SỞ GIÁO DỤC VÀ ĐẠO TÀO

THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG THPT NGUYỄN TẤT THÀNH

**GỢI Ý HƯỚNG DẪN HỌC SINH TỰ HỌC – HỌC KÌ II - TUẦN 01, 02**

**MÔN Tin HỌC – KHỐI 11**

| **NỘI DUNG** | |
| --- | --- |
| **Tên bài học/ chủ đề** | **Chủ đề 4: kiểu dữ liệu CÓ CẤU TRÚC**  **Bài 11. KIỂU MẢNG (ARRAY)** |
| **Hoạt động 1: *Đọc tài liệu và thực hiện các yêu cầu.*** | 1. **Tài liệu tham khảo:**  * Sách giáo khoa Tin học 11 (bản chuẩn):   Bài 11: Kiểu mảng   * Video bài giảng:   Bài 11 <https://youtu.be/3mXPOB5au-I>   * Tóm tắt kiến thức cần ghi nhớ (phụ lục 1 – đính kèm) * Mục tiêu **bài 11**: **Kiểu mảng (array)** * Về kiến thức: * Nắm được khái niệm mảng một chiều; * Nhận biết khai báo mảng một chiều theo hai cách (trực tiếp, gián tiếp); * Hiểu được một số thao tác xử lí đơn giản trong mảng một chiều (nhập, in, tính tổng, đếm…); * Về kĩ năng: * Nhận biết các thành phần trong khai báo kiểu mảng một chiều; * Biết viết khai báo mảng một chiều; * Biết cách sử dụng câu lệnh FOR…DO khi tham chiếu, xử lí các phần tử của mảng một chiều; * Về thái độ: * Giúp học sinh thấy rõ sự cần thiết và hữu ích của việc sử dụng kiểu dữ liệu có cấu trúc trong lập trình; * Bước đầu hình thành tư duy lập trình sử dụng kiểu dữ liệu có cấu trúc trong học sinh;  1. **Yêu cầu:**  * Học sinh ghi chép cẩn thận phụ lục 1 vào vở bài tập, cần đánh dấu, tô màu các cú pháp (bộ qui tắc để viết chương trình) học sinh thấy khó ghi nhớ, các ví dụ, các chú ý, các trình tự thao tác. Khi đánh dấu, tô màu, kẻ bảng, kẻ khung, gạch dưới, vẽ hình cần cẩn thận, rõ, sạch. * Trong quá trình đọc và ghi chép, nếu thắc mắc học sinh điền vào Phiếu tổng hợp thắc mắc (phụ lục 2 – đính kèm) và sớm liên hệ với giáo viên để được kịp thời giải đáp. |
| **Hoạt động 2: Kiểm tra, đánh giá quá trình tự học.** | * Hoàn thành phiếu học tập (phụ lục 3 – đính kèm), chụp và nộp lại theo yêu cầu của giáo viên. |

