Problem Solving With C - UE24CS151B

Strings in C

Prof. Sindhu R Pai
PSWC Theory Anchor, Feb-May, 2025
Department of Computer Science and Engineering

PROBLEM SOLVING WITH C
Strings in C

1. Introduction

2. Declaration

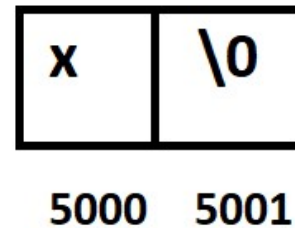3. Initialization

4. Demo of C Code

5. String v/s Pointer

Introduction

- An array of characters and terminated with a special character '\0' or NULL. ASCII value of NULL character is 0.

- String constants are always enclosed in double quotes. It occupies one byte more to store the null character.

- Example: **"X" is a String constant**

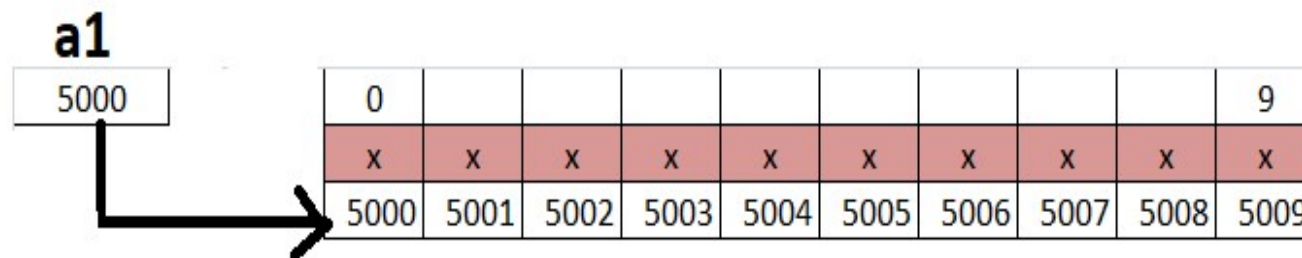| x	\0
5000	5001

Declaration

Syntax: **char variable_name[size];** //Size is compulsory

       char a1[10];   // valid declaration
       char a1[];   // invalid declaration

a1

5000

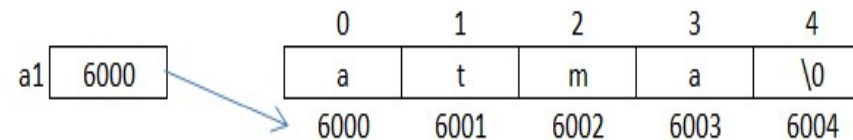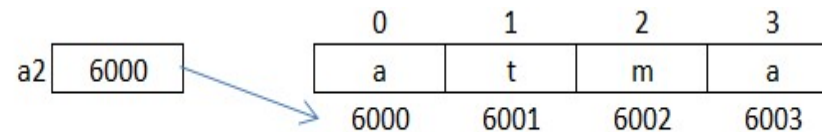| 0									9
x	x	x	x	x	x	x	x	x	x
5000	5001	5002	5003	5004	5005	5006	5007	5008	5009

Initialization

Syntax: **char variable_name[size] = {Elements separated by comma};**

Version 1:
- char a1[] = {'a', 't', 'm', 'a', '\0' };
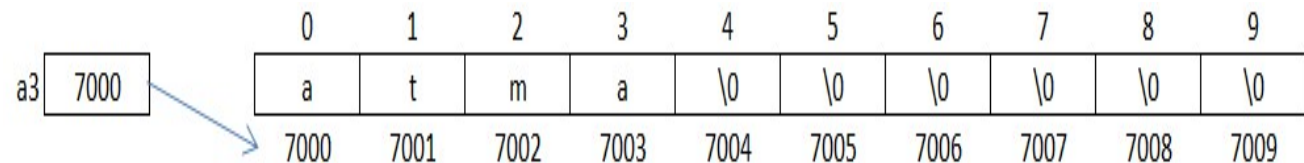- Shorthand notation: char a1[] = "atma";

	0	1	2	3	4
a1 6000	a	t	m	a	\0
	6000	6001	6002	6003	6004

Version 2:   char a2[] = {'a', 't', 'm', 'a' };

	0	1	2	3
a2 6000	a	t	m	a
	6000	6001	6002	6003

Version 3: Partial initialization
- char a3[10] = {'a','t','m','a'};

	0	1	2	3	4	5	6	7	8	9
a3 7000	a	t	m	a	\0	\0	\0	\0	\0	\0
	7000	7001	7002	7003	7004	7005	7006	7007	7008	7009

Initialization continued..

