

ĐỀ CƯƠNG ÔN TẬP TIN HỌC 11 – HKII – NĂM HỌC 2021 - 2022

Phần 1: Lý Thuyết

Chương IV: Kiểu dữ liệu có cấu trúc

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

Bài 11: Mảng một chiều

Mảng một chiều là dãy hữu hạn các phần tử cùng kiểu Khái báo:

C1: Var <tên biến mảng>: array[kiểu chỉ số] of <kiểu phần tử>;

C2: Type <tên kiểu mảng>= array[kiểu chỉ số] of <kiểu phần tử>;

Var <tên biến mảng>: <tên kiểu mảng>;

Tham chiếu phần tử mảng: tên biến mảng[chỉ số phần tử].

Bài 12: Kiểu dữ liệu xâu

Xâu là dãy kí tự trong bộ mã ASCII, số lượng kí tự trong xâu là độ dài của xâu, xâu có độ dài =0 gọi là xâu rỗng, kí hiệu: '', độ dài lớn nhất của xâu không vượt quá 255 Khai báo:

C1: Var <tên biến xâu>: String[độ dài lớn nhất của xâu];

C2: Type <tên kiểu xâu>= string;

Var <tên biến xâu>: <tên kiểu xâu>;

Tham chiếu phần tử xâu: tên biến xâu[chỉ số phần tử] Các thao tác xử lí thường sử dụng

- o Phép ghép xâu: (+)
- o Phép so sánh: >, <, =, <>, >=, <=
- o Các thủ tục chuẩn:
 - + Delete(s, vt, n);
 - + Insert(s1, s2, vt);
- o Các hàm chuẩn:
 - + Length(s)
 - + Copy(s, vt, n)
 - + Pos (s1,s2)
 - + Upcase(ch)

Chương V: Tập và thao tác với tập

1. Vai trò của kiểu tập

Dữ liệu kiểu tập có những đặc điểm sau:

- + Được lưu trữ lâu dài ở bộ nhớ ngoài (đĩa từ, CD, . . .) và không bị mất khi tắt nguồn điện.
- + Lượng thông tin lưu trữ trên tập có thể rất lớn và chỉ phụ thuộc vào dung lượng đĩa.

2. Phân loại tập:

Hai cách phân loại tập:

Theo cách tổ chức dữ liệu	Theo cách thức truy cập
---------------------------	-------------------------

- Tập văn bản - Tập truy cập tuần tự - Tập có cấu trúc - Tập truy cập trực tiếp

Khác với mảng, số lượng phần tử của tập không cần xác định trước.

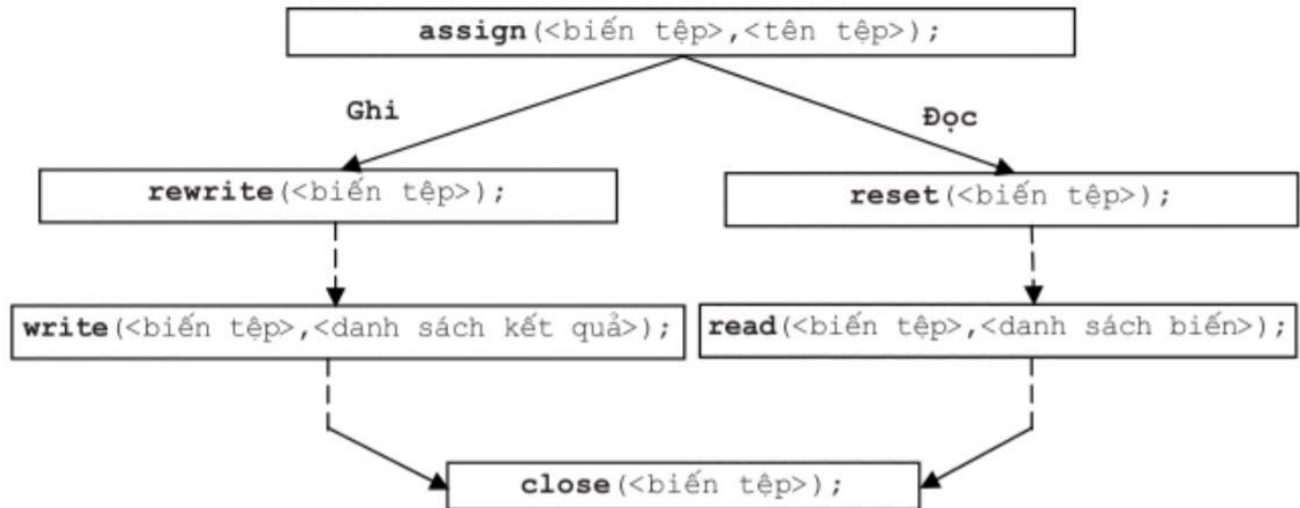
Hai thao tác cơ bản đối với tập là ghi dữ liệu vào tập và đọc dữ liệu từ tập.

