

Linguagens de Programação 1

Francisco Sant'Anna

Sala 6020-B

`francisco@ime.uerj.br`

`http://github.com/fsantanna-uerj/LP1`

Vetores
Arrays
Strings

Vetores

```
int v;
```

```
int vs[4];
```

```
v = 10;
```

```
printf("%d", v);
```

```
vs = ?;
```

```
printf("%?", vs);
```

endereço	id	valor
	v	
	vs	
	ys	

Vetores

```
int xs[4];
```

```
int ys[4];
```

```
printf("%p %p\n", &xs, &ys);
```

```
printf("%p %p %p %p %p %p\n",  
       xs, xs+0, xs+1, xs+2, xs+3, xs+4);
```

endereço	id	valor
	xs	
	ys	
	...	

Vetores

```
int xs[4];
```

```
int ys[4];
```

```
*(xs+3) = 10; // xs[3]=10
```

```
printf("%d\n", xs[3]);
```

```
printf("%p %p\n", (xs+3), &xs[3]);
```

```
scanf("%d", &xs[1]);
```

endereço	id	valor
	XS	
	YS	
	...	

Exercícios 1 - X

- No site:
- <https://github.com/fsantanna-uerj/LP1/blob/master/Exercicios/lp1-06-vetores.md>

Arrays

- Vetor: array de dimensão 1
 - `int vs[2];`
- C também suporta arrays de múltiplas dimensões
 - `int vs[3][2];`
 - 3 linhas e 2 colunas

Arrays

```
int vs[3][2] = { {1,2},{3,4},{5,6} };  
printf("%d %d\n", vs[0][1], vs[1][0]);  
printf("%p %p\n", &vs, &vs[0][0]);  
printf("%p\n",      &vs[0][1]);  
printf("%p\n",      &vs[1][0]);
```

endereço	id	valor
	VS	

Arrays

```
int xs[3][2] = { {1,2}, {3,4}, {5,6} };  
... f(xs) ...
```

```
void f (int ys[3][2]);
```

```
void f (int ys[][3]);
```

```
void f (int ys[3][]);
```

```
void f (int* ys);
```

```
void f (int** ys);
```

endereço	id	valor
	XS	
	YS	

Arrays

```
void f (int ys[3][2]);  
void f (int ys[][3]);  
void f (int ys[3][]);  
void f (int* ys);  
void f (int** ys);  
  
void f (int* ys) {  
    int L,C = ?,?;  
    for (int i=0; i<L; i++) {  
        for (int j=0; j<C; j++) {  
            ... ys[i][j] ...  
        }  
    }  
}
```

endereço	id	valor
	XS	
	ys	

Exercícios 1 - X

- No site:
- <https://github.com/fsantanna-uerj/LP1/blob/master/Exercicios/lp1-06-arrays.md>

Strings

```
char s1[] = "abc";
```

```
char s2[] = "def";
```

```
printf("%s/%s\n", s1, s2);
```

```
printf("%d\n", strlen(s1));
```

```
printf("%d %d %d %d %d\n",  
      s1[0], s1[1], s1[2], s1[3], s1[4]);
```

endereço	id	valor
	s1	

Strings

```
char s1[] = "abc";
```

```
char s2[4];
```

```
char s2[0] = 'a';
```

```
char s2[1] = 'b';
```

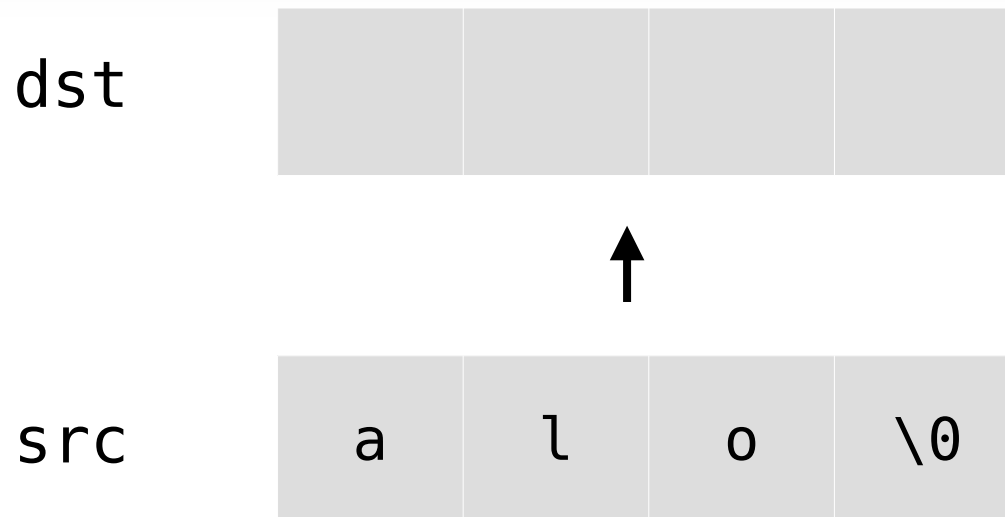
```
char s2[2] = 'c';
```

```
char s2[3] = '\0'; // 0
```

```
printf("%s/%s\n", s1, s2);
```

