`

Hiển thị mã số, họ tên, điểm số

3 Chương trình chính

`bangDiem = nhapBangDiem()` # Mã số (cột 0), họ tên (cột 1), điểm số (cột 2)

Sắp xếp bảng điểm `bangDiem` dựa vào cột thứ 2 (điểm số) tăng dần

`inBangDiem(bangDiem)` # In bảng điểm của môn học

Chương trình tham khảo:

```
1 # Nhập bảng điểm của môn học
2 def nhapBangDiem():
3     bangDiem = []
4     tiepTuc = True
5     while tiepTuc:
6         maSo = int(input("Nhập mã số (kết thúc = 0): "))
7         if maSo <= 0:
8             tiepTuc = False
9         elif maSo in [row[0] for row in bangDiem]:
10            print(f"Trùng mã số {maSo}.")
11        else:
12            hoTen = input("Nhập họ tên: ")
13            diemSo = input("Nhập điểm số: ")
14            bangDiem.append([maSo,hoTen,diemSo])
15    return bangDiem
16 # In bảng điểm của môn học
17 def inBangDiem(bangDiem):
18    print("Bảng điểm theo thứ tự điểm số tăng dần:")
19    for i in range(len(bangDiem)):
20        maSo,hoTen,diemSo = bangDiem[i]
21        print(f"{i+1}. {maSo} - {hoTen}: {diemSo}")
22 # Chương trình chính
23 bangDiem = nhapBangDiem()
24 bangDiem.sort(key = lambda bgDiem: bgDiem[2])
25 inBangDiem(bangDiem)
```

Nhiệm vụ 3. Tìm k giá trị lớn nhất

Yêu cầu: Viết chương trình nhập danh sách A gồm các số nguyên và số nguyên dương k. In ra màn hình danh sách A và k giá trị lớn nhất trong danh sách A.

Hướng dẫn: Tạo danh sách B bằng cách gán từng phần tử khác nhau của danh sách A cho danh sách B. Sắp xếp danh sách B theo thứ tự giảm dần. Sau đó, tạo danh sách C chứa k phần tử đầu tiên của danh sách B, là k giá trị lớn nhất trong danh sách A. Cụ thể như sau:

1. Định nghĩa hàm `cacGiaTriLonNhat(A,k)`: # Hàm tìm k giá trị lớn nhất của danh sách A

   Tạo danh sách B rỗng

   Với mỗi phần tử x nằm trong danh sách A:

   # Tạo danh sách B gồm các phần tử khác nhau của danh sách A

   Nếu x không nằm trong danh sách B thì:

   Chèn x cuối danh sách B

   Sắp xếp B theo thứ tự giảm dần

   Tạo danh sách C rỗng

   So sánh k với `len(B)` và gán `k = len(B)` nếu `k > len(B)`

   Lần lượt chèn k phần tử đầu của B vào cuối danh sách C

   Trả về danh sách C

2. Chương trình chính

   Nhập danh sách A

   Nhập số nguyên dương k

   `C = cacGiaTriLonNhat(A)` # Tạo danh sách k giá trị

   In ra màn hình danh sách A và danh sách C

Chương trình tham khảo:

```
def cacGiaTriLonNhat(A,k):
    B = []
    for x in A:
        if x not in B:          # B chứa các giá trị khác nhau của A
            B.append(x)
    B.sort(reverse = True)      # Sắp xếp lại B
    C = []                      # C chứa k phần tử đầu tiên của B
    if k > len(B):
        k = len(B)
    for i in range(k):
        C.append(B[i])
    return C

# Chương trình chính
A = list(map(int,input("Nhập danh sách A: ").split()))
k = int(input("Nhập số nguyên k > 0: "))
C = cacGiaTriLonNhat(A,k)    # Tạo danh sách k giá trị
print("Danh sách A các số nguyên: ",A)
print(f"{len(C)} giá trị lớn nhất của A:",end=" ")
for x in C:
    print(x,end=" ")
```

LUYỆN TẬP

Tiếp tục chương trình nhập của Nhiệm vụ 2, sử dụng định nghĩa hàm `cacGiaTriLonNhat(C,k)` ở Nhiệm vụ 3, viết thêm các lệnh để hiển thị danh sách các học sinh có điểm số nằm trong k điểm số lớn nhất nhằm giải quyết vấn đề ở KHỞI ĐỘNG.

Hướng dẫn: Duyệt bảng điểm để tạo danh sách `C` là cột điểm số (cột 2) của bảng điểm. Sau đó, tạo danh sách `B` là danh sách có `k` điểm số lớn nhất bằng cách gọi hàm `cacGiaTriLonNhat(C,k)` ở Nhiệm vụ 3. Cuối cùng, duyệt bảng điểm để hiển thị các học sinh có điểm số nằm trong danh sách `B` (là `k` điểm số lớn nhất).

VẬN DỤNG

Viết chương trình phân tích một số tự nhiên n > 1 thành tích các thừa số nguyên tố.

Dữ liệu vào: Số tự nhiên n > 1.

Dữ liệu ra: Danh sách các thừa số nguyên tố có tích số bằng n.

Thông tin nhập	Kết quả hiển thị
`Số tự nhiên n > 1: 2024`	`Danh sách các thừa số nguyên tố có tích bằng 2024: 2 2 2 11 23`

Hướng dẫn: Sử dụng danh sách `A` chứa các thừa số nguyên tố; ban đầu, danh sách `A` là rỗng. Phân tích số tự nhiên `n > 1` thành các thừa số nguyên tố bằng cách chia `n` cho các số tự nhiên `i` bắt đầu từ `2`. Nếu `n` chia hết cho `i` thì chèn `i` vào danh sách `A`, rồi gán `n` bằng phần nguyên của phép chia `n` cho `i`. Nếu `n` không chia hết cho `i` thì tăng `i` lên `1`. Quá trình chia `n` cho `i` sẽ kết thúc khi `n = 1`. Cuối cùng, in ra màn hình danh sách `A`. Cụ thể như sau:

1. Định nghĩa hàm `tichSoNguyenTo(n)`: # Hàm liệt kê các thừa số nguyên tố có tích số bằng n.

```
Tạo danh sách A rỗng # Danh sách chứa các thừa số nguyên tố cần tìm
i = 2 # Bắt đầu từ số 2
Vòng lặp trong khi n > 1:
    Nếu n chia hết cho i thì:
        Chèn i vào cuối danh sách A
        n = lấy phần nguyên của n chia i
    Ngược lại: # n không chia hết cho i
        Tăng i lên 1
Trả về danh sách A
```