3. Thao tác với tập

a. Khai báo tập văn bản

Var <Tên biến tập>: Text;

b. Thao tác với tập



* Hàm chuẩn thường dùng trong khi đọc/ghi tập văn bản:

Hàm EOF(<Tên biến tập>) trả về giá trị True nếu con trỏ tập đang chỉ tới cuối tập.

Hàm EOFLN(<Tên biến tập>) trả về giá trị True nếu con trỏ tập đang chỉ tới cuối dòng.

Chương VI: Chương trình con và lập trình có cấu trúc

1. Khái niệm chương trình con

* Khái niệm: Chương trình con (CTC) là một dãy lệnh mô tả một số thao tác nhất định và có thể được thực hiện (được gọi) từ nhiều vị trí trong chương trình.

* Lợi ích của việc sử dụng chương trình con:

- Tránh được việc phải viết lặp đi lặp lại cùng một dãy lệnh.
- Hỗ trợ việc thực hiện các chương trình lớn.
- Phục vụ cho quá trình trừu tượng hóa.
- Mở rộng khả năng ngôn ngữ.
- Thuận tiện cho phát triển, nâng cấp chương trình.

2. Phân loại và cấu trúc của chương trình con:

a. Phân loại chương trình con: có 2 loại

Hàm (từ khóa Function)	Thủ tục (từ khóa Procedure)
- Là CTC thực hiện một số thao tác nào đó - Trả về một giá trị qua tên của nó Ví dụ: sin(x), cos(x), sqr(x), length(x)...	- Là CTC thực hiện các thao tác nhất định - Không trả về giá trị nào qua tên của nó Ví dụ: readln, writeln, delete, insert...

b. Cấu trúc chương trình con:

<Phần đầu>
[<Phần khai báo>]
<Phần thân>

- * **Phần đầu:** bắt buộc phải có để khai báo tên chương trình con, nếu là hàm thì phải có khai báo kiểu dữ liệu trả về.
- * **Phần khai báo:** khai báo biến cho dữ liệu vào/ra, các hằng.
- * **Phần thân:** một dãy lệnh thực hiện nhiệm vụ nhất định trong CTC, nằm giữa 2 từ khóa Begin và End;

 Một số lưu ý:

- Tham số hình thức: là tham số được đưa vào khi định nghĩa CTC
- Tham số thực sự: là tham số được viết trong lời gọi CTC
- Biến cục bộ: là các biến được khai báo để dùng riêng trong CTC
- Biến toàn cục: là các biến được khai báo trong

chương trình chính c. Cấu trúc của hàm và thủ tục:

* **Hàm:**

```
Function<tên hàm>[(<danh sách tham số>)]:<kiểu
dữ liệu>; [<phần khai báo>]
Begin
    [<dãy các lệnh>];
    Tên hàm:= <Biểu thức>;
End;
```

