

章节复习

引论

一维数组

二维数组

字符数组

思维导图



C语言程序设计第五章

——数组

主讲人：李伦彬



CONTENTS

章节复习

引 论

一维数组

二维数组

字符数组

思维导图



章节复习

引论

一维数组

二维数组

字符数组

思维导图



章节复习



引论部分思维导图



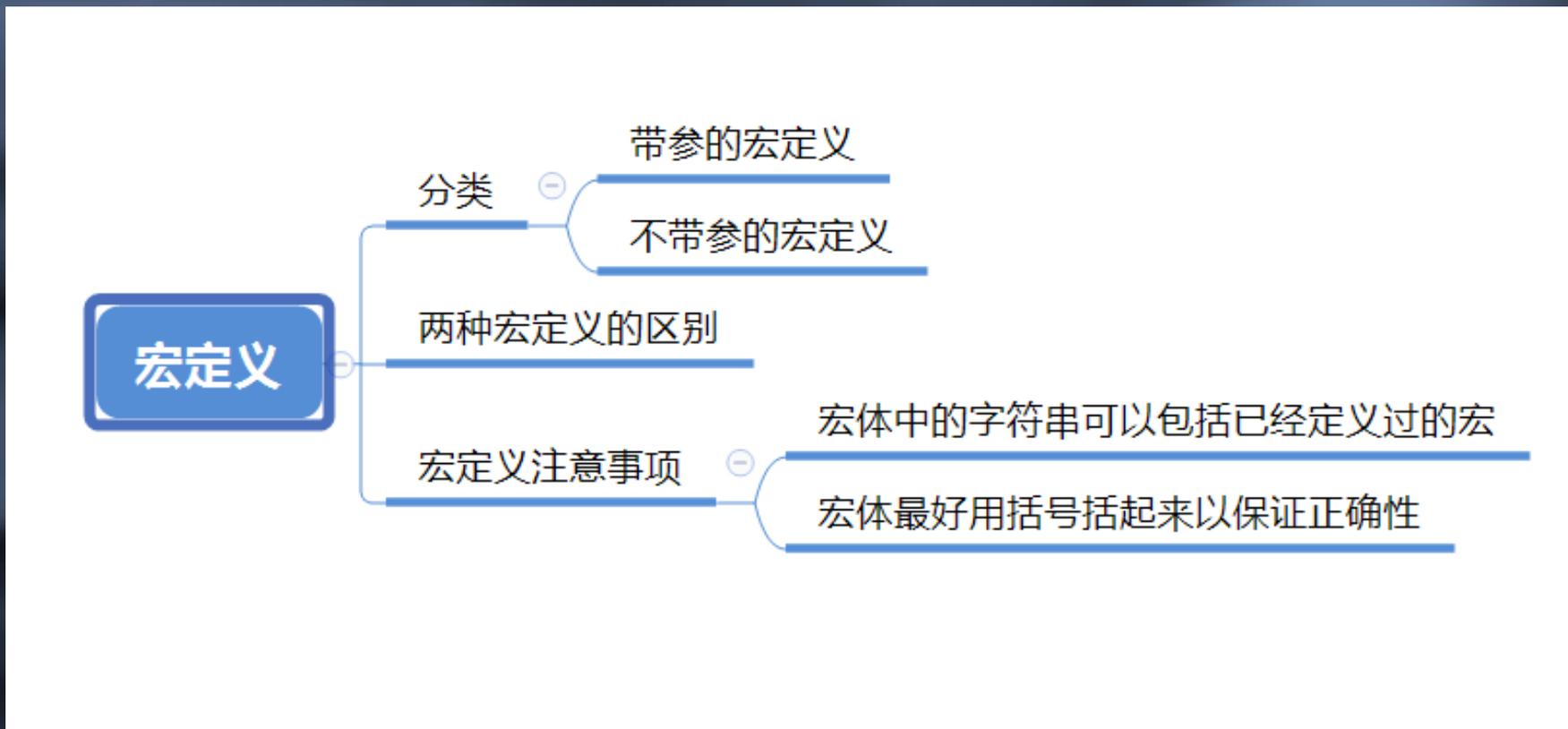
函数部分思维导图



变量部分思维导图



宏定义部分思维导图



章节复习

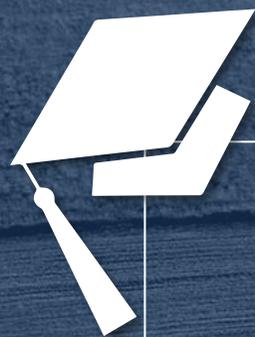
引论

一维数组

二维数组

字符数组

思维导图



引论



一个问题

float

s1, s2, s3, s4, s5..., s96, s97, s98, s99, s100;