2. Chương trình chính

```
Nhập số nguyên n > 1
A = tichSoNguyenTo(n)  # Danh sách A chứa các thừa số nguyên tố
In thông báo "Danh sách các thừa số nguyên tố có tích bằng n: "
In các phần tử trong danh sách A, phân cách nhau bởi khoảng trắng.
```

Chân trời sáng tạo

Chủ đề G

HƯỚNG NGHIỆP VỚI TIN HỌC

BÀI G1 NHÓM NGHỀ THIẾT KẾ ĐỒ HOẠ

MỤC TIÊU **Sau bài học này, em sẽ:**

- Trình bày được thông tin hướng nghiệp về nhóm nghề thiết kế đồ hoạ:
  + Những nét sơ lược về công việc chính mà người làm nghề phải thực hiện.
  + Yêu cầu thiết yếu về kiến thức và kĩ năng cần có để làm nghề.
  + Ngành học có liên quan ở các bậc học tiếp theo.
  + Nhu cầu nhân lực hiện tại và tương lai của nhóm nghề đó.
- Tự tìm kiếm và khai thác được thông tin hướng nghiệp.
- Giao lưu được với bạn bè qua các kênh truyền thông tin số để tham khảo và trao đổi ý kiến về những thông tin trên.

KHỞI ĐỘNG

Thảo luận với bạn và cho biết công việc nào có thể tạo ra các sản phẩm như *Hình 1*.

Hình 1a. Thiết kế logo

Hình 1b. Bìa sách

Hình 1c. Poster quảng cáo

Hình 1. Các sản phẩm

KHÁM PHÁ

1. Tìm hiểu về nhóm nghề thiết kế đồ hoạ

Nhóm nghề thiết kế đồ hoạ được quy định ở mục 2166 (Nhà thiết kế đồ hoạ và truyền thông đa phương tiện) của bảng danh mục nghề nghiệp Việt Nam theo Quyết định số 34/2020/QĐ-TTg của Thủ tướng Chính phủ ngày 26/11/2020.

Trong thời đại ngày nay, khi thông tin và dữ liệu trở thành một phần không thể thiếu trong cuộc sống, cách thức truyền tải thông tin ngày càng trở nên quan trọng. Con người không chỉ tiếp nhận thông tin bằng lời nói hay chữ viết mà còn qua những hình ảnh sinh động. Hình ảnh có khả năng truyền tải ý nghĩa nhanh chóng và hiệu quả hơn, tạo ra những kết nối cảm xúc mạnh mẽ với người xem. Chính vì vậy, đồ hoạ đã trở thành sự kết hợp hoàn hảo giữa nghệ thuật và thông tin, là một lĩnh vực truyền thông giúp trình bày nội dung một cách trực quan và sáng tạo, làm nổi bật thông điệp trong thời đại số hoá. Thiết kế đồ hoạ là quá trình tạo ra các giải pháp bằng hình ảnh để truyền thông. Nhóm nghề thiết kế đồ hoạ là một công việc ứng dụng kết hợp hình ảnh, chữ viết và ý tưởng một cách sáng tạo để truyền đạt thông tin hiệu quả dưới hình thức các sản phẩm in ấn hoặc kĩ thuật số.

Thiết kế đồ hoạ có thể ứng dụng vào rất nhiều công việc trong cuộc sống như: thiết kế quảng cáo, thiết kế đồ hoạ trang web, thiết kế đồ hoạ in ấn,... Thiết kế đồ hoạ được áp dụng trong nhiều lĩnh vực khác nhau nhằm truyền đạt hiệu quả truyền thông cao nhất, giúp trực quan hoá dữ liệu và thông tin, phục vụ cho việc quảng cáo, kinh doanh hoặc tuyên truyền các hoạt động.

Quá trình thiết kế đồ hoạ thường đòi hỏi sự sáng tạo, kiến thức về mĩ thuật và thiết kế cũng như hiểu biết về công nghệ và công cụ sử dụng để tạo ra những sản phẩm đồ hoạ. Sản phẩm có thể bắt gặp ở mọi nơi trong cuộc sống như: tấm banner quảng cáo, poster phim, bìa sách,... giúp doanh nghiệp, tổ chức hoặc cá nhân truyền đạt thông tin hiệu quả.

Một số công việc thuộc nhóm nghề thiết kế đồ hoạ có thể kể đến như:

Nhà thiết kế đồ hoạ: Liên quan đến việc tạo ra các hình ảnh trực quan như logo, banner, bao bì sản phẩm,...

Nhà chuyên môn về hoạt hình: Liên quan đến việc sáng tạo ra các hình ảnh chuyển động để kể những câu chuyện cho những mục đích khác nhau như phim hoạt hình, video quảng cáo,...

Hoạ sĩ vẽ minh hoạ: Liên quan đến việc tạo ra các hình ảnh nhằm mục đích minh hoạ cho sách, truyện tranh, tạp chí và nhiều sản phẩm khác.

Nhà thiết kế đồ hoạ cho trò chơi máy tính: Liên quan đến việc tạo ra các hình ảnh và hiệu ứng sống động nhằm mang đến trải nghiệm tốt cho các trò chơi máy tính.

Nghệ sĩ kĩ thuật số: Liên quan đến việc sử dụng máy tính để tạo ra các tác phẩm nghệ thuật sáng tạo và sống động.

Nhà thiết kế xuất bản: Liên quan đến việc tạo ra bố cục và trang trí cho các ấn phẩm như sách, tạp chí và các tài liệu in ấn khác.

Nhà thiết kế trang web: Liên quan đến việc tạo ra các giao diện trực quan và dễ sử dụng cho các trang web.

Theo em, tại sao hiện nay, các bạn trẻ yêu thích công việc thiết kế đồ hoạ?

