

## BÀI THỰC HÀNH SỐ 9 :: HÀM (FUNCTION) NÂNG CAO

### MỤC TIÊU

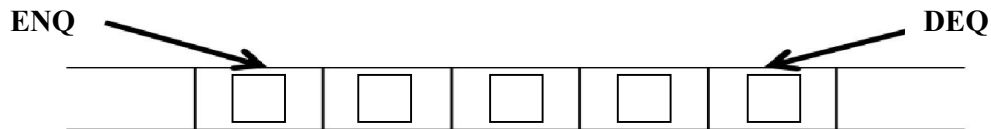
- Vận hành hàm nhiều cấp (hàm gọi hàm)
- Hiểu các khái niệm tham số hình thức, tham số thực sự, tham trị, tham biến
- Xác định được các tác vụ của bài toán, từ đó định nghĩa đúng và sử dụng được hàm để giải quyết tác vụ cho các bài toán con.

### NỘI DUNG THỰC HÀNH

#### Vấn đề 1:

Cái ống A hở 2 đầu, thực hiện các thao tác ENQ(n) và DEQ trên A như sau:

- ENQ(n): Chèn dữ liệu n vào bên trái của A
- DEQ: Lấy 1 dữ liệu ra khỏi ống A từ phía phải



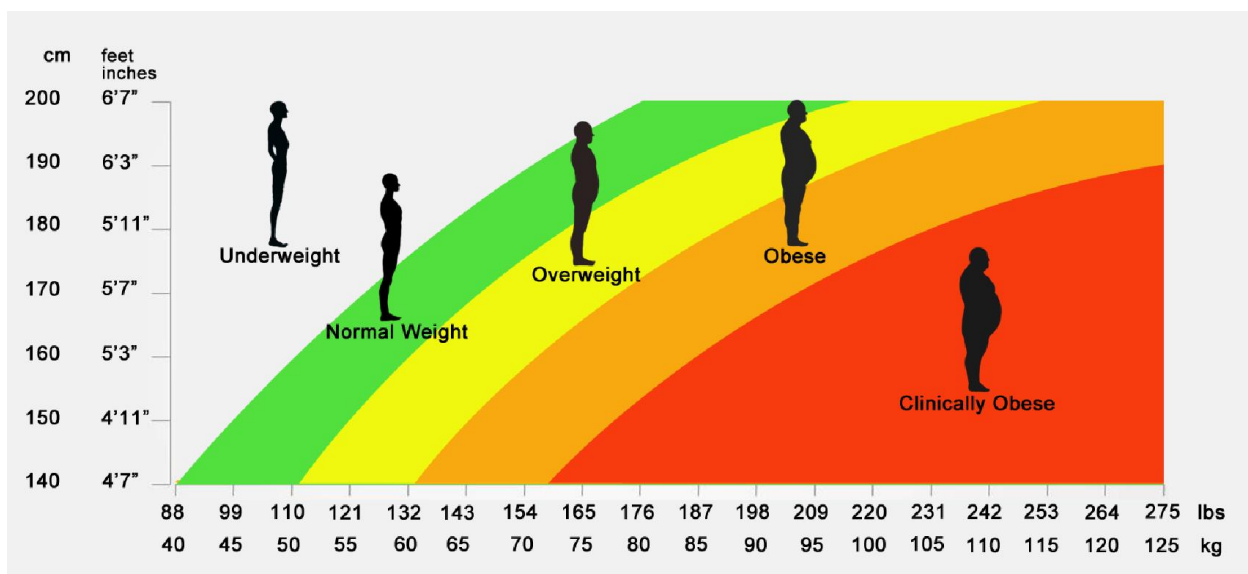
Các phép toán ENQ(1), ENQ(2), DEQ, ENQ(4), ENQ(5), DEQ, ENQ(6), DEQ and DEQ được thực hiện trên một ống A rỗng. Giá trị nào được lấy ra khỏi hàng đợi nếu thực hiện phép toán DEQ tiếp theo?