这样做可以么？是否有更好的方法？



章节复习

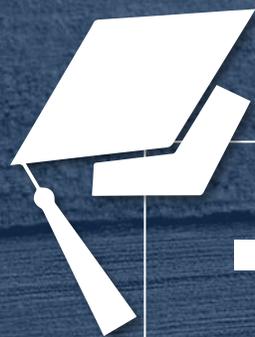
引论

一维数组

二维数组

字符数组

思维导图



一维数组





本节内容

一维数组的概念

一维数组的定义与引用

一维数组的初始化

练习巩固



数组的概念

- 数组：具有一定**顺序关系**的若干**相同类型变量**的集合,用数组名标识。
- 元素：组成数组的变量，用**数组名和下标**确定。
- 数组是一种**构造数据类型**

一维数组的定义与引用

□ 定义方式：数据类型 数组名[常量表达式]；

□ 数组元素的引用：数组名[下标]

例

表示元素个数
下标从0开始

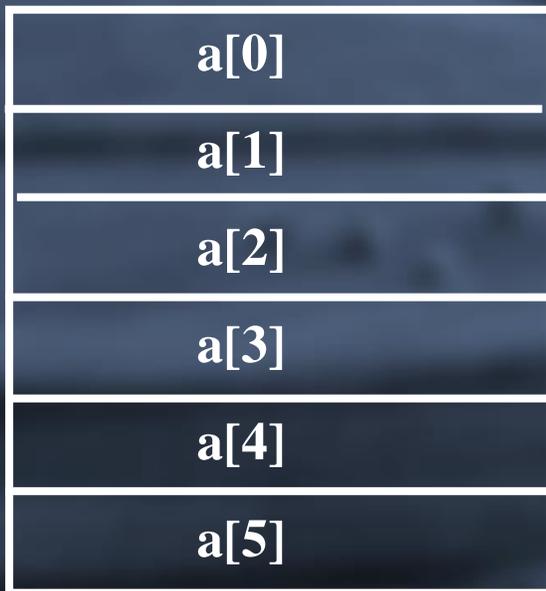
a[0], a[1], a[2], a[3], a[4], a[5]



一维数组的定义与引用

```
int a[6];
```

a → 0



数组名表示内存首地址，是地址常量

编译时分配连续内存
内存字节数=数组大小×
每元素的字节



一维数组的初始化

□ 在**定义数组时**对数组元素赋初值

```
int a[5]={1,2,3,4,5};
```

等价于: `a[0]=1; a[1]=2; a[2]=3; a[3]=4; a[4]=5;`

□ 只给**一部分元素**赋初值

如 `int a[5]={6,2,3};`

等价于: `a[0]=6; a[1]=2; a[2]=3; a[3]=0; a[4]=0;`

一维数组的初始化

□ 数组元素值全部为0

```
int a[5]={0,0,0,0,0};
```

```
或int a[5]={0};
```

□ 对整个数组元素赋初值时，可以不指定长度

```
int a[ ]={1,2,3,4,5,6};
```

编译系统根据初值个数确定数组大小



练习巩固

- Problem description
输入10名学生的成绩，并从中选出最高和最低的成绩
- Input
输入占一行。输入10个成绩，用空格分隔。
- Output
输出占一行。输出最大最小的两个成绩

```
1 #include<stdio.h>
2 /**
3  * Author: Lee
4  * Time:2018/08/22
5  * Content:一维数组练习巩固
6  * 统计十名学生的最高最低分
7  * */
8 int main(){
9     int a[10],max=-1,min=200;
10    for(int i = 0; i < 10; i++){
11        scanf("%d",&a[i]);
12        if(a[i]>max){
13            max = a[i];
14        }
15        if(a[i]<min){
16            min = a[i];
17        }
18    }
19    printf("max = %d,min = %d",max,min);
20
21    return 0;
22 }
```

```
1 3 5 7 10 5 6 8 9 4
max = 10,min = 1
-----
Process exited with return value 0
Press any key to continue . . .
```