- Thiết kế đồ hoạ là quá trình tạo ra các giải pháp bằng hình ảnh để truyền thông. Nhóm nghề thiết kế đồ hoạ là công việc ứng dụng kết hợp hình ảnh, chữ viết và ý tưởng một cách sáng tạo để truyền đạt thông tin hiệu quả dưới hình thức các sản phẩm in ấn hoặc kĩ thuật số.
- Một số công việc thuộc nhóm nghề thiết kế đồ hoạ có thể kể đến như: Nhà thiết kế đồ hoạ; Nhà chuyên môn về hoạt hình; Hoạ sĩ vẽ minh hoạ; Nhà thiết kế đồ hoạ cho trò chơi máy tính; Nghệ sĩ kĩ thuật số; Nhà thiết kế xuất bản; Nhà thiết kế trang web,...

2. Một vài yêu cầu cơ bản của nhóm nghề thiết kế đồ hoạ

Vì là nhóm nghề kết hợp giữa nghệ thuật và thông tin, nên ngoài kĩ năng vẽ, người làm việc trong nhóm nghề thiết kế đồ hoạ cần đáp ứng được các yêu cầu sau:

Luôn học hỏi kiến thức chung: Người làm đồ hoạ cần luôn tìm hiểu những kiến thức mới về các lĩnh vực (toán học, vật lí, nghệ thuật,...). Đồng thời, họ cũng cần nắm bắt các thông tin về xu hướng, nhu cầu của xã hội để có thể ứng dụng trong công việc của mình.

Nắm vững chuyên môn về mĩ thuật: Người làm đồ hoạ cần nắm vững chuyên môn về nguyên tắc mĩ thuật và thiết kế; cách sử dụng các yếu tố thiết kế như màu sắc, hình ảnh, phông chữ và kiểu đồ hoạ để cân bằng và hài hoà trong thiết kế.

Kĩ năng thiết kế và sử dụng thành thạo các phần mềm vẽ đồ hoạ: Người làm đồ hoạ cần có kĩ năng thiết kế và sử dụng thành thạo các phần mềm đồ hoạ. Việc thành thạo các công cụ như Adobe Illustrator, Adobe Photoshop, Adobe InDesign, Inkscape, GIMP,... giúp các nhà thiết kế tạo ra các sản phẩm đồ hoạ chất lượng cao. Sự thành thạo này không chỉ nâng cao khả năng sáng tạo mà còn đảm bảo hiệu quả trong việc truyền tải thông điệp, cải thiện giao diện người dùng, tạo ra những hình ảnh ấn tượng, đáp ứng yêu cầu của khách hàng và các dự án thiết kế.

Ngoài các kĩ năng cứng, người làm trong nhóm nghề thiết kế đồ hoạ còn cần có khả năng sáng tạo, tư duy thẩm mĩ, tinh thần đam mê, trách nhiệm với mỗi bản thiết kế của cá nhân.

Ngoài kĩ năng vẽ, người làm trong nhóm nghề thiết kế đồ hoạ cần đáp ứng được các yêu cầu sau: luôn học hỏi kiến thức chung, nắm vững chuyên môn về mĩ thuật, kĩ năng thiết kế và sử dụng thành thạo các phần mềm đồ hoạ.

3. Học tập, giao lưu và nhu cầu nhân lực trong nhóm nghề thiết kế đồ hoạ

a) Nhu cầu nhân lực

Khi xã hội ngày càng phát triển, nhu cầu về thẩm mĩ của con người ngày càng cao. Thế nên, các sản phẩm ngoài độ bền còn cần đến yêu cầu về mặt thẩm mĩ, tạo cơ hội cho nhóm nghề thiết kế đồ hoạ phát triển. Theo báo cáo trong Sách Trắng Công nghệ thông tin và Truyền Thông Việt Nam năm 2023, tổng số xuất bản phẩm dạng điện tử năm 2022 khoảng 3 350 xuất bản phẩm, tăng 0.67% so với năm 2021. Điều này cho thấy nhu cầu thiết kế đồ hoạ đang ngày càng tăng đáng kể trong bối cảnh xã hội phát triển và công nghệ thông tin ngày càng phổ biến. Với những cơ hội nghề nghiệp đa dạng, thu nhập hấp dẫn, nhóm nghề thiết kế đồ hoạ đang thu hút sự quan tâm của người có đam mê về sáng tạo, thẩm mĩ. Điều này cũng làm nổi bật sự quan trọng của việc đào tạo, phát triển nhân lực có chất lượng cao để đáp ứng nhu cầu ngà càng tăng trong thời đại số hoá ngày nay.

b) Thông tin hướng nghiệp

Ngoài một số công việc được nhắc đến ở mục 1, người làm trong nhóm nghề thiết kế đồ hoạ còn có thể tự thành lập doanh nghiệp, các công ty về thiết kế, hoặc chuyên gia tư vấn về thiết kế,… Đồng thời, còn có khả năng tiếp tục theo học các chương trình đào tạo chuyên sâu (thạc sĩ, tiến sĩ,…) thuộc nhóm ngành mĩ thuật, thiết kế; thực hiện các nghiên cứu chuyên sâu và trở thành cán bộ giảng dạy, cán bộ nghiên cứu tại các cơ sở giáo dục đại học, cao đẳng và viện nghiên cứu liên quan. Ngoài ra, với đặc thù nhóm nghề, người làm trong nhóm nghề thiết kế đồ hoạ còn có khả năng làm việc độc lập hoặc theo nhóm tại nhà trong điều kiện thay đổi (freelancer).

c) Các kênh giao lưu và học tập

Để bắt đầu với nhóm nghề thiết kế đồ hoạ, em có thể tham gia, trao đổi, thảo luận trên các diễn đàn trực tuyến liên quan đến nhóm nghề thiết kế đồ hoạ; các hội nhóm chuyên ngành trên các nền tảng xã hội như Facebook, Telegram,... Các em có thể tìm hiểu thông tin về ngành này trên trang web các trường đại học, cao đẳng về mĩ thuật, kiến trúc (Đại học Kiến trúc TPHCM, Đại học Mĩ thuật TPHCM,…) hay các cơ sở đào tạo có chuyên ngành thiết kế đồ hoạ. Ngoài ra, các em cũng có thể tìm kiếm, trao đổi thông tin về hướng nghiệp, việc làm trên Internet thông qua:

Cổng thông tin điện tử của Bộ Lao động – Thương binh và xã hội (https://molisa.gov.vn/): Cung cấp thông tin về các chính sách việc làm, hướng nghiệp và đào tạo nghề.

