|  |  |  |
| --- | --- | --- |
|  | **char arr[] = "Hello"** | **char \*ptr = "Hello"** |
| **type** | `arr` is an array | `ptr` is a pointer variable  This is similar to:  /\* \_\_unnamed is magic because modifying it gives UB. \*/  static char \_\_unnamed[] = "Hello";  char \*ptr = \_\_unnamed; |
| **size** | sizeof(arr) = 6bytes for array    a == &a  printf("%p\n", a); // F804DFF6EA  printf("%p\n", &a); // F804DFF6EA | sizeof(ptr) = 6bytes for array + 4/8bytes for pointer .    p != &p  printf("%p\n", p); // F693395050  printf("%p\n", &p); // F804DFF6E0 |
| **storage** | "Hello" is stored in the **stack** section of the memory | `ptr` is stored at **stack** but "Hello" is stored in a **read-only block** (generally in data segment) |

|  |  |  |
| --- | --- | --- |
| **assignment** | char a[] = "hello";  a = "new"; // invalid  because array is a **constant pointer.**  !! We can assign a new string to arr by using gets(), scanf(), strcpy() or by assigning characters one by one.  gets(a);  scanf("%s", a);  strcpy(a, "new");  a[0] = 'n';  a[1] = 'e';  a[2] = 'w';  a[3] = '\0';  //also  cin >> a;  getline(a,n,d); | char \*p = "hello";  p = "new"; // valid    the pointer can be changed to pointer another char[] (as in above case) but it cannot change the element in the array. Modifying a string literal causes undefined behavior, so the following operations are **invalid**.  char \*ptr = "Hello";  ptr[0] = 'Y'; or \*ptr = 'Y';  gets(name);  scanf("%s", ptr);  strcpy(ptr, "source");  strcat(ptr, "second string");  // "hello" is in read-only data-segment, we cannot change  This is because when a string value is directly assigned to a pointer, in most of the compilers, it’s stored in a **read-only block** (generally in data segment) that is shared among functions.  //we are recommended to write as:  const char\* hello2 = "hello";  // but if create dynamically allocate the string, we can change the element;  int n, i;  char \*ptr;  scanf("%d", &n);  **ptr = (char \*)malloc(n \* sizeof(char));**  for (i = 0; i < n; i++) {  printf("Enter ptr[%d]: ", i);  scanf(" %c", ptr + i);  }  \*(ptr) = 'H'; // right, because the content hello3 points to lies on stack like other variables |

|  |  |  |  |
| --- | --- | --- | --- |
| 1000 | 1001 | 1002 | 1003 |
| n | e | w | \0 |

|  |
| --- |
| 2000…2007 |
| ptr |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1000 | 1001 | 1002 | 1003 | 1004 | 1006 |
| h | e | l | l | o | \0 |

6bytes is allocated to store 6 characters + additional 8 bytes to store pointer

arr[]