

Aritmética de ponteiros

Aritmética de ponteiros

- Embora ponteiros sejam usados para armazenar endereço, é possível realizar operações sobre eles:
 - Incremento
 - Decremento
 - Comparação

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000		
0x1004		
0x1008		
0x1012		
0x1016		
0x1020		
0x1024		
0x1028		
0x1032		
0x1036		
0x1040		
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

→

```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016		
0x1020		
0x1024		
0x1028		
0x1032		
0x1036		
0x1040		
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024		
0x1028		
0x1032		
0x1036		
0x1040		
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	NULL	p
0x1028		
0x1032		
0x1036		
0x1040	NULL	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	NULL	p
0x1028		
0x1032		
0x1036		
0x1040	NULL	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	NULL	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```


Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	NULL	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

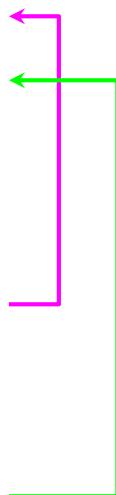


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1004	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

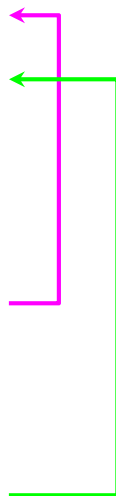


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	10	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1004	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

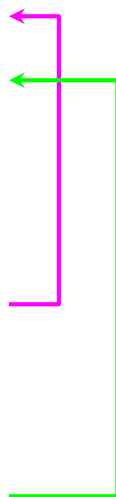


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1004	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

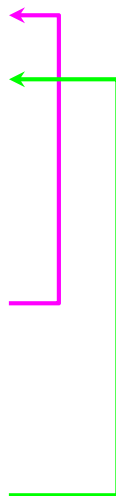


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1004	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

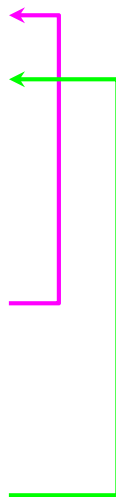


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1004	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

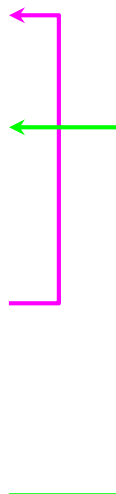


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1008	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

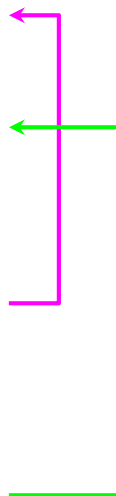


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1000	p
0x1028		
0x1032		
0x1036		
0x1040	0x1008	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

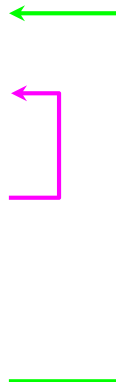


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```


Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1016	p
0x1028		
0x1032		
0x1036		
0x1040	0x1008	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		

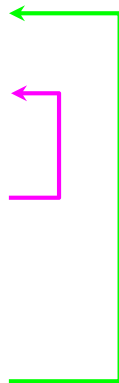


```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```

Aritmética de ponteiros

Endereço Conteúdo Nome

0x1000	5	x
0x1004		
0x1008		
0x1012		
0x1016	20	y
0x1020		
0x1024	0x1016	p
0x1028		
0x1032		
0x1036		
0x1040	0x1008	q
0x1044		
0x1048		
0x1052		
0x1056		
0x1060		



```
int main(void) {  
  
    int x = 10;  
    int y = 20;  
  
    int* p, *q;  
  
    p = &x;  
    q = p + 1;  
  
    *p = *p - 5;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
  
    q++;  
  
    p = q + 2;  
  
    printf("%p\n", p);  
    printf("%i\n", *p);  
    printf("%p\n", q);  
    printf("%i\n", *q);  
  
    return 0;  
}
```