Trang web của Tổng cục giáo dục nghề nghiệp (https://mhctructuyen.gdnn.gov.vn/): Cung cấp thông tin về các khoá đào tạo nghề, chương trình học, cơ hội nghề nghiệp trong ngành thiết kế đồ hoạ.

Cổng thông tin việc làm: Cổng thông tin việc làm tại các tỉnh thành cung cấp thông tin về các cơ hội việc làm, tuyển dụng và hướng nghiệp trong nhiều lĩnh vực, bao gồm thiết kế đồ hoạ.

Diễn đàn và mạng xã hội: Tham gia vào diễn đàn và mạng xã hội để trao đổi và tìm hiểu các thông tin từ những người đi trước để có thêm thông tin hữu ích về định hướng nghề nghiệp của bản thân.

Hội thảo và sự kiện: Tham gia các hội thảo, sự kiện, triển lãm và các buổi đàm thoại liên quan đến nhóm nghề thiết kế đồ hoạ để gặp gỡ, giao lưu, học hỏi từ các chuyên gia trong lĩnh vực này.

Em hãy kể tên một số lĩnh vực mà người học có thể làm liên quan đến nhóm nghề thiết kế đồ hoạ.

- Em có thể tìm hiểu về nhóm nghề thiết kế đồ hoạ thông qua các trang web của các trường đại học, cơ sở có đào tạo uy tín. Đồng thời có thể tìm kiếm và trao đổi thông tin về hướng nghiệp, việc làm thông qua các diễn đàn và mạng xã hội; các hội thảo và sự kiện liên quan đến chuyên ngành thiết kế đồ hoạ trên cả nước.
- Nhu cầu nhân lực trong ngành thiết kế đồ hoạ ngày càng tăng cao và nhiều công việc cũng như cách thức làm việc rất đa dạng.

LUYỆN TẬP

1. Em hãy cho biết người làm nghề thiết kế đồ hoạ có thể làm việc trong những cơ sở, tổ chức nào?
2. Kể tên một số phần mềm cần thiết cho nhóm nghề thiết kế đồ hoạ.

VẬN DỤNG

1. Tìm hiểu một số trang web về tuyển dụng nhân lực thuộc nhóm nghề thiết kế đồ hoạ và trình bày cho các bạn trong lớp.
2. Em hãy thảo luận với bạn và tìm hiểu chương trình đào tạo ngành thiết kế đồ hoạ tại các trường đại học, cao đẳng có đào tạo ngành nghề đó. Sau đó, liệt kê các thông tin cơ bản về kiến thức và kĩ năng của người học sau khi hoàn thành chương trình học.

Chân trời sáng tạo

BÀI G2 NHÓM NGHỀ LẬP TRÌNH

MỤC TIÊU **Sau bài học này, em sẽ:**

- Trình bày được thông tin hướng nghiệp về nhóm nghề lập trình:
  + Những nét sơ lược về công việc chính mà người làm nghề phải thực hiện.
  + Yêu cầu thiết yếu về kiến thức và kĩ năng cần có để làm nghề.
  + Ngành học có liên quan ở các bậc học tiếp theo.
  + Nhu cầu nhân lực hiện tại và tương lai của nhóm nghề đó.
- Tự tìm kiếm và khai thác được thông tin hướng nghiệp về một vài ngành nghề khác trong lĩnh vực tin học.
- Giao lưu được với bạn bè qua các kênh truyền thông để tham khảo và trao đổi ý kiến về những thông tin trên.

KHỞI ĐỘNG

Với sự bùng nổ của công nghệ thông tin, các phần mềm ứng dụng ngày nay xuất hiện trong hầu hết các lĩnh vực của cuộc sống. Việc xây dựng các phần mềm ứng dụng này cần một lượng lớn nhân sự lập trình viên trình độ cao. Em hãy tìm hiểu và giới thiệu với các bạn những thông tin về nghề lập trình, những nội dung kiến thức cần đạt được để có thể trở thành một lập trình viên và mức thu nhập hiện nay của lập trình viên tại Việt Nam cũng như trên thế giới.

KHÁM PHÁ

Chân trời sáng tạo

1. Tìm hiểu về nhóm nghề lập trình

Nhóm nghề lập trình bao gồm nhiều lĩnh vực khác nhau, mỗi lĩnh vực đều tập trung vào một khía cạnh cụ thể của công nghệ thông tin và phát triển phần mềm. Ngoài nhiệm vụ viết mã nguồn, tạo ra các phần mềm ứng dụng hoạt động hiệu quả và an toàn, công việc của nhân sự làm nghề lập trình (lập trình viên) còn bao gồm kiểm tra, sửa lỗi và nâng cấp phần mềm, nghiên cứu và phát triển công nghệ mới,...

Được quy định ở mục 251 (Nhà chuyên môn về phân tích và phát triển phần mềm và các ứng dụng) của Bảng danh mục nghề nghiệp Việt Nam, theo Quyết định số 34/2020/QĐ-TTg của Thủ tướng Chính phủ, nhóm nghề lập trình có một số nghề điển hình như:

Lập trình viên phát triển phần mềm (2512): nghiên cứu, phân tích và đánh giá những yêu cầu cho các ứng dụng, hệ điều hành, phần mềm hiện có hoặc tạo mới, đồng thời thiết kế, phát triển, thử nghiệm và duy trì các giải pháp phần mềm để đáp ứng các yêu cầu này.

Lập trình viên thiết kế trò chơi điện tử (2513): thuộc nhóm các nhà phát triển web và truyền thông đa phương tiện, kết hợp kiến thức thiết kế và kĩ thuật lập trình để nghiên cứu, phân tích, đánh giá, thiết kế, lập trình và sửa đổi các trang web, những ứng dụng kết hợp văn bản, đồ hoạ, hoạt hình, hình ảnh, âm thanh, video và các phương tiện tương tác khác.

Chuyên viên phân tích thiết kế hệ thống (2511): nghiên cứu, phân tích và đánh giá các yêu cầu, quy trình hoặc vấn đề công nghệ thông tin của khách hàng, đồng thời phát triển và thực hiện các đề xuất, khuyến nghị và kế hoạch cải thiện hệ thống thông tin hiện tại hoặc trong tương lai.