Trong đó kiểu dữ liệu: integer, real, char, boolean, string là kiểu dữ liệu của giá trị mà hàm trả

về

* **Thủ tục:**

```
Procedure <tên thủ tục>[(<danh sách tham số>)];
[<phần khai báo>]
Begin
    [<dãy các lệnh>];
End;
```

Phần 2: Bài tập

I. Phần trắc nghiệm: Hãy chọn câu trả lời đúng.

Câu 1: Số phần tử của mảng một chiều là:

- a. Vô hạn
- b. Có giới hạn
- c. Có nhiều nhất 100 phần tử
- d. Có nhiều nhất 1000 phần tử.

Câu 2: Kiểu DL của các phần tử trong mảng là:

- a. Mỗi phần tử là một kiểu
- b. Có cùng một kiểu DL
- c. Có cùng một kiểu đó là kiểu số nguyên
- d. Có cùng một kiểu đó là kiểu số thực.

Câu 3: Cú pháp để khai báo mảng một chiều trực tiếp là:

- a. Var < tên biến mảng > : array[kiểu chỉ số] <kiểu phần tử>;
- b. Type < tên biến mảng > : array[kiểu chỉ số] <kiểu phần tử>;
- c. Var < tên biến mảng > : array[kiểu chỉ số] of <kiểu phần tử>;
- d. Type < tên biến mảng > : array[kiểu chỉ số] of <kiểu phần tử>;

Câu 4: Cú pháp câu lệnh sau đây là đúng?

- a. Var mang : array[1...100] of char;
- b. Var mang : array[1 - 100] of char;
- c. Var mang1c : array(1..100) of char;
- d. Var mang1c : array[1..100] of char;

Câu 5: Cú pháp câu lệnh sau đây là đúng?

- a. Type mang : array [-100 to 100] of integer;
- b. Type mang : array [-100 .. 100] of integer;
- c. Type mang = array [-100 to 100] of integer;
- d. Type mang = array [-100 .. 100] of integer;

Câu 6: Với khai báo A:array[1..100] of integer; thì việc truy xuất đến phần tử thứ 5 như sau:

- a. A(5)
- b. A[5]
- c. A5
- d. A 5

Câu 7: Cú pháp khai báo mảng 2 chiều trực tiếp là:

- a. Var < tên biến mảng > : array[Kiểu chỉ số hàng , kiểu chỉ số cột] < kiểu phần tử>;
- b. Var < tên biến mảng > : array[Kiểu chỉ số hàng ; kiểu chỉ số cột] < kiểu phần tử>;
- c. Var < tên biến mảng > : array[Kiểu chỉ số hàng, kiểu chỉ số cột] of < kiểu phần tử>;
- d. Var < tên biến mảng > : array[Kiểu chỉ số hàng ; kiểu chỉ số cột] of < kiểu phần tử>;

Câu 8: Cú pháp câu lệnh sau đây là đúng?

- a. Var mang : array[1...100,1...100] of char;
- b. Var mang : array[1...100 ; 1...100] of char
- c. Var mang2c : array(1..100,1..100) of char;
- d. Var mang2c : array[1..100 , 1..100] of char;

Câu 9: Cú pháp câu lệnh sau đây là đúng?

- a. Type mang : array [-100 to 100 , -100 to 100] of integer;
- b. Type mang : array [-100 .. 100, -100..100] of integer;
- c. Type mang = array [-100 to 100 , -100 to 100] of integer;
- d. Type mang = array [-100 .. 100, -100..100] of integer;

Câu 10: Với khai báo A:array[1..100,1..100] of integer; thì việc truy xuất đến các phần tử như sau:

- a. A(i , j)
- b. A[i , j]
- c. A(i ; j)
- d. A[i ; j]

Câu 11: Để khai báo biến xâu trực tiếp ta sử dụng cú pháp nào?