- a) 1                                      b) 2                                      c) 5                                      d) 6

#### Vấn đề 2:

Chỉ số BMI (*Body Mass Index*) cho biết cân nặng của một người có phù với chiều cao của họ hay chưa, điều này không chỉ tốt cho dáng vóc bề ngoài mà còn liên quan đến sức khỏe của bạn. Công thức tính BMI của một người khi biết cân nặng (kg) và chiều cao (m) như sau:

$$BMI = \frac{can\_nang}{chieu\_cao * chieu\_cao}$$



Và chỉ số BMI theo khuyến nghị của tổ chức Y tế thế giới (WHO) (trừ người có thai) thì:

- BMI dưới 18.5 tình trạng là “*thiếu cân*”
- BMI từ 18.5 đến 22.99 tình trạng là “*bình thường*”
- BMI từ 23 đến 24.99 tình trạng là “*thừa cân*”
- BMI > 25 tình trạng là “*béo phì*”

***Yêu cầu:***

1. Định nghĩa hàm ***bmi()*** tính và trả về chỉ số bmi của một người khi biết chiều cao (m) và cân nặng (kg) của người đó.
2. Định nghĩa hàm ***beophi()*** nhận vào chỉ số bmi và in ra màn hình tình trạng của người đó (theo khuyến nghị của WHO).
3. Viết chương trình nhập vào thông tin chiều cao, cân nặng của n người. Hiển thị bảng dữ liệu sau:

THONG KE BMI CUA CONG TY ABC				
STT	CHIEU_CAO	CAN_NANG	BMI	TINH TRANG
1	1.7	70	24.22	Thua can
2	1.69	49	17.16	Thieu can
3	1.65	54	19.83	Binh thuong
4	1.6	65	25.39	Beo phi

**Vấn đề 3:**

1. Chọn phương án đúng khi thực hiện chương trình có gọi đến hàm sub(x, y) được trình bày bên dưới. Biết rằng x là tham trị và y là tham biến.

<b>Main Program</b> $a = 2;$ $b = 3;$ $\text{sub}(b, a);$	<b>Subprogram sub(x, y)</b> $x = x + y;$ $y = x + y;$ $\text{return};$
--	---

- a)  $a=2, b=3$       b)  $a=2, b=5$       c)  $a=7, b=3$       d)  $a=7, b=5$

2. Chuyển mã giả của Program thành chương trình

**Vấn đề 4:** Tìm phân số tối giản của phân số  $\frac{a}{b}$  (a là số nguyên và b là số nguyên dương).

<i>input</i>	<i>output</i>
$\frac{2}{4}$	$\frac{1}{2}$
$\frac{1}{7}$	$\frac{1}{7}$
$\frac{-3}{6}$	$\frac{-1}{2}$

***Hướng dẫn:***

1. Định nghĩa hàm **void input()** nhận vào một phân số **phanso(a, b)**;
2. Định nghĩa hàm **void output()** hiển thị **phanso(a, b)** dạng a/b lên màn hình;
3. Cài đặt hàm **int ucln(int a, int b)** trả về ước số chung lớn nhất của 2 số nguyên không âm a và b;
4. Cài đặt hàm **void toigian()** để tối giản phân số **phanso(a,b)**;
5. Viết chương trình nhận vào 1 phân số (tử số và mẫu số). Hiển thị lên màn hình phân số tối giản của nó.