Như vậy, dù làm nghề nào trong nhóm nghề lập trình, lập trình viên đều cần thực hiện các nhiệm vụ sau:

Viết mã nguồn: sử dụng ngôn ngữ lập trình Python, Java, C++,… để tạo ra chương trình và ứng dụng.

Kiểm tra và sửa lỗi: Đảm bảo phần mềm hoạt động đúng như thiết kế bằng cách kiểm tra và sửa lỗi (debugging).

Nâng cấp và bảo trì: Cập nhật và cải tiến phần mềm để đáp ứng các yêu cầu mới hoặc khắc phục những vấn đề phát sinh.

Phân tích yêu cầu: Làm việc với khách hàng để hiểu rõ yêu cầu và xây dựng giải pháp phù hợp.

Phối hợp làm việc nhóm: Phối hợp cùng nhà thiết kế, người quản lí dự án và chuyên gia ở những lĩnh vực khác để đảm bảo sản phẩm cuối cùng đáp ứng yêu cầu.

Lập trình viên cần liên tục cập nhật kiến thức để theo kịp các xu hướng công nghệ mới, luôn luôn sáng tạo, kiên nhẫn và có khả năng giải quyết vấn đề tốt. Với sự phát triển không ngừng của công nghệ thông tin, đóng vai trò quan trọng nhóm nghề lập trình ngày càng trở nên quan trọng, có mức thu nhập đáng mơ ước và có nhiều cơ hội phát triển trong tương lai (*Hình 1*).

Hình 1. Lập trình viên đang lập trình phần mềm

1. Trong phát triển phần mềm, khái niệm "debugging" đề cập đến nhiệm vụ gì?

A. Viết mã nguồn. B. Kiểm tra và sửa lỗi.
C. Thiết kế giao diện. D. Quản lí dự án.

2. Để trở thành chuyên viên phân tích thiết kế hệ thống, kĩ năng nào sau đây là quan trọng nhất?

A. Thiết kế đồ hoạ. B. Lập trình C và C++.
C. Lắp ráp máy tính. D. Phát triển web.

3. Một lập trình viên phát triển phần mềm sẽ có khả năng được tuyển dụng cao nhất bởi công ty nào?

A. Công ty phát hành game trực tuyến. B. Công ty sản xuất thiết bị điện tử.
C. Công ty thiết kế đồ hoạ. D. Công ty tư vấn tài chính.

- Nghề lập trình là công việc của lập trình viên, sử dụng ngôn ngữ lập trình để thiết kế, xây dựng và bảo trì các phần mềm ứng dụng.
- Một số nghề điển hình trong nhóm nghề lập trình: Lập trình viên phát triển phần mềm, Lập trình viên phát triển trò chơi điện tử, Chuyên viên phân tích thiết kế hệ thống.

2. Những yêu cầu của lập trình viên

Lập trình viên, còn được gọi là kĩ sư phần mềm, có thể làm việc trong nhiều lĩnh vực khác nhau như phát triển phần mềm, phát triển web, phát triển ứng dụng di động, lập trình game, phân tích thiết kế hệ thống,… Do đó, để có thể trở thành một lập trình viên, em cần trang bị các kiến thức và kĩ năng sau:

Về kiến thức:

– Ngôn ngữ lập trình: thành thạo ít nhất 1 ngôn ngữ lập trình như Python, Java, C++. Mỗi ngôn ngữ có ưu điểm riêng và phù hợp với các loại dự án khác nhau.

– Cấu trúc dữ liệu và thuật toán: hiểu biết sâu về các cấu trúc dữ liệu (mảng, danh sách,…) và các thuật toán (sắp xếp, tìm kiếm, đệ quy) để giải quyết vấn đề một cách hiệu quả.

– Cơ sở dữ liệu: hiểu biết về các hệ quản trị cơ sở dữ liệu để thiết kế và quản lí dữ liệu hiệu quả, giải quyết được các bài toán thực tế.

– Hệ điều hành và mạng máy tính: kiến thức về hệ điều hành (Windows, Linux) và mạng máy tính giúp lập trình viên hiểu rõ cách phần mềm tương tác với phần cứng và mạng.

– Bảo mật thông tin: hiểu biết về các nguyên tắc bảo mật để bảo vệ dữ liệu và ứng dụng, khỏi các mối đe doạ trong môi trường chuyên nghiệp.

Về kĩ năng:

– Kĩ năng lập trình: đây là nền tảng quan trọng nhất của một lập trình viên.

– Kĩ năng phân tích và giải quyết vấn đề: khả năng phân tích các vấn đề phức tạp, xác định được nguyên nhân và đưa ra giải pháp tối ưu.

– Kĩ năng làm việc nhóm: giao tiếp rõ ràng, lắng nghe ý kiến và cùng đóng góp ý tưởng để xây dựng được phần mềm đáp ứng yêu cầu, hoạt động hiệu quả và an toàn.

– Tư duy mở và sáng tạo: sẵn sàng chấp nhận và thử nghiệm các ý tưởng mới, tìm ra các giải pháp đột phá cho các vấn đề kĩ thuật.

– Khả năng tự học và cập nhật kiến thức: vì công nghệ liên tục thay đổi và cập nhật, lập trình viên cũng cần được trang bị khả năng tự học, tự cập nhật kiến thức mới.

Để sẵn sàng cho việc trở thành một lập trình viên, em cần có sự chuẩn bị đầy đủ và nghiêm túc bằng việc chủ động tìm hiểu và tham gia những khoá học chuyên sâu về ngôn ngữ lập trình, cấu trúc dữ liệu và thuật toán, cơ sở dữ liệu, hệ điều hành và mạng máy tính, phát triển ứng dụng di động, bảo mật thông tin,…

Các cơ sở giáo dục tại Việt Nam có đào tạo các nghề thuộc nhóm nghề lập trình gồm có Đại học Bách khoa Hà Nội (chuyên ngành Khoa học máy tính và Kĩ thuật phần mềm),

Đại học Công nghệ - Đại học Quốc gia Hà Nội (chuyên ngành Công nghệ thông tin, Hệ thống thông tin, Mạng máy tính và Truyền thông dữ liệu), Đại học Khoa học Tự nhiên – Đại học Quốc gia TP.HCM (chuyên ngành Khoa học máy tính và Công nghệ thông tin), Đại học Công nghệ Thông tin – Đại học Quốc gia TP.HCM (chuyên ngành Công nghệ thông tin và Truyền thông), Học viện Công nghệ Bưu chính viễn thông (chuyên ngành Công nghệ thông tin và Kĩ thuật phần mềm),…