Strings

- `strncpy`
 - Copia uma string para um destino
 - `char* strncpy (char* dst, char* src, int n)`



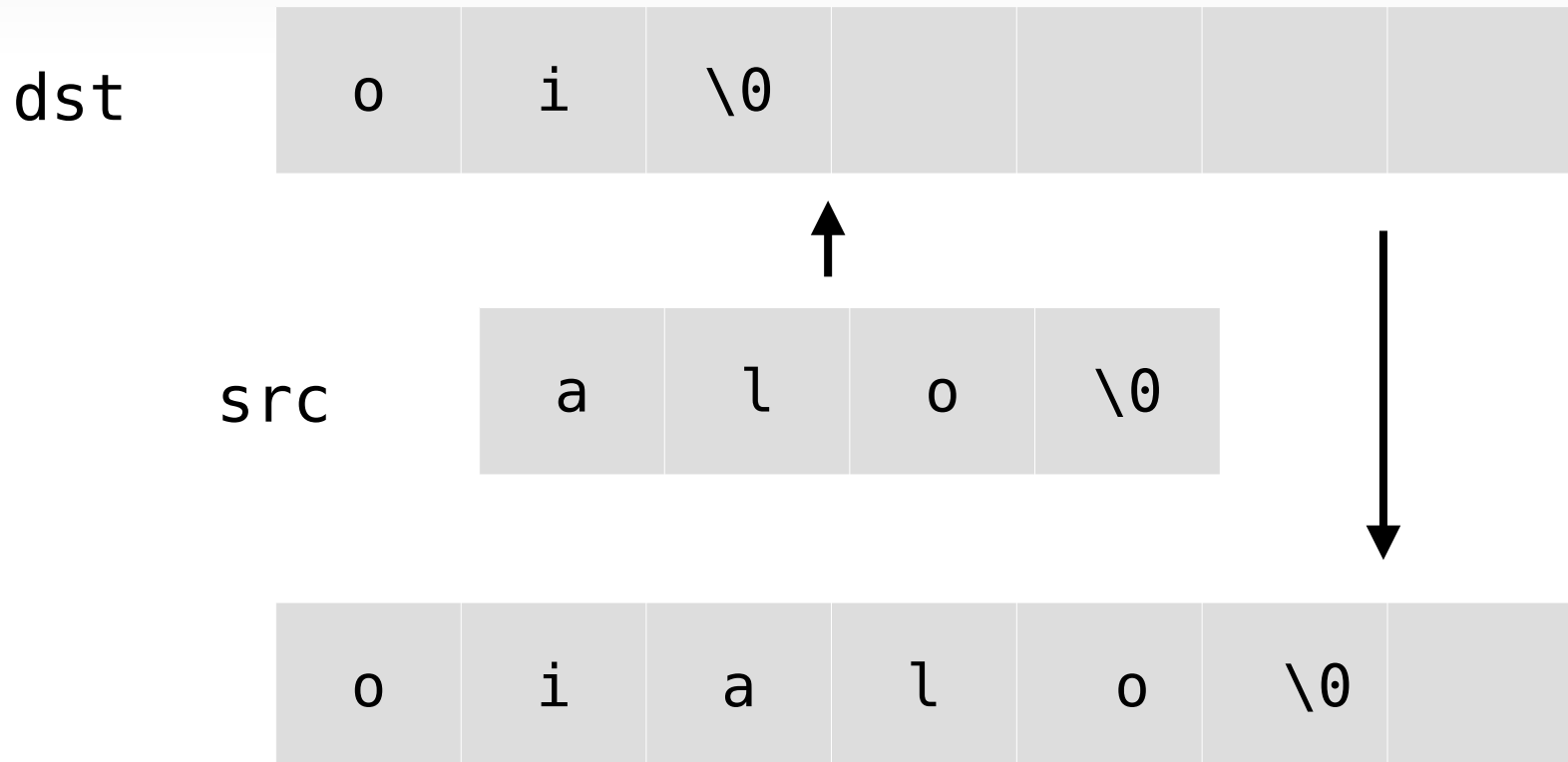
`dst = src; (?)`

03-strncpy.c

Strings

04-strncat.c

- `strncat`
 - Concatena (une) duas strings
 - `char* strncat (char* dst, char* src, int n)`



Exercícios 1 - 2

- No site:
- <https://github.com/fsantanna-uerj/LP1/blob/master/Exercicios/lp1-06-strings.md>