微软拼音 半 :

章节复习

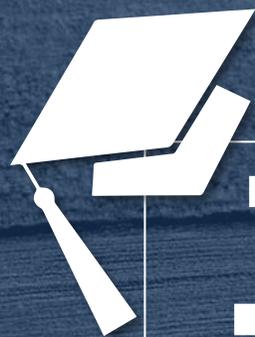
引论

一维数组

二维数组

字符数组

思维导图



二维数组





本节内容

二维数组的定义

二维数组的引用

二维数组的初始化

练习巩固



二维数组的定义

□ 定义方式：

数据类型 数组名[常量表达式][常量表达式]；

例 `int a[3][4];` `float b[2][5];`

□ 数组元素的存放顺序

按行序优先

元素个数=行数*列数



二维数组的理解

二维数组a是由3个元素组成

例 `int a[3][4]`

<code>a[0][0]</code>	<code>a[0][1]</code>	<code>a[0][2]</code>	<code>a[0][3]</code>
<code>a[1][0]</code>	<code>a[1][1]</code>	<code>a[1][2]</code>	<code>a[1][3]</code>
<code>a[2][0]</code>	<code>a[2][1]</code>	<code>a[2][2]</code>	<code>a[2][3]</code>

每个元素`a[i]`由包含4个元素的一维数组组成

0	<code>a[0][0]</code>	<code>a[0]</code>
1	<code>a[0][1]</code>	
2	<code>a[0][2]</code>	
3	<code>a[0][3]</code>	
4	<code>a[1][0]</code>	<code>a[1]</code>
5	<code>a[1][1]</code>	
6	<code>a[1][2]</code>	
7	<code>a[1][3]</code>	
8	<code>a[2][0]</code>	<code>a[2]</code>
9	<code>a[2][1]</code>	
10	<code>a[2][2]</code>	
11	<code>a[2][3]</code>	



二维数组的引用

□ 形式：**数组名 [下标] [下标]**

- 下标是整型或字符型的**常量**，**变量或表达式**。

如：`a[1][2]` `a[i][j]`

- 数组元素可出现在表达式中，也可被赋值。

如：`a[1][2]=a[2][2]/2`

- 使用数组元素时，应注意**不要超出其定义的范围**。

如：`int a[2][3];`

二维数组的初始化

- 分行初始化
- 按元素排列顺序初始化

第一维长度省略初始化

```
例 int a[][3]={1,2,3,4,5,6};
```

1	2	3	4	5	6
a[0][0]	a[0][1]	a[0][2]	a[1][0]	a[1][1]	a[1][2]





练习巩固

- Problem description
创建一个 $M*N$ 的int型二维数组，并为其赋值。输出所有元素之和
- Input
输入占M行，每行N个元素。
- Output
输出占一行。输出该二维数组所有元素之和

```
1 #include<stdio.h>
2 /**
3  * Author: Lee
4  * Time:2018/08/22
5  * Content:二维数组练习巩固
6  * 计算二维数组元素和
7  * */
8 #define M 3
9 #define N 2
10 int main(){
11     int a[M][N],sum = 0;
12     //printf("M = %d,N = %d\n",M,N);
13     for(int i = 0; i < M; i++){
14         for(int j = 0; j < N; j++){
15             scanf("%d",&a[i][j]);
16             sum += a[i][j];
17         }
18     }
19     printf("sum = %d",sum);
20
21     return 0;
22 }
23
24
25
```

```
1 2
3 4
5 6
sum = 21
-----
Process exited with return value 0
Press any key to continue . . .
```

微软拼音 半 :

章节复习

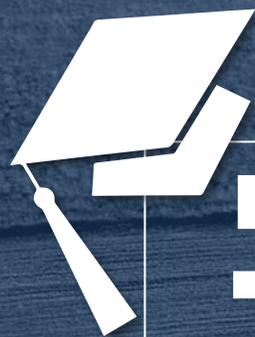
引论

一维数组

二维数组

字符数组

思维导图



字符数组





本节内容

字符数组的定义、引用
和初始化

字符数组的输入输出

练习巩固



字符串

- 无字符串变量，用字符数组处理字符串
- 字符串结束标志：'\0'