Ngoài ra, em còn có thể tham gia các khoá học trực tuyến về đào tạo lập trình viên của FUNiX, CodeGym, Coursera, edX, Udacity,… cũng như tìm hiểu thêm để thi và nhận được các chứng chỉ quốc tế như Microsoft Certified: Azure Fundamentals (lập trình với nền tảng đám mây của Microsoft), AWS Certified Developer Associate (lập trình với nền tảng đám mây của Amazon), Google Associate Android Developer (lập trình ứng dụng trên nền tảng Android của Google),…

Ngoài việc chuẩn bị về kiến thức và kĩ năng, kinh nghiệm thực tế cũng là một yêu cầu rất quan trọng trong tuyển dụng đối với các nghề thuộc nhóm nghề lập trình. Trong quá trình học tập và rèn luyện, em cần chủ động tìm kiếm cơ hội tham gia các dự án thực tế, thực tập tại các công ty, tổ chức để tích luỹ kinh nghiệm và làm giàu thêm cho hồ sơ năng lực của bản thân.

Em hãy tìm hiểu và giới thiệu về một đơn vị giáo dục gần địa phương nơi em ở có đào tạo ngành nghề thuộc nhóm nghề lập trình.

- Để trở thành một lập trình viên trong tương lai, em cần chủ động trang bị các kiến thức về ngôn ngữ lập trình, cấu trúc dữ liệu và thuật toán, cơ sở dữ liệu, hệ điều hành và mạng máy tính, bảo mật thông tin,… cũng như kĩ năng lập trình, kĩ năng phân tích và giải quyết vấn đề, kĩ năng làm việc nhóm, tư duy mở và sáng tạo, khả năng tự học và tự cập nhật kiến thức.
- Lập trình viên cần liên tục tự trau dồi, cập nhật kiến thức, chủ động tìm hiểu, học và thi các chứng chỉ quốc tế phục vụ cho công việc lập trình như Microsoft Certified: Azure Fundamentals, AWS Certified Developer Associate, Google Associate Android Developer,…

3. Nhu cầu nhân lực và các ngành nghề liên quan nhóm nghề lập trình

Lập trình viên đang là một trong những nghề thu hút rất nhiều người lao động. Khi Việt Nam gia nhập các tổ chức thương mại thế giới, nhu cầu của xã hội về nghề lập trình ngày càng tăng mạnh, không chỉ các công ty trong nước mà các công ty nước ngoài cũng có nhu cầu săn đón các lập trình viên Việt Nam vì khả năng lập trình của lao động ở Việt Nam khá tốt, làm việc cần cù, chịu khó, học hỏi. Theo Sách Trắng Công nghệ thông tin và Truyền thông Việt Nam, tổng kim ngạch xuất khẩu Công nghệ thông tin là 138 tỉ USD, tăng 11,6% so với năm 2021.

Với sự phát triển không ngừng của công nghệ và số hoá, nhu cầu nhân lực trong ngành lập trình viên tại Việt Nam dự báo đến năm 2025 sẽ cần khoảng trên 500 000 nhân lực với đa dạng lĩnh vực từ phát triển phần mềm, phát triển web, phát triển ứng dụng di động,…

Mức độ yêu cầu nhân lực lập trình viên không những tăng ở số lượng mà còn tăng ở chất lượng. Lập trình viên cần có kĩ năng chuyên môn vững vàng, khả năng giải quyết vấn đề tốt và luôn tự cập nhật kiến thức mới, sẵn sàng chinh phục những vấn đề kĩ thuật phát sinh trong quá trình xây dựng, phát triển và bảo trì phần mềm.

Với kĩ năng lập trình và khả năng giao tiếp tốt, lập trình viên còn có cơ hội làm việc cho các công ty quốc tế có trụ sở tại Việt Nam hay làm việc từ xa cho các dự án toàn cầu, mở ra cơ hội việc làm với mức thu nhập hấp dẫn.

Với đặc thù làm việc theo dự án, nhóm nghề lập trình có liên quan đến nhiều ngành nghề khác nhau thuộc lĩnh vực công nghệ thông tin cũng như các lĩnh vực liên quan. Lập trình viên cũng có cơ hội nghề nghiệp đối với các ngành nghề liên quan đến nhóm nghề lập trình như sau:

Kĩ sư dữ liệu: thu thập, phân tích và xử lí dữ liệu. Kĩ sư dữ liệu làm việc với các hệ thống cơ sở dữ liệu lớn, sử dụng các công cụ phân tích dữ liệu phát triển để đưa ra các quyết định dựa trên dữ liệu.

Chuyên gia an ninh mạng: bảo vệ hệ thống và dữ liệu khỏi các mối đe doạ bảo mật. Chuyên gia an ninh mạng phối hợp với lập trình viên thiết kế và triển khai các biện pháp bảo mật, phát hiện và phản ứng với các cuộc tấn công mạng.

Chuyên gia kiểm thử phần mềm: kiểm tra và đảm bảo chất lượng phần mềm. Chuyên gia kiểm thử phần mềm thực hiện các bài kiểm tra để phát hiện lỗi và đảm bảo phần mềm hoạt động đúng như mong đợi.

Chuyên gia phát triển trò chơi điện tử: thiết kế và phát triển các trò chơi điện tử cho nhiều nền tảng khác nhau, làm việc với các công cụ như Unity và Unreal Engine.

Chuyên gia phát triển web: thiết kế, phát triển các trang web và ứng dụng web. Chuyên gia phát triển web làm việc với các ngôn ngữ HTML, CSS, JavaScript,…

Ngoài ra, lập trình viên còn có cơ hội làm việc tại các vị trí thuộc nhóm nghiên cứu và triển khai giải pháp tại các cơ sở nghiên cứu; giảng dạy tại các cơ sở đào tạo; tư vấn các giải pháp CNTT;...

Để tìm kiếm thông tin hướng nghiệp, việc làm thuộc nhóm nghề lập trình trên mạng Internet, em có thể truy cập các địa chỉ sau:

Cổng thông tin điện tử của Bộ Lao động – Thương binh và Xã hội (https://molisa.gov.vn/): Cung cấp thông tin về các chính sách việc làm, hướng nghiệp, và đào tạo nghề, bao gồm lĩnh vực công nghệ thông tin và lập trình.

