

I117 (4) 文字、文字列処理 (その2)

知念

北陸先端科学技術大学院大学 情報科学研究科
School of Information Science,
Japan Advanced Institute of Science and Technology

左／右詰め

多くの場合はこれで足りる

```
sprintf(msg, "%-15s", "abc");  
printf("|%s|\n", msg);  
sprintf(msg, "%15s", "xyz");  
printf("|%s|\n", msg);
```

結果

```
|abc|  
|          xyz|
```