Version 4: Partial initialization
- char a4[10] = {'a','t','m','a', '\0'};
- char a4[10] = "atma";

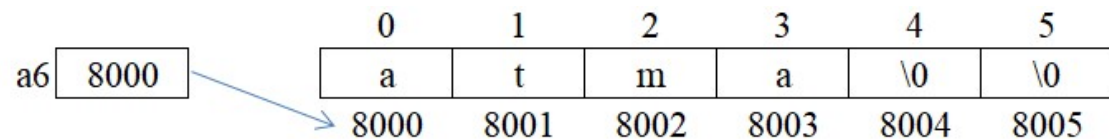|	0	1	2	3	4	5	6	7	8	9
a4 8000	a	t	m	a	\0	\0	\0	\0	\0	\0
	8000	8001	8002	8003	8004	8005	8006	8007	8008	8009
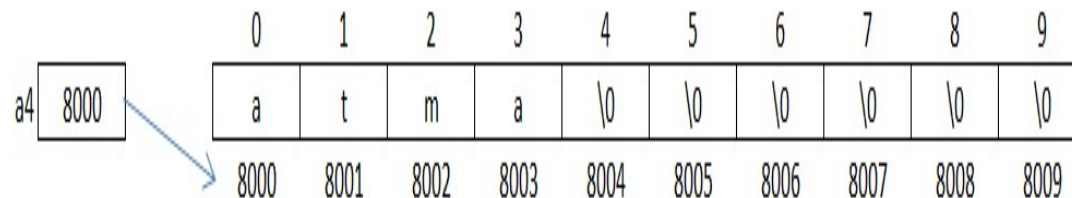
Version 5: char a5[ ] = {'a', 't', 'm', 'a', '\0', 't', 'r', 'i', 's', 'h', 'a', '\0' };

	0	1	2	3	4	5	6	7	8	9	10	11
a5 9000	a	t	m	a	\0	t	r	i	s	h	a	\0
	9000	9001	9002	9003	9004	9005	9006	9007	9008	9009	9010	9011

Version 6: char a6[ ] = "atma\0" ;

	0	1	2	3	4	5
a6 8000	a	t	m	a	\0	\0
	8000	8001	8002	8003	8004	8005
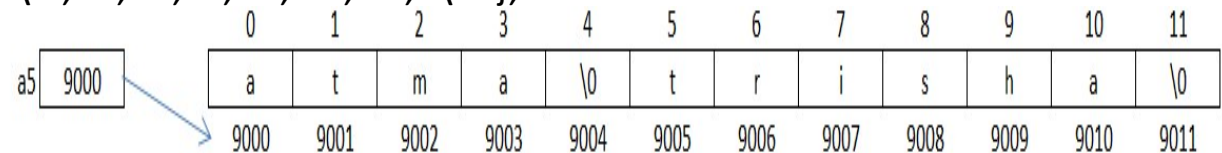
Version 7: char a7[ ] = "atma\\0" ;

	0	1	2	3	4	5	6
a7 8000	a	t	m	a	\	0	\0
	8000	8001	8002	8003	8004	8005	8006

Version 8: char a8[ ] = "at\0ma" ;

Demo of C Code

- To read and display a string in C
- Points to note:
  - If the string is hard coded, it is programmer's responsibility to end the string with '\0' character.
  - scanf terminates when white space is given in the user input.
  - scanf with %s will introduce '\0' character at the end of the string. printf with %s requires the address and will display the characters until '\0' character is encountered
  - If you want to store the entire input from the user until user presses new line in one character array variable, use [^\n] with %s in scanf

String v/s Pointer

- char x[] = "pes"; // x is an array of 4 characters with 'p', 'e', 's', '\0'

  Stored in the Stack segment of memory

- Can change the elements of x. x[0] = 'P';

- Can not increment x as it is an array name. x is a constant pointer.

- char *y = "pes";

  y is stored at stack. "pes" is stored at code segment of memory. It is read only.

- y[0] = 'P' ; // undefined behaviour

- Can increment y . y is a pointer variable.

THANK YOU

Department of Computer Science and Engineering

Dr. Shylaja S S, Director, CCBD & CDSAML, PESU
Prof. Sindhu R Pai - sindhurpai@pes.edu