- a. Var < Tên biến > : < Tên kiểu>;
- b. Var < tên biến > : String [độ dài lớn nhất của xâu];
- c. Var < tên biến > = < Tên kiểu >;
- d. Var < tên biến > = String [độ dài lớn nhất của xâu];

Câu 12: Trong các phát biểu sau, phát biểu nào sai? a.

Xâu không có kí tự nào được gọi là xâu rỗng.

b. Thao tác nhập và xuất đối với DL kiểu xâu như nhập hay xuất giá trị của biến kiểu DL chuẩn.

c. Xâu có chiều dài không vượt quá 255.

d. Có thể tham chiếu đến từng kí tự trong xâu.

Câu 13: Trên kiểu DL xâu có các phép toán nào?

a. Phép toán so sánh và phép gán. b. Phép so sánh và phép gán.

c. Phép gán và phép nối.

d. Phép gán, phép nối và phép so sánh.

Câu 14: Trong các phát biểu sau, phát biểu nào sai?

a. Xâu A lớn hơn xâu B nếu có độ dài xâu A lớn hơn độ dài xâu B.

b. Xâu A lớn hơn xâu B nếu kí tự đầu tiên khác nhau giữa chúng kể từ trái sang, trong xâu A có mã ASCII lớn hơn.

c. Nếu A và B là các xâu có độ dài khác nhau và A là đoạn đầu của B thì A nhỏ hơn B.

d. Hai xâu bằng nhau nếu chúng giống nhau hoàn toàn.

Câu 15: Xâu 'ABBA' lớn hơn xâu:

a. 'ABC'

b. 'ABABA'

c. 'ABCBA'

d. 'BABA'

Câu 16: Xâu 'ABBA' nhỏ hơn xâu

a. 'A'

b. 'B'

c. 'AAA'

d. 'ABA'

Câu 17: Xâu 'ABBA' bằng xâu:

a. 'A'

b. 'B'

c. 'abba'

d. Tất cả đều sai

Câu 18: Xâu '2007' nhỏ hơn xâu:

a. '20007'

b. '207'

c. '1111111111'

d. '1010101010'

Câu 19: Có xâu A='abc' và xâu B='ABC' khi đó A+B cho kết quả gì?

a. 'aAbBcC'

b. 'abcABC'

c. 'AaBbCc'

d. 'ABCabc'

Câu 20: Thủ tục delete(st,p,n) thực hiện:

a. Xóa n ký tự của biến xâu st bắt đầu từ vị trí p.

b. Xóa p ký tự của biến xâu st bắt đầu từ vị trí n.

c. Xóa các ký tự của biến xâu st bắt đầu từ vị trí n đến vị trí p.

d. Xóa các ký tự của biến xâu st bắt đầu từ vị trí p đến vị trí n.

Câu 21: Cho xâu s='123456789' sau khi thực hiện thủ tục delete(s,3,4) thì:

a. s='1256789'

b. s='12789'

c. s='123789'

d. s=''

Câu 22: Thủ tục Insert(s1,s2,p) thực hiện:

a. Chèn xâu s1 vào xâu s2 bắt đầu ở vị trí p.

b. Chèn xâu s2 vào xâu s1 bắt đầu ở vị trí p.

c. Chèn p ký tự của xâu s1 vào đầu xâu s2;

d. Chèn p ký tự của xâu s2 vào đầu xâu s1;

Câu 23: Cho xâu s1='123', s2='abc' sau khi thực hiện thủ tục insert(s1, s2, 2) thì:

a. s1='123', s2='a123bc'

b. s1='1abc23', s2='abc'

c. s1='123', s2='12abc'

d. s1='ab123', s2='abc'

Câu 24: Cho s='500' hàm length(s) cho giá trị bằng:

a. 500

b. 9

c. '5'

d. '500'