**PHỤ LỤC 1**

# **CHƯƠNG IV. KIỂU DỮ LIỆU CÓ CẤU TRÚC**

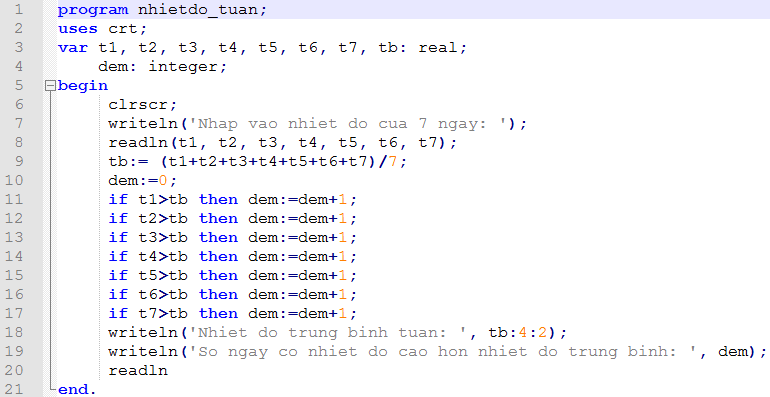
## BÀI 11. KIỂU MẢNG

1. **Kiểu mảng một chiều**

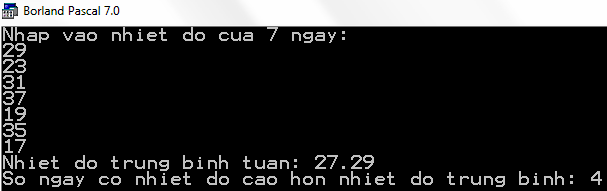
* Mảng một chiều là dãy hữu hạn các phần tử cùng kiểu.
* Mảng được đặt tên và mỗi phần tử của nó có một chỉ số.
* Để mô tả mảng một chiều cần xác định kiểu của các phần tử và cách đánh số các phần tử của nó.
* Để người lập trình có thể xây dựng và sử dụng kiểu mảng một chiều, các ngôn ngữ lập trình có qui tắc, cách thức cho phép xác định:
* Tên kiểu mảng một chiều;
* Số lượng phần tử;
* Kiểu dữ liệu của phần tử;
* Cách khai báo biến mảng;
* Cách tham chiếu đến phần tử;
* Có thể truy cập (hay thao tác) trên mỗi phần tử của mảng, trong việc làm đó mỗi phần tử của mảng được xác định bởi tên của mảng và chỉ số tương ứng của phần tử này.
* Kiểu dữ liệu có cấu trúc được xây dựng từ những kiểu dữ liệu đã có theo qui tắc, khuôn dạng do ngôn ngữ lập trình cung cấp.
* **Kiểu mảng là một kiểu dữ liệu có cấu trúc**, rất cần thiết và hữu ích trong nhiều chương trình.

**Ví dụ**, xét bài toán nhập vào nhiệt độ (trung bình) của mỗi ngày trong tuần, tính và đưa ra màn hình nhiệt độ trung bình của tuần và số lượng ngày trong tuần có nhiệt độ cao hơn nhiệt độ trung bình của tuần.

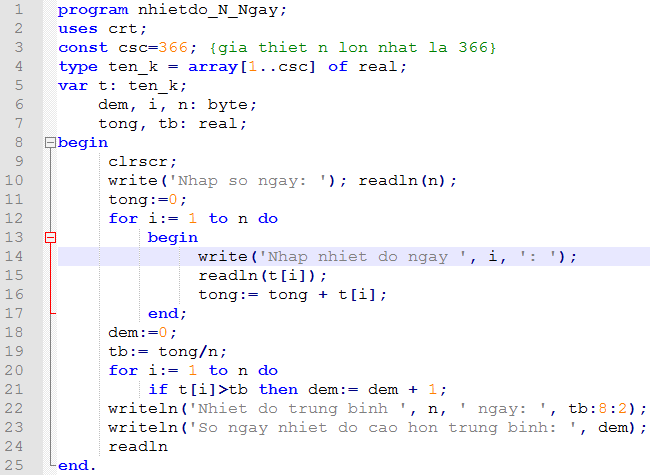
☞ Với bài toán này và với kiến thức đã được học tại thời điểm này, ta có thể dùng   
**7 biến thực** để lưu trữ nhiệt độ của các ngày trong tuần. Chương trình giải bài toán này có thể được viết bằng Pascal như sau:



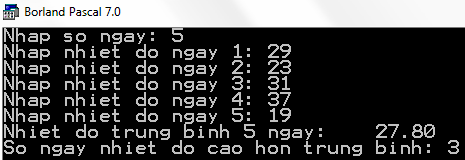
* ⇨ Khi chạy chương trình, kết quả có thể có dạng như hình dưới đây:



* Khi cần giải bài toán trên với với N ngày (N khá lớn) thì cách làm tương tự không những đòi hỏi một khối lượng khai báo khá lớn, mà đoạn chương trình tính toán cũng khá dài.
* Để giải quyết vấn đề đó, ta **sử dụng kiểu dữ liệu mảng một chiều** để mô tả dữ liệu. **Chương trình giải bài toán tổng quát** với **N ngày** trong **Pascal** có thể như sau:



⇨ Khi chạy chương trình, kết quả có thể có dạng như hình dưới đây:



Trong chương trình này đã **khai báo biến** mảng một chiều (**thông qua khai báo kiểu mảng**) như sau:

|  |  |
| --- | --- |
| type **ten\_k** = array[1..csc] of real;  var t**: ten\_k**; | Khai báo kiểu mảng một chiều gồm **csc** số thực. Khai báo biến mảng t qua kiểu mảng.  **Chú ý**: tên kiểu mảng có tên chẳng hạn là **ten\_k** thì **ten\_k** ở **dòng trên** và **dòng dưới** hoàn toàn **giống nhau**. |

1. **Khai báo**

**Khai báo mảng** một chiều có dạng tổng quát như sau:

**Cách 1.** **Khai báo trực tiếp** biến mảng:

var <tên biến mảng> : array[kiểu chỉ số] of <kiểu phần tử>;

**Cách 2.** **Khai báo gián tiếp** biến mảng **thông qua kiểu mảng**:

type **<tên kiểu mảng>** **=** array[kiểu chỉ số] of <kiểu phần tử>;

var <tên biến mảng> **:** **<tên kiểu mảng>;**

***Trong đó:***

* **Kiểu chỉ số** thường là một đoạn số nguyên liên tục có dạng n1..n2 với n1, n2 là các **hằng** hoặc biểu thức **nguyên** xác định **chỉ số đầu** và **chỉ số cuối** (n1≤n2);
* Kiểu phần tử là kiểu dữ liệu của các phần tử mảng.

**Ví dụ.**

* **Các khai báo kiểu mảng sau đây là hợp lệ**
* ArrayReal = array[-100..200] of real; hoặc A\_real = array[1..301] of real;
* arrayBoolean = array[-n+1..n+1] of boolean; {trong đó n là hằng nguyên}
* arrayInt = array[-100..0] of integer; hoặc a\_Int = array[-100..0] of integer;
* **Các khai báo kiểu mảng sau đây là KHÔNG hợp lệ**
* Ten\_k = array[3.4..4.8] of integer;

là **sai** vì kiểu chỉ số là kiểu real, kiểu KHÔNG đếm được.

* Ten\_k = array[10, 13] of integer;

là **sai** vì kiểu chỉ số không theo đúng mẫu qui định là dùng dấu **..** chứ không phải dấu phẩy (chỉ số đầu **..**chỉ số cuối ≡ hằng\_cận\_dưới **..**hằng\_cận\_trên). Máy sẽ báo lỗi ngay nếu thiếu **..**

**🖎 Nên** luyện tập **khai báo mảng theo kiểu gián tiếp** vì điều này **sẽ thuận lợi** để dùng khi khai báo tham số cho chương trình con (CTC) như Procedure, Function. Pascal ***không chấp nhận*** kiểu viết khai báo trực tiếp khi khai báo các tham số của chương trình con (sẽ được học ở chương VI) vì Mảng là kiểu dữ liệu có cấu trúc, không phải là kiểu vô hướng.

1. **Một số ví dụ**

**Ví dụ 1. Tìm phần tử lớn nhất của dãy số nguyên**

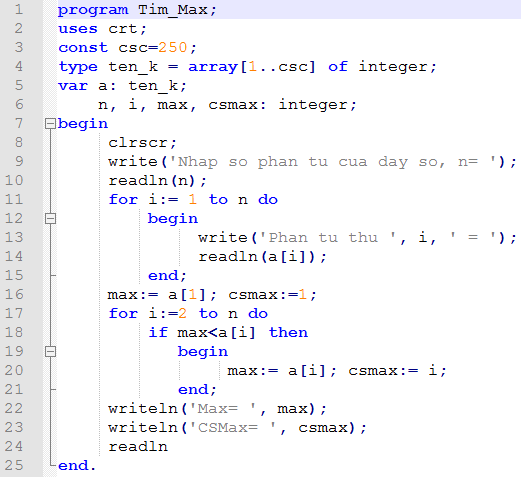
**Input**: số nguyên dương n (n ≤ 250) và dãy n số nguyên a1, a2, …, an; mỗi số đều không vượt quá 500 chẳng hạn.