#### **Vấn đề 5:** Tính chất chia hết của số nguyên

1. Định nghĩa hàm **int sumDigits(int N)** trả về tổng các chữ số của số nguyên không âm N.
2. Định nghĩa hàm **int rightN(int N)** trả về chữ số hàng đơn vị của số N
3. Định nghĩa hàm **int div2(int N)** trả về 1 nếu N chia hết cho 2 và trả về 0 trong trường hợp ngược lại.
4. Định nghĩa hàm **int div3(int N)** trả về 1 nếu N chia hết cho 3 và trả về 0 trong trường hợp ngược lại.
5. Định nghĩa hàm **int div5(int N)** trả về 1 nếu N chia hết cho 5 và trả về 0 trong trường hợp ngược lại.
6. Định nghĩa hàm **int div6(int N)** trả về 1 nếu N chia hết cho 6 và trả về 0 trong trường hợp ngược lại.
7. Định nghĩa hàm **int div9(int N)** trả về 1 nếu N chia hết cho 9 và trả về 0 trong trường hợp ngược lại.
8. Viết chương trình nhập vào số nguyên dương N và in ra màn hình bảng thống kê.  
Chẳng hạn với N = 45, thì bảng thông kê có dạng:

Chia hết	Yes/No
2	No
3	Yes
5	Yes
6	No
9	Yes

## **BÀI TẬP**

### **Bài 0054**

**Định nghĩa:** Số hoàn hảo là số có tổng các ước số của nó gấp đôi số đó.

**Ví dụ:**

- 6 là số hoàn hảo vì 6 có các ước số là 1, 2, 3, 6 và  $1+2+3+6 = 12 = 6*2$ .
- 28 là số hoàn hảo vì 28 có các ước số là 1, 2, 4, 7, 14, 28 và  $1+2+4+7+14+28 = 56 = 28*2$ .

**Yêu cầu:**

1. Định nghĩa hàm **int hoanhao(int K)** trả về 1 nếu K là số hoàn hảo và trả về 0 trong trường hợp ngược lại.

- Viết chương trình nhập vào 1 số nguyên N, in ra màn hình tất cả các số hoàn hảo bé hơn hoặc bằng N.

**Bài 0055**

- Định nghĩa hàm *float max(float a, float b)* trả về số lớn hơn trong 2 số a và b
- Định nghĩa hàm *float min(float a, float b)* trả về số bé hơn trong 2 số a và b
- Viết chương trình nhập vào n ( $n \geq 1$ ) số thực từ bàn phím. Hiển thị lên màn hình số lớn nhất và số bé nhất trong các số vừa nhập.

Hướng dẫn:

- Nhập 1 số từ bàn phím và giả sử số lớn nhất và số bé nhất tạm thời là số vừa nhập vào.
- Lặp lại n-1 lần nhập vào 1 số, so sánh số nhập vào với max/min tạm thời để quyết định có thay đổi giá trị của min/max hay không.

**Bài 0056**

- Định nghĩa hàm *float luythua(float x, int k)* trả về  $x^k$
- Định nghĩa hàm *double giaithua(int k)* trả về  $k!$
- Định nghĩa hàm để tính *float cos(float x, float ε)* để tính  $\cos x$  với sai số  $\varepsilon$  theo công thức sau:

$$\cos x \approx 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^n \frac{x^{2n}}{(2n)!} + \dots$$

Quá trình lặp sẽ dừng khi  $\left| (-1)^n \frac{x^{2n}}{(2n)!} \right| < \varepsilon$  với  $\varepsilon$  là số đủ bé cho trước.

- Viết chương trình nhập vào giá trị của x và sai số  $\varepsilon$ . Hiển thị lên màn hình  $\cos x$ .

**Bài 0057**

- Định nghĩa hàm để tính *float sin(float x, float ε)* để tính  $\sin x$  với sai số  $\varepsilon$  theo công thức sau:

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots$$

Quá trình lặp sẽ dừng khi  $\left| (-1)^n \frac{x^{2n+1}}{(2n+1)!} \right| < \varepsilon$  với  $\varepsilon$  là số đủ bé cho trước.

- Viết chương trình nhập vào giá trị của x và sai số  $\varepsilon$ . Hiển thị lên màn hình  $\sin x$ .