Câu 25: Hàm pos(s1,s2) cho giá trị là:

a. Vị trí xuất hiện đầu tiên của s1 trong s2;

b. Vị trí xuất hiện đầu tiên của s2 trong s1;

c. Vị trí xuất hiện cuối cùng của s1 trong s2;

d. Vị trí xuất hiện cuối cùng của s2 trong s1;

Câu 26: Cho s1='010' và s2='1001010' hàm pos(s1,s2) cho giá trị bằng:

a. 3

b. 4

c. 5

d. 0

Câu 27: Hàm copy(s,p,n) cho giá trị là:

a. Một xâu gồm n kí tự liên tiếp bắt đầu từ vị trí p của xâu S.

b. Một xâu gồm p kí tự liên tiếp bắt đầu từ vị trí n của xâu S.

- c. Một xâu gồm n kí tự liên tiếp bắt đầu từ vị trí p-n của xâu S.
 d. Một xâu gồm p kí tự liên tiếp bắt đầu từ vị trí n-p của xâu S.

Câu 28: Cho xâu s='123456789' hàm copy(s,2,3) cho giá trị bằng:

- a. '234' b. 234 c. '34' d. 34

Câu 29: Khai báo nào trong các khai báo sau là sai khi khai báo xâu kí tự?

- a. var s_s: string; b. var s:string[256];
 c. var abc:string[100]; d. var cba:string[1];

Câu 30: Để định nghĩa một kiểu bản ghi mới, ta dùng cú pháp

- | | |
|--|--|
| <p>a. Type tênkiểu = record
 tên trường 1 : kiểu 1;
 tên trường 2 : kiểu 2;

 tên trường n : kiểu n;
 end;</p> | <p>b. Type tênkiểu = record
 tên trường 1 = kiểu 1;
 tên trường 2 = kiểu 2;

 tên trường n = kiểu n;
 end;</p> |
| <p>c. Type tênkiểu : record
 tên trường 1 : kiểu 1;
 tên trường 2 : kiểu 2;

 tên trường n : kiểu n;
 end;</p> | <p>d. Type tênkiểu : record
 tên trường 1 = kiểu 1;
 tên trường 2 = kiểu 2;

 tên trường n = kiểu n;
 end;</p> |

Câu 31: Để khai báo bản ghi gián tiếp ta sử dụng cú pháp:

- | | |
|---|---|
| <p>a. Type tên kiểu = record
 tên trường 1 : kiểu 1;
 tên trường 2 : kiểu 2;

 tên trường n : kiểu n;
 end;
 var tên biến : tên kiểu;</p> | <p>b. Type tên kiểu = record
 tên trường 1 = kiểu 1;
 tên trường 2 = kiểu 2;

 tên trường n = kiểu n;
 end;
 var tên biến : tên kiểu;</p> |
| <p>c. Type tên kiểu : record
 tên trường 1 : kiểu 1;
 tên trường 2 : kiểu 2;

 tên trường n : kiểu n;
 end;
 var tên biến : tên kiểu;</p> | <p>d. Type tên kiểu : record
 tên trường 1 = kiểu 1;
 tên trường 2 = kiểu 2;

 tên trường n = kiểu n;
 end;
 var tên biến : tên kiểu;</p> |

Câu 32: Trong kiểu bản ghi, mỗi bản ghi dùng để?

- a. Mô tả hay lưu trữ thông tin về nhiều đối tượng cần quản lý.
 b. Mô tả hay lưu trữ thông tin về một thuộc tính cần quản lý.
 c. Mô tả hay lưu trữ thông tin về một đối tượng cần quản lý.
 d. Mô tả hay lưu trữ thông tin về nhiều thuộc tính cần quản lý.