例 “hello”共5个字符，在内存占6个字节



字符数组的定义

□ 形式：

- char **数组名[常量表达式]**
- char **数组名[常量表达式][常量表达式]**
- 常量表达式：整数、字符、符号常量

例 char c[10],
ch[3][4];

- 一维字符数组：可存放一个字符串（每个数组元素存放一个字符）
- 二维字符数组：可存放多个字符串（行数是字符串的个数）、
- 可以用整型数组存放字符型数据，但**浪费存储空间**。

字符数组的引用

□ 形式：

- 数组名[下标]
- 数组名[下标][下标]

例：a[3], b[1][2]

字符数组的初始化

- 逐个字符赋值
- 用字符串常量



```
例 char ch[5]={ "good" };
char ch[5]= "good" ;
char ch[ ]= "good" ;
```

g	o	o	d	\0
ch[0]	ch[1]	ch[2]	ch[3]	ch[4]

字符数组的初始化

二维字符数组初始化

```
例 char fruit[][7]={ "Apple" , " Orange" ,  
                    " Grape" , " Pear" , " Peach" };
```

fruit[0]	A	p	p	l	e	\0	\0
fruit[1]	O	r	a	n	g	e	\0
fruit[2]	G	r	a	p	e	\0	\0
fruit[3]	P	e	a	r	\0	\0	\0
fruit[4]	P	e	a	c	h	\0	\0



字符串的输入输出

1. 用格式符“%c” 逐个输入和输出

输入时，最后人为加入‘\0’，构成字符串

输出时用‘\0’作为输出结束标志。

```
1 char a[10];
2 int i;
3 for (i = 0; i < 9; i++)
4     scanf("%c",&a[i]);
5
```

字符串的输入输出

2.用 "%s" 格式符整体输入和输出

(1)使用printf输出字符串

```
1  #include<stdio.h>
2  int main(){
3      char c[10] = "hello";
4      printf("%s",c);
5      return 0;
6  }
7
```

字符串的输入输出

2.用 "%s" 格式符**整体**输入和输出

(2)使用scanf输入字符串

```
1  #include<stdio.h>
2  int main(){
3      char c[10];
4      scanf("%s",c);
5      return 0;
6  }
```

字符串的输入输出

3. 用字符串处理函数进行输入和输出

(1) 使用puts()函数输出字符串

```
1  #include<stdio.h>
2  int main(){
3      char c[10] = "hello";
4      puts(c);
5      return 0;
6  }
```

字符串的输入输出

3.用字符串处理函数进行输入和输出

(1)使用gets()函数输入字符串

```
1  #include<stdio.h>
2  int main(){
3      char c[10];
4      gets(c);
5      return 0;
6  }
7
```

字符串处理函数

头文件：`#include <string.h>`

1. 求字符串长度函数 `strlen()`

格式：`int strlen(字符数组名)`

功能：统计字符数组中字符串的长度(不包括' \0')，
并将其作为函数值返回



字符串处理函数

2. 字符串复制函数strcpy()

格式：`strcpy(字符数组1, 字符数组2)`

功能：将以“字符数组2”为首地址的字符串复制到以

“字符数组1”为首地址的字符数组中，即把“字符数组2”的值拷贝到“字符数组1”中。



字符串处理函数

3.字符串比较函数strcmp()

格式：**strcmp(字符数组1,字符数组2)**

功能：比较“字符数组1”和“字符数组2”的大小，规则：**逐个字符比较ASCII码**，直到遇到不同字符或'\0'。

若相等，返回值为0

若小于，返回值为-1

若大于，返回值为1

字符串处理函数

4.字符串连接函数strcat()