Cổng thông tin của Bộ Thông tin và Truyền thông (https://mic.gov.vn/): Cung cấp thông tin về các chương trình phát triển công nghệ thông tin, các hội thảo, và cơ hội hợp tác trong lĩnh vực lập trình và công nghệ.

Trang web của Tổng cục giáo dục nghề nghiệp (https://mhctructuyen.gdnn.gov.vn/): Cung cấp thông tin về các khoá đào tạo nghề lập trình, chương trình học và cơ hội nghề nghiệp trong ngành công nghệ thông tin.

1. Em hãy cho biết những công việc nào dưới đây liên quan đến nghề lập trình:
   a) Phát triển trang web dựa trên các module có sẵn.
   b) Viết phần mềm trên điện thoại.
   c) Phát triển giao diện của phần mềm quản lí bán hàng.
   d) Tạo ra một con robot lau nhà.

2. Em hãy truy cập trang web topdev.vn để tìm hiểu về những yêu cầu về chuyên môn do các đơn vị tuyển dụng đặt ra cùng mức đãi ngộ đối với nghề lập trình viên sử dụng Python.

- Nhu cầu nhân lực về nhóm nghề lập trình tại Việt Nam và trên thế giới ngày càng tăng, cả về số lượng và chất lượng.
- Một số ngành nghề liên quan đến nhóm nghề lập trình: kĩ sư dữ liệu, chuyên gia an ninh mạng, chuyên gia kiểm thử phần mềm, chuyên gia phát triển trò chơi điện tử, chuyên gia phát triển web.

LUYỆN TẬP

1. Trong những nhiệm vụ dưới đây, em hãy chọn ra những nhiệm vụ của lập trình viên.
   a) Tạo logo cho trang web.
   b) Sửa lỗi cho chương trình máy tính.
   c) Tạo giao diện cho phần mềm máy tính.
   d) Chỉnh sửa màu sắc, hình dáng của nút lệnh.
   e) Làm cho nút lệnh hoạt động, tương tác với người dùng.
   g) Xây dựng chức năng, cấu trúc, giao diện phần mềm máy tính.
   h) Kiểm thử, bảo trì phần mềm.

2. Em hãy tìm hiểu và giới thiệu với các bạn về một nghề có liên quan đến nhóm nghề lập trình: nhiệm vụ của nghề; yêu cầu về kiến thức, kĩ năng; nhu cầu nhân lực tại địa phương, trong khu vực và trên thế giới; mức đãi ngộ bình quân.

VẬN DỤNG

Em hãy truy cập vào website của Bộ Thông tin và Truyền thông, tìm hiểu về Sách Trắng Công nghệ thông tin và Truyền thông Việt Nam để đọc và nhận được bức tranh tổng thể về sự phát triển của Công nghệ thông tin – Truyền thông tại Việt Nam mỗi năm, qua đó rút ra được nhận xét về nhu cầu nhân lực công nghệ thông tin trong thời gian sắp tới.

Chân trời sáng tạo

PHỤ LỤC

Phụ lục 1. Các văn bản pháp luật quản lí, điều chỉnh hành vi của cá nhân, tổ chức hoạt động trong lĩnh vực Công nghệ thông tin tại Việt Nam

STT	Văn bản pháp luật	Số hiệu	Cơ quan ban hành	Ngày ban hành	Ngày có hiệu lực
1	Luật Công nghệ thông tin	67/2006/QH11	Quốc hội	29/06/2006	01/01/2007
2	Luật An toàn thông tin mạng	86/2015/QH13	Quốc hội	19/11/2015	01/07/2016
3	Luật An ninh mạng	24/2018/QH14	Quốc hội	12/06/2018	01/01/2019
4	Luật Giao dịch điện tử	20/2023/QH15	Quốc hội	22/06/2023	01/07/2024
5	Nghị định về ứng dụng công nghệ thông tin trong hoạt động của cơ quan nhà nước	64/2007/NĐ-CP	Chính phủ	10/04/2007	25/04/2007
6	Nghị định về thương mại điện tử	85/2021/NĐ-CP	Chính phủ	25/09/2021	01/01/2022
7	Nghị định về quản lí, cung cấp, sử dụng Internet và thông tin trên mạng	72/2013/NĐ-CP	Chính phủ	15/07/2013	01/09/2013
8	Nghị định về xử phạt vi phạm hành chính trong lĩnh vực bưu chính, viễn thông, tần số vô tuyến điện, công nghệ thông tin và giao dịch điện tử	15/2020/NĐ-CP	Chính phủ	03/02/2020	15/04/2020
9	Nghị định về chống tin nhắn rác, thư điện tử rác, cuộc gọi rác	91/2020/NĐ-CP	Chính phủ	14/08/2020	01/10/2020
10	Nghị định về bảo vệ dữ liệu cá nhân	13/2023/NĐ-CP	Chính phủ	17/04/2023	01/07/2023
11	Thông tư quy định về quản lí và sử dụng tài nguyên Internet	24/2015/ TT-BTTTT	Bộ Thông tin và Truyền thông	18/08/2015	10/10/2015
12	Bộ quy tắc ứng xử trên mạng xã hội	874/QĐ-BTTTT	Bộ Thông tin và Truyền thông	17/06/2021	17/06/2021