Câu 33: Trong các phát biểu sau, phát biểu nào đúng:

- a. Có thể nhập giá trị của một biến kiểu bản ghi như nhập giá trị của biến kiểu DL chuẩn.
 b. Có thể xuất giá trị của một biến kiểu bản ghi như xuất giá trị của biến kiểu DL chuẩn.
 c. Có thể nhập hay xuất giá trị của một biến kiểu bản ghi như nhập hay xuất giá trị của biến kiểu DL chuẩn.

d. Các thao tác nhập, xuất hay xử lý mỗi trường bản ghi phải tuân theo quy định của kiểu trường này. Câu 34: Để truy cập đến DL của trường nào đó trên bản ghi ta sử dụng cú pháp

- a. Tên trường . tên biến; b. Tên trường : tên biến;
c. Tên biến . tên trường; d. Tên biến : tên trường;

Câu 35: Kiểu Dữ liệu tệp:

- a. Được lưu trữ trên ROM b. Được lưu trữ trên RAM
c. Chỉ lưu trữ trên đĩa cứng d. Được lưu trữ ở bộ nhớ ngoài

Câu 36: Để khai báo 2 biến văn bản ta sử dụng cú pháp:

- a. var t1, t2 : text; b. var t1, t2 : txt; c. var t1.txt , t2.txt; d. var t1.txt ; t2.txt;

Câu 37: Để gán tệp ketqua.txt cho biến tệp f ta sử dụng câu lệnh

- a. f := 'ketqua.txt'; b. 'ketqua.txt' := f; c. assign(f, 'ketqua.txt'); d. assign('ketqua.txt' , f);

Câu 38: Nếu hàm eof(<biến tệp>) cho giá trị true thì con trỏ tệp nằm ở vị trí:

- a. Đầu dòng b. Đầu tệp c. Cuối dòng d. Cuối tệp

Câu 39: Chương trình con là:

- a. Một dãy lệnh mô tả một số thao tác nhất định và có thể được thực hiện từ nhiều vị trí trong chương trình.
b. Một dãy lệnh mô tả một số thao tác nhất định và trả về một giá trị qua tên của nó.
c. Một dãy lệnh mô tả một số thao tác nhất định nhưng không trả về một giá trị qua tên của nó.
d. Chương trình xây dựng từ các chương trình con khác.

Câu 40: Phát biểu nào sau đây là sai về chương trình con?

- a. Phần đầu và phần thân nhất thiết phải có, phần khai báo có thể có hoặc không.
b. Phần khai báo có thể có hoặc không có tùy thuộc vào từng chương trình cụ thể.
c. Phần đầu có thể có hoặc không.
d. Phần đầu nhất thiết phải có để khai báo tên chương trình con.

Câu 41: Sự khác nhau cơ bản giữa chương trình con thủ tục và hàm là:

- a. Hàm khác thủ tục ở phần khai báo. b. Hàm trả về giá trị còn thủ tục thì không.
c. Không có sự khác biệt d. Cả A và B.

Câu 42: Khẳng định nào sau đây là đúng?

- a. Các thủ tục và hàm đều có thể có tham số hình thức.
b. Chỉ có thủ tục mới có thể có tham số hình thức.
c. Chỉ có hàm mới có thể có tham số hình thức.
d. Thủ tục và hàm nào cũng có tham số hình thức.

Câu 43: Khẳng định nào sau đây là đúng?

- a. Lời gọi hàm bắt buộc phải có tham số thực sự còn lời gọi thủ tục thì không nhất thiết phải có tham số thực sự.
b. Lời gọi thủ tục bắt buộc phải có tham số thực sự còn lời gọi hàm thì không nhất thiết phải có tham số thực sự.
c. Cả lời gọi hàm và thủ tục đều phải có tham số thực sự.
d. Lời gọi hàm và thủ tục có thể có tham số thực sự hoặc không có tham số thực sự tùy thuộc vào từng thủ tục, hàm.

Câu 44: Từ khóa để khai báo hàm là:

- a. Program b. function c. procedure d. tất cả đều sai.

Câu 45: Lượng dữ liệu lưu trữ trên tệp:

- a. Không được lớn hơn 8K b. Không được lớn hơn 128M
c. Không được lớn hơn 1G d. Có thể rất lớn và chỉ phụ thuộc vào dung lượng đĩa.