格式：**strcat(字符串数组1,字符串数组2)**

功能：把“字符串数组2”和“字符串数组1”的后面，从

“字符串数组1”原来的‘\0’ (字符串结束标记)处开始

连接，‘\0’被覆盖，结果存放在“字符串数组1”中。





练习巩固

- Problem description
自己编写并实现strcat函数的相关功能
- Input
输入占两行，包含字符串1和字符串2
- Output
输出占一行。输出两个字符串拼接后的字符串3

```
1 #include<stdio.h>
2 #include<string.h>
3 char* myStrcat(char* str1,char* str2){
4     int len = strlen(str1);
5     for(int i = 0;i <= strlen(str2);i++){
6         str1[len+i] = str2[i];
7     }
8     return str1;
9 }
10
11 int main(){
12     char c[2][100];
13     scanf("%s%s",c[0],c[1]);
14     printf("%s",myStrcat(c[0],c[1]));
15     return 0;
16 }
17
18
19
```

```
hello
world
helloworld
-----
Process exited with return value 0
Press any key to continue . . .
```

所有错误:	0
总错误:	0
输出大小:	102.9345703125 KiB
中止	

编译器: MinGW GCC 4.8.1 32-bit Debug
执行 g++.exe...
g++.exe "C:\Users\Eric\Desktop\C语言课件\Chapter05-数组\03-练习\Chapter05_homework03.cpp" -o "C:\Users\Eric\Desktop\C语言课件\Chapter05-数
编译 1.17 秒后成功