Chân trời sáng tạo

BẢNG GIẢI THÍCH THUẬT NGỮ

Thuật ngữ	Giải thích	Trang
Điều chế mã xung (Pulse Code Modulation – PCM)	Là quá trình số hoá âm thanh, thông qua việc lấy mẫu tại các điểm cố định trong khoảng thời gian nhất định.	38
Độ sâu màu (Bit depth)	Số lượng bit được dùng để biểu thị màu sắc của mỗi pixel.	40
Giao diện người dùng (User Interface – UI)	Là tất cả những thành phần mà người dùng tương tác trực tiếp trong một sản phẩm hoặc ứng dụng, bao gồm tất cả các yếu tố mà người dùng nhìn thấy và sử dụng.	170
GPS (Global Positioning System)	Là hệ thống định vị toàn cầu, sử dụng một mạng lưới vệ tinh để cung cấp thông tin về vị trí, tốc độ và thời gian cho các thiết bị trên toàn thế giới.	15
Hợp ngữ (Assembly)	Là một ngôn ngữ lập trình cấp thấp, được sử dụng cho các bộ vi xử lí hoặc các thiết bị có lập trình.	77
Học sâu (Deep Learning)	Là một lĩnh vực trong Học máy, trong đó các mô hình có thể tự trích xuất đặc trưng từ dữ liệu để học và giải quyết vấn đề.	14
Hệ thống quản lí tài nguyên doanh nghiệp (Enterprise Resource Planning – ERP)	Hệ thống dùng để hoạch định tài nguyên trong một tổ chức, một doanh nghiệp. Một hệ thống ERP điển hình là nó bao hàm tất cả những chức năng cơ bản của một tổ chức, doanh nghiệp,…	21
Biểu trưng (Logo)	Là một thiết kế biểu trưng đại diện cho một tổ chức, doanh nghiệp hoặc cá nhân. Logo có thể bao gồm hình ảnh, chữ viết hoặc kết hợp cả hai và thường được sử dụng để nhận diện thương hiệu, tạo dấu ấn cá nhân và xây dựng giá trị cốt lõi cho thương hiệu.	168
Nhận dạng kí tự quang học (Optical Character Recognition – OCR)	Là một ứng dụng công nghệ chuyên dùng để đọc văn bản từ tệp ảnh, có khả năng nhận diện các kí tự, chữ đánh máy và cả chữ viết tay.	44
Phát trực tiếp (Streaming Video)	Là một thuật ngữ được sử dụng để chỉ việc truyền tải dữ liệu video liên tục từ nguồn đến thiết bị người dùng mà không cần phải chờ đợi hoặc tải về bản hoàn chỉnh.	15
Tần số	Là số lần một hiện tượng lặp lại trên một đơn vị thời gian.	38
Trí tuệ nhân tạo (Artificial Intelligence – AI)	Là một nhánh của ngành Khoa học máy tính, liên quan đến việc tạo ra các máy tính thông minh có khả năng thực hiện các tác vụ đòi hỏi trí thông minh của con người.	13

Nhà xuất bản Giáo dục Việt Nam xin trân trọng cảm ơn các tác giả có tác phẩm, tư liệu được sử dụng, trích dẫn trong cuốn sách này.

Chịu trách nhiệm xuất bản:
Chủ tịch Hội đồng thành viên – Tổng Giám đốc NGUYỄN TIẾN THANH

Chịu trách nhiệm nội dung:
Tổng biên tập PHẠM VĨNH THÁI

Biên tập nội dung: VŨ NHÂN KHÁNH – BÙI THANH THUÝ VY
Biên tập mĩ thuật: LÂM NGUYỄN LAN TRINH
Thiết kế sách: TRẦN THỊ THANH THẢO
Trình bày bìa: ĐẶNG NGỌC HÀ – TỐNG THANH THẢO
Minh hoạ: TRẦN ANH NHÂN – TRẦN THỊ THANH THẢO
Sửa bản in: PHẠM MINH NHẬT – MÃ TRƯỜNG VINH
Chế bản: CÔNG TY CỔ PHẦN DỊCH VỤ XUẤT BẢN GIÁO DỤC GIA ĐỊNH

Bản quyền thuộc Nhà xuất bản Giáo dục Việt Nam.

Tất cả các phần của nội dung cuốn sách này đều không được sao chép, lưu trữ, chuyển thể dưới bất kì hình thức nào khi chưa có sự cho phép bằng văn bản của Nhà xuất bản Giáo dục Việt Nam.

TIN HỌC 10 – ĐỊNH HƯỚNG KHOA HỌC MÁY TÍNH
(CHÂN TRỜI SÁNG TẠO)

Mã số: Khổ 19 x 26,5 cm.
Mã số ISBN:
Số ĐKXB:
Số QĐXB:
Nộp lưu chiểu quý ... năm...

In..... bản (QĐ in số .....) tại các nhà in:

Chân trời sáng tạo

HUÂN CHƯƠNG HỒ CHÍ MINH

BỘ SÁCH GIÁO KHOA LỚP 10 – CHÂN TRỜI SÁNG TẠO

1. Toán 10, Tập một
2. Toán 10, Tập hai
3. Chuyên đề học tập Toán 10
4. Ngữ văn 10, Tập một
5. Ngữ văn 10, Tập hai
6. Chuyên đề học tập Ngữ văn 10
7. Tiếng Anh 10
   Friends Global – Student Book
8. Lịch sử 10
9. Chuyên đề học tập Lịch sử 10
10. Địa lí 10
11. Chuyên đề học tập Địa lí 10
12. Giáo dục kinh tế và pháp luật 10
13. Chuyên đề học tập Giáo dục kinh tế và pháp luật 10
14. Vật lí 10
15. Chuyên đề học tập Vật lí 10
16. Hoá học 10
17. Chuyên đề học tập Hoá học 10
18. Sinh học 10
19. Chuyên đề học tập Sinh học 10
20. Tin học 10 – Định hướng Tin học ứng dụng
21. Chuyên đề học tập Tin học 10 – Định hướng Tin học ứng dụng
22. Tin học 10 – Định hướng Khoa học máy tính
23. Chuyên đề học tập Tin học 10 – Định hướng Khoa học máy tính
24. Âm nhạc 10
25. Chuyên đề học tập Âm nhạc 10
26. Hoạt động trải nghiệm, hướng nghiệp 10 (1)
27. Hoạt động trải nghiệm, hướng nghiệp 10 (2)
28. Giáo dục quốc phòng và an ninh 10

Các đơn vị đầu mối phát hành

- **Miền Bắc:** CTCP Đầu tư và Phát triển Giáo dục Hà Nội
  CTCP Sách và Thiết bị Giáo dục miền Bắc
- **Miền Trung:** CTCP Đầu tư và Phát triển Giáo dục Đà Nẵng
  CTCP Sách và Thiết bị Giáo dục miền Trung
- **Miền Nam:** CTCP Đầu tư và Phát triển Giáo dục Phương Nam
  CTCP Sách và Thiết bị Giáo dục miền Nam
  CTCP Sách và Thiết bị Giáo dục Cửu Long

Sách điện tử: http://hanhtrangso.nxbgd.vn