**Output**: chỉ số và giá trị của phần tử lớn nhất trong dãy số đã cho (nếu có nhiều phần tử lớn nhất thì chỉ cần đưa ra một trong số chúng).

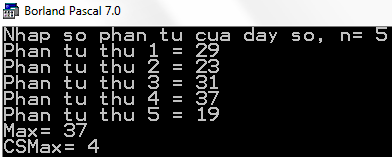
**Nhắc lại thuật toán tìm Max của dãy số (ở bài 4 trong chương trình tin học lớp 10):**

|  |
| --- |
| Bước 1. Nhập n (nguyên dương) và dãy số gồm a1, a2, …, an;  Bước 2. max:= a1, i:=2;  Bước 3. If i > n thì đưa ra giá trị max, rồi kết thúc;  Bước 4.  Bước 4.1 If max < ai then max:= ai  Bước 4.2 i:= i +1 (tăng i 1 đơn vị) rồi quay lại Bước 3 |
| ***Thuật toán tìm phần tử lớn nhất của dãy số*** |

Chương trình dưới đây thực hiện việc **duyệt tuần tự** các phần tử **để tìm ra số lớn nhất** (GTLN) của dãy số nguyên.



⇨ Khi chạy chương trình, kết quả có thể có dạng như hình dưới đây:



* Từ chương trình này, chúng ta **rút ra được một số điều cơ bản cần phải quan tâm**, đó là:
* Dùng mảng có kiểu phần tử là nguyên để biểu diễn một dãy hữu hạn số nguyên và cách khai báo mảng này.
* Câu lệnh for-do thứ nhất trong chương trình thể hiện một nhiệm vụ trong bước 1 của thuật toán, dùng để nhập các phần tử của mảng (nhập mảng). Số phần tử thực sự của mảng do người chạy chương trình nhập vào bởi câu lệnh ngay trước câu lệnh for-do này.
* Câu lệnh for-do thứ hai trong chương trình thể hiện vòng lặp (gồm bước 3 và 4 trong thuật toán), dùng để duyệt tuần tự từng phần tử trong mảng lọc lấy phần tử tạm thời là lớn nhất (trong các phần tử đã duyệt qua).

**PHỤ LỤC 2**

**PHIẾU TỔNG HỢP CÂU HỎI – THẮC MẮC   
CỦA HỌC SINH TRONG QUÁ TRÌNH TỰ HỌC – HỌC KÌ II - TUẦN 01, 02**

Trường THPT Nguyễn Tất Thành

Lớp: 11A…

Họ tên học sinh: Stt:

|  |  |  |
| --- | --- | --- |
| **BÀI** | **NỘI DUNG HỌC TẬP** | **CÂU HỎI CỦA HỌC SINH** |
| 11. Kiểu mảng (array) | Mục: …  Phần: … |  |
|  |  |  |

**PHỤ LỤC 3**

**PHIẾU HỌC TẬP**

**Phần 1. Câu hỏi**

**Bài 11. Kiểu mảng**

1. Mảng một chiều là:
2. Dãy hữu hạn các phần tử cùng kiểu. Mảng được đặt tên và mỗi phần tử của nó có một chỉ số.
3. Dãy hữu hạn các phần tử cùng kiểu. Mảng không được đặt tên và mỗi phần tử của nó có một chỉ số.
4. Dãy hữu hạn các phần tử khác kiểu. Mảng được đặt tên và mỗi phần tử của nó có một chỉ số.
5. Dãy vô hạn các phần tử cùng kiểu. Mảng được đặt tên và mỗi phần tử của nó có một chỉ số.
6. Để **khai báo biến mảng một chiều**, ta cần phải có các thông tin gì?
7. Tên biến mảng.
8. Số lượng phần tử của mảng.
9. Kiểu của các phần tử mảng.
10. Tất cả đáp án A, B và C.
11. Để khai báo **trực tiếp** biến mảng một chiều, ta sử dụng cú pháp:
12. var <tên mảng>: array[kiểu chỉ số] of <kiểu phần tử>
13. var <tên biến mảng>: array[kiểu chỉ số] of <kiểu phần tử>;
14. var <tên kiểu mảng>: array[kiểu chỉ số] of <kiểu phần tử>;
15. var <tên biến mảng>: array[kiểu phần tử] of <kiểu chỉ số>;
16. Để khai báo **gián tiếp** biến mảng một chiều, ta sử dụng cú pháp:
17. type <tên mảng> = array[kiểu chỉ số] of <kiểu phần tử>;