章节复习

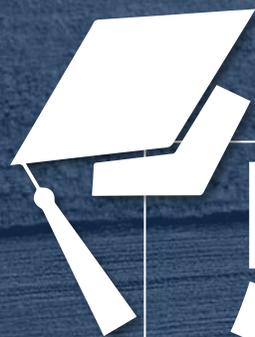
引论

一维数组

二维数组

字符数组

思维导图



思维导图



引论部分思维导图

数组

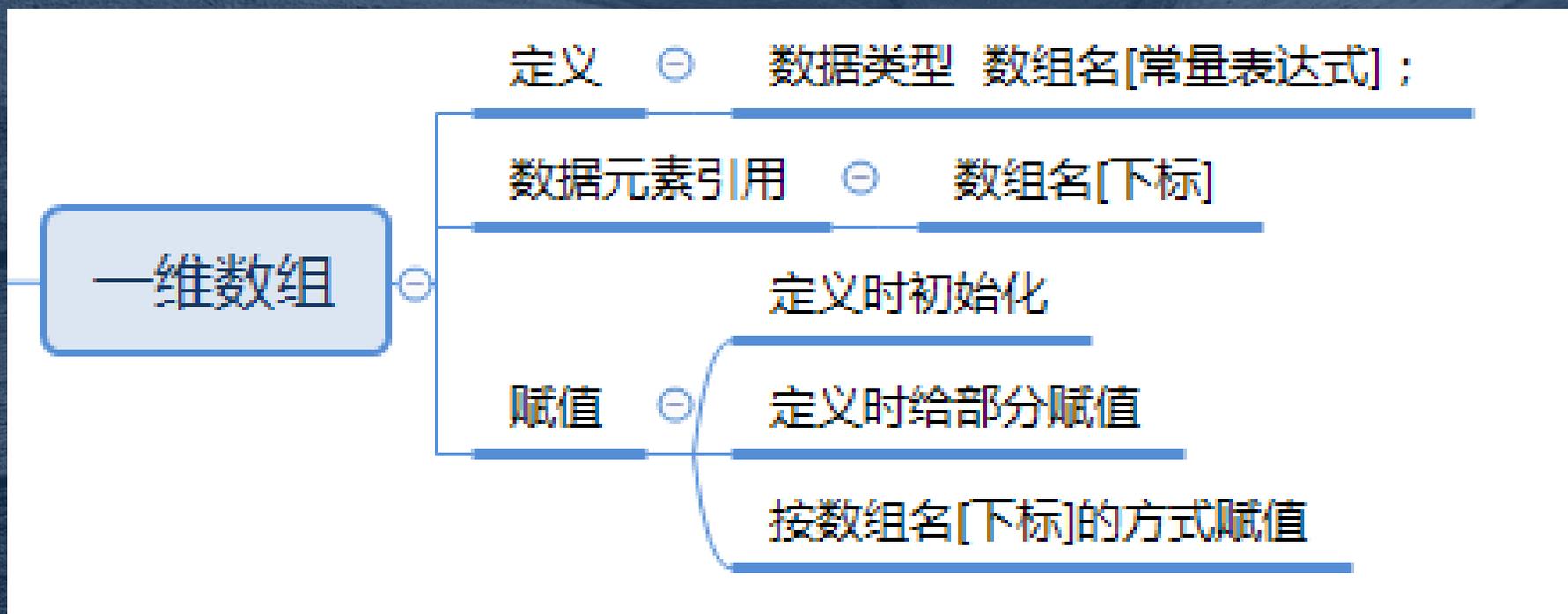
具有一定顺序关系的若干相同类型变量的集合,用数组名标识

元素:组成数组的变量,用数组名和下标确定

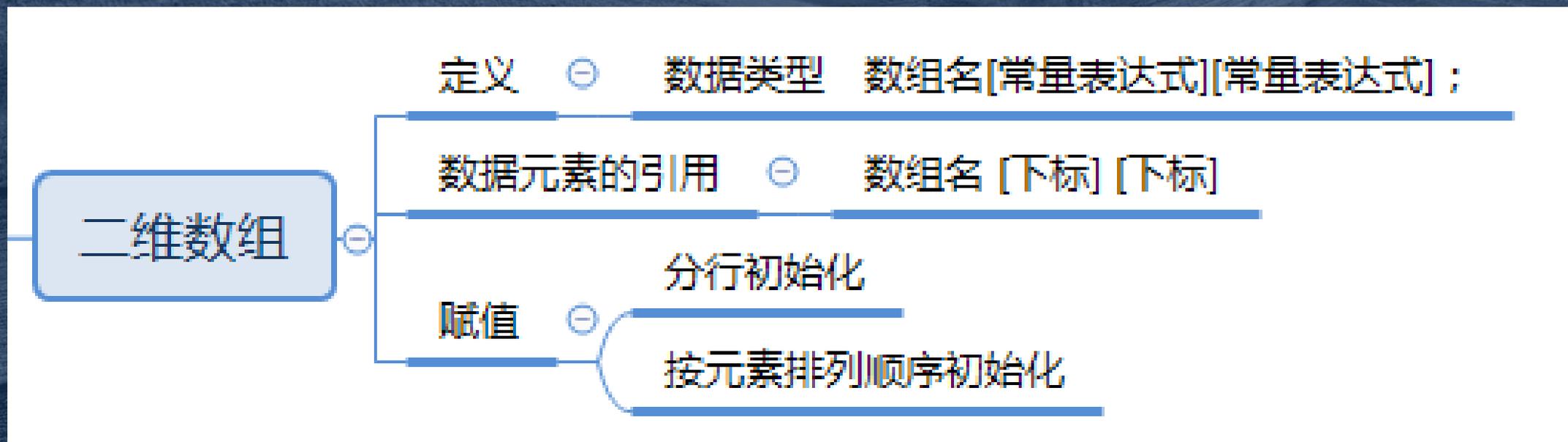
数组是一种构造数据类型



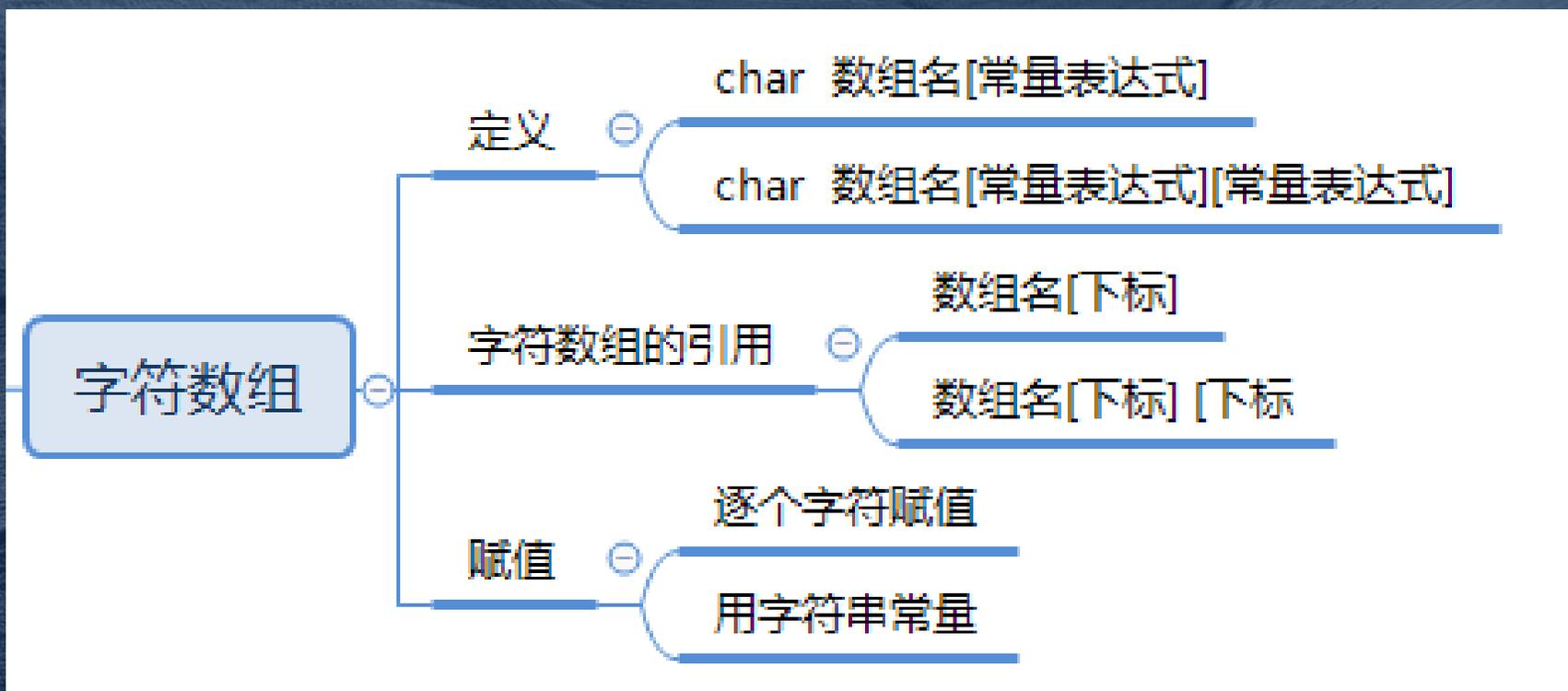
一维数组部分思维导图



二维数组部分思维导图



字符数组部分思维导图



教学评价

完成本章的作业和练习题后，请在QQ群中根据自己对本章知识的掌握情况，进行投票，我将根据同学们的投票，及时了解同学们对本章的掌握情况，并做出教学调整。

< 返回

投票

发布投票

进行中



★李伦彬135552966

24秒前

完成本章作业后，认真复习和总结，你掌握本章知识了吗？

选项预览

共0票

- A. 已掌握，可独立完成作业。
- B. 基本掌握，参考课件和微视频，可完成作业...
- C. 没有掌握。

立即投票



练习题库-笔试

FSCapture

练习题库-上机

Dev-Cpp编程工具

自测系统

ZoomIt