Câu 46: Để truy xuất đến các phần tử của mảng 1 chiều ta dùng mấy vòng lặp for.....do

- a. 1 b. 2 c. 3 d. 4

Câu 47: Để truy xuất đến các phần tử của mảng 2 chiều ta dùng mấy vòng lặp for.....do

- a. 1 b. 2 c. 3 d. 4

Câu 48: Đoạn chương trình sau:

For i:=1 to n do

Begin

 Writeln('phan tu thu', i);

 Read(a[i]);

End;

- a. Đếm phần tử của mảng b. Tính tổng các phần tử của mảng
c. Nhập các phần tử của mảng. d. Xuất các phần tử của mảng.

Câu 49: Đoạn chương trình sau:

For i:=1 to n do

 If a[i] mod 2 = 0 then writeln(a[i]);

- a. In ra tất cả các số của mảng a b. In ra các số chẵn của mảng a
c. In ra các số lẻ của mảng a d. In tổng của mảng a.

Câu 50: Ta có công nhân bao gồm: họ tên, giới tính, lương, ngày công, thực lãnh. Trong đó thực lãnh =(lương * ngày công)/30.

Khai báo sau, khai báo nào đúng?

- | | |
|--|---|
| <p>a. Type cong nhan=record
 Hoten,gioitinh:string;
 Luong,ngaycong,thucanh:real;
Var cn:array[1..50] of cong nhan;</p> | <p>b. Type cong nhan=record
 Hoten,gioitinh:string;
 Luong,ngaycong,thucanh:integer;
Var cn:array[1..50] of cong nhan;</p> |
| <p>c. Type cong nhan=record
 Hoten,gioitinh:string;
 Luong,ngaycong,thucanh:real;
End;
Var cn:array[1..50] of cong nhan;</p> | <p>d. Type cong nhan=record
 Hoten,gioitinh:string;
 Luong,ngaycong,thucanh:integer;
End;
Var cn:array[1..50] of cong nhan;</p> |

II. Phần tự luận:

Bài tập 1: Viết chương trình nhập vào dãy số nguyên.

- a. Hiển thị các số đó ra màn hình.
b. Tính tổng các phần tử chẵn.
c. Đếm xem có bao nhiêu phần tử chia hết cho 3.
d. Tính giá trị trung bình của các phần tử.
e. In ra dãy số đã sắp xếp các phần tử tăng dần (giảm dần).
f. In ra danh sách các số chẵn và tính tổng của chúng.

Bài tập 2:Viết chương trình để quản lý công nhân.

Bao gồm: họ tên, Giới tính = 1 là giới tính Nam - giới tính = 0 là giới tính Nữ, lương, ngày công, thực lãnh. Trong đó thực lãnh =(lương * ngày công)/30. Với ngày công và thực lãnh là các số nguyên.

- a. Nhập dữ liệu.
b. Tính hàm thực lãnh.
c. Xuất DL ra màn hình.

Bài tập 3: Viết chương trình nhập một dãy số nguyên với (n<= 100) và tính:

- a. Đếm xem xâu đó có độ dài bao nhiêu?
b. In xâu đó ra màn hình?

c. In ra màn hình xâu ngược lại?

Bài tập sử dụng chương trình con.

Bài tập 1: Viết chương trình sắp xếp dãy số nguyên tăng dần (giảm dần).

Bài tập 2: Viết chương trình. Nhập vào một xâu.

a. Đếm xem xâu đó có độ dài bao nhiêu?

b. Đếm xem có bao nhiêu chữ số trong xâu.

c. Đếm xem có bao nhiêu ký tự trong xâu.

d. Tìm xâu đảo ngược của xâu đó và in ra màn hình.

e. Đổi xâu đó thành xâu hoa.

--- Hết ---