var <tên biến mảng>: <tên mảng>;

1. type <tên biến mảng> = array[kiểu chỉ số] of <kiểu phần tử>;

var <tên mảng>: <tên biến mảng>;

1. type <tên kiểu mảng> = array[kiểu chỉ số] of <kiểu phần tử>;

var <tên mảng>: <tên kiểu mảng>

1. type <tên kiểu mảng> = array[kiểu chỉ số] of <kiểu phần tử>;

var <tên biến mảng>: <tên kiểu mảng>;

1. Để tham chiếu tới phần tử của mảng, ta sử dụng cú pháp:
2. Tên biến mảng[chỉ số phần tử]
3. Tên biến mảng[kiểu phần tử]
4. Tên biến mảng[kiểu chỉ số]
5. Tên biến mảng[kiểu mảng]
6. Khai báo biến **DiemThi** để lưu trữ điểm thi môn tin học của 45 học sinh, ta sử dụng:
7. var DiemThi: array[1..45] of real;
8. var DiemThi: array[1..45] of integer;
9. var DiemThi: array[1...45] of real;
10. var DiemThi: array[1...45] of integer;
11. Khai báo biến **DTB** để lưu trữ điểm trung bình môn tin học của 54 học sinh, ta sử dụng:
12. var DTB: array[0..53] of real;
13. var DTB: array[0..53] of integer;
14. var DTB: array[0..54] of real;
15. var DTB: array[0..54] of integer;
16. Cho **var A: array[1..99] of integer;** để tham chiếu tới phần tử thứ **20** của biến mảng A, ta sử dụng:
17. A20
18. A[20]
19. A[1..20]
20. A[1..99]
21. Khai báo biến **DTB** để lưu trữ điểm trung bình môn tin học của 25 học sinh, ta sử dụng:
22. type KieuDTB = array[0..24] of real;

var DTB: KieuDTB;

1. type KieuDTB = array[0..25] of real;

var DTB: KieuDTB;

1. type DTB = array[0..24] of real;

var KieuDTB: DTB;

1. type DTB = array[0..25] of real;

var KieuDTB: DTB;

1. Khai báo biến **DiemThi** để lưu trữ điểm thi môn tin học của 52 học sinh, ta sử dụng:
2. type Kieu\_DiemThi = array[0..52] of real;

var DiemThi: Kieu\_DiemThi;

1. type Kieu\_DiemThi = array[1..52] of integer;

var DiemThi: Kieu\_DiemThi;

1. type Kieu\_DiemThi = array[2..53] of real;

var DiemThi: Kieu\_DiemThi;

1. type DiemThi = array[3..54] of real;

var Kieu\_DiemThi: DiemThi;

1. Cho trước **var B: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

tong:= 0;

for i:= 1 to N do

if B[i] **>** 0 then tong:= tong + B[i];

1. Gán giá trị 0 cho biến *tong*.
2. Tính tổng giá trị của các phần tử ở trong mảng.
3. Tính tổng giá trị của các phần tử lớn hơn 0 ở trong mảng.
4. Tính tổng giá trị của các phần tử nhỏ hơn 0 ở trong mảng.
5. Cho khai báo:

type KieuMangSoThuc = array[**0**..100] of real;

var MangSoThuc: KieuMangSoThuc;