Xmind-8



课件下载网址：

<http://www.jsjx.com/index.aspx>

[推荐资料1 - C语言经典编程282例](#)

[推荐资料2 - 华为C语言编程规范](#)

[推荐资料3 - 嗨翻C](#)

[推荐资料4 - C&C++ API.chm](#)



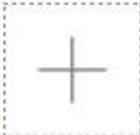
程序社团-技术交流群 : 540653133



Programming Association
540653133

发消息

首页 成员 设置 [编辑资料](#)

群介绍 本群创建于2017/10/17: 群主很懒,什么都没有留下

群标签 [行业交流](#) [IT/互联网](#) [程序设计](#) [IT](#) [计算机](#)

群主/管理员        

成员分布 (136/200)

66%	13%	84%	16%
男-91人	黑河-19人	90后-115人	单身-23人

本群星级  该群一切正常

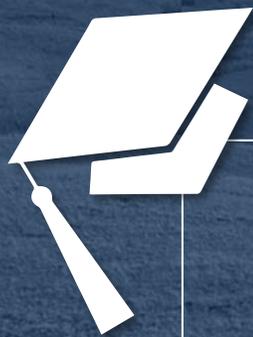


课件下载网址：

<http://jsjxy.hhhxy.cn/info/1074/2602.htm>

课件下载二维码：





感谢聆听

下一章内容：地址和指针

主讲人：李伦彬

请同学们提前预习课本相关知识点