**Bài 0058**

- Định nghĩa hàm *float emux(float x, float ε)* để tính  $e^x$  theo công thức sau:

$$e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$$

Quá trình lặp sẽ dừng khi  $\left| \frac{x^n}{n!} \right| < \varepsilon$  với  $\varepsilon$  là số đủ bé cho trước.

- Nhập giá trị của x và sai số  $\varepsilon$ . Tính giá trị  $e^x$ .

**Hướng dẫn:** Sử dụng lại các hàm *luythua()* và *giaithua()* ở bài tập 0056

**Bài 0059****Phân rã bài toán thành các hàm**

Viết một chương trình để tính lương thực tế khi tổng thu nhập ( $> 0$  và  $\leq 50000\ 000$ ) là đầu vào và được kiểm tra. Tiến trình xử lý dừng lại khi nhập vào tổng thu nhập là không. Sử dụng thang tính thuế sau để tính lương thực tế:

**Tổng thu nhập(VND)      Thuế**

0 - 9 000 000

0

9 000 001 – 20000 000      25% của số tiền vượt quá 9 000 000

20000 001– 50000 000      thuế của 20.000.000 đầu tiên + 40% của số tiền vượt quá 20.000.000

Tính toán và xuất ra lương thực tế cho từng người. Sau khi dùng xử lý, xuất ra trung bình lương thực tế và trung bình thuế.

**Yêu cầu 1:** Phân chia bài toán trên thành các hàm

Viết đặc tả cho từng hàm

- Đầu vào
- Xử lý
- Đầu ra

**Yêu cầu 2:** Viết chương trình giải quyết vấn đề đặt ra, có sử dụng các hàm đã định nghĩa

**Bài 0060**

**Bài toán:** Viết chương trình tính toán chi phí đậu xe trong bãi đỗ xe. Chương trình cho phép nhập vào giờ vào bãi đậu xe (làm tròn lên), giờ lấy xe ra khỏi bãi xe. Sau đó, chương trình tính ra số giờ gửi xe và hiển thị tiền phí dựa trên các mức giá sau:

**Giờ      Đơn giá/giờ**

Từ giờ đầu tiên đến giờ thứ 4

2 000

Từ giờ thứ 5 đến giờ thứ 7

1 000

Từ giờ thứ 8 đến giờ thứ 24

5 000

Chương trình cho phép nhập liên tục số giờ cho đến khi nhập giá trị 0, chương trình sẽ hiển thị các thông kê sau:

- Tổng số xe.
- Tổng số giờ.
- Tổng cộng phí đỗ xe.
- Số giờ trung bình một chiếc xe ở trong bãi đậu xe.
- Chi phí trung bình

**Yêu cầu 1:** Phân chia bài toán trên thành các hàm

Viết đặc tả cho từng hàm

- Đầu vào
- Xử lý
- Đầu ra

**Yêu cầu 2:** Hãy trình bày bảng từ điển dữ liệu

BIẾN	MÔ TẢ	GIÁ TRỊ HỢP LỆ	KIỂU

**Yêu cầu 3:** Trình bày giải thuật (Lựa chọn cách trình bày)

- Ngôn ngữ tự nhiên
- Lưu đồ
- Mã giả

**Yêu cầu 4:** Viết chương trình C thể hiện giải thuật đã lựa chọn.

**Yêu cầu 5:** Xây dựng Test case

Inputs	Expected Outputs

**Lưu ý:** Bảng phân công nhiệm vụ

Họ tên	Vai trò
	Nhóm trưởng
	Thư kí
	Trình bày

### English

. The following function  $f(n, k)$  exists:

$$f(n, k) = \begin{cases} 1 & (k=0) \\ f(n-1, k-1) + f(n-1, k) & (0 < k < n) \\ 1 & (k=n) \end{cases}$$

What is the value of  $f(4, 2)$ ?

- a) 3                      b) 4                      c) 5                      d) 6