để tham chiếu tới phần tử thứ **99** của biến mảng **MangSoThuc**, ta sử dụng:

1. MangSoThuc[98]
2. MangSoThuc[99]
3. KieuMangSoThuc[98]
4. KieuMangSoThuc[99]
5. Cho trước **var A: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

for i:= 1 to N do

begin

write(‘Nhap gia tri phan tu thu ‘, i, ‘: ‘); readln(A[i]);

end;

1. Nhập dữ liệu cho mảng.
2. Đưa dữ liệu của mảng ra màn hình.
3. Thông báo nhập giá trị cho từng phần tử của mảng và chờ người dùng nhập vào.
4. Tất cả đáp án A, B và C đều sai.
5. Cho trước **var A: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

write(‘Day so vua nhap vao la: ‘);

for i:= 1 to N do write(A[i]:4);

1. Hiển thị trên màn hình nội dung: “A[i]:4 “
2. Hiển thị trên màn hình nội dung: “Day so vua nhap vao la: “
3. Hiển thị trên màn hình nội dung: “Day so vua nhap vao la: A[i]:4 “
4. Tất cả đáp án A, B và C đều sai.
5. Cho trước **var A: array[1..N] of longint;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

max:= A[1];

for i:= 2 to N do

if A[i] > max then max:= A[i];

1. Tìm phần tử lớn nhất của mảng.
2. Tìm phần tử nhỏ nhất của mảng.
3. Tìm phần tử lớn nhất từ A[2] đến A[N].
4. Tất cả đáp án A, B và C đều đúng.
5. Cho trước **var A: array[1..N] of byte;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

min:= A[1];

for i:= 2 to N do

if A[i] < min then min:= A[i];

1. Tìm phần tử lớn nhất của mảng.
2. Tìm phần tử nhỏ nhất của mảng.
3. Tìm phần tử nhỏ nhất từ A[2] đến A[N].
4. Tất cả đáp án A, B và C đều đúng.
5. Cho trước **var B: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

tong:= 0;

for i:= 1 to N do

if B[i] **<** 0 then tong:= tong + B[i];

1. Gán giá trị 0 cho biến *tong*.
2. Tính tổng giá trị của các phần tử ở trong mảng.
3. Tính tổng giá trị của các phần tử lớn hơn 0 ở trong mảng.
4. Tính tổng giá trị của các phần tử nhỏ hơn 0 ở trong mảng.
5. Cho trước **var B: array[1..N] of word;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

tong:= 0;

for i:= 1 to N do tong:= tong + B[i];

1. Gán giá trị 0 cho biến *tong*.
2. Tính tổng giá trị của các phần tử ở trong mảng.
3. Tính tổng giá trị của các phần tử lớn hơn 0 ở trong mảng.
4. Tính tổng giá trị của các phần tử nhỏ hơn 0 ở trong mảng.
5. Cho trước **var B: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

tong:= 0;

for i:= 1 to N do

if B[i] **mod** 2 **= 0** then tong:= tong + B[i];

1. Gán giá trị 0 cho biến *tong*.
2. Tính tổng giá trị của các phần tử ở trong mảng.
3. Tính tổng giá trị của các phần tử là số lẻ ở trong mảng.
4. Tính tổng giá trị của các phần tử là số chẵn ở trong mảng.
5. Cho trước **var B: array[1..N] of real;** Hãy cho biết đoạn lệnh sau dùng để làm gì?

tong:= 0;

for i:= 1 to N do

if B[i] **mod** 2 **<> 0** then tong:= tong + B[i];

1. Gán giá trị 0 cho biến *tong*.
2. Tính tổng giá trị của các phần tử ở trong mảng.
3. Tính tổng giá trị của các phần tử là số lẻ ở trong mảng.
4. Tính tổng giá trị của các phần tử là số chẵn ở trong mảng.

**Phần 2. Trả lời**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Câu**  **Đáp án** | **01** | **02** | **03** | **04** | **05** | **06** | **07** | **08** | **09** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** |
| **A** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |
| **B** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |
| **C** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |
| **D** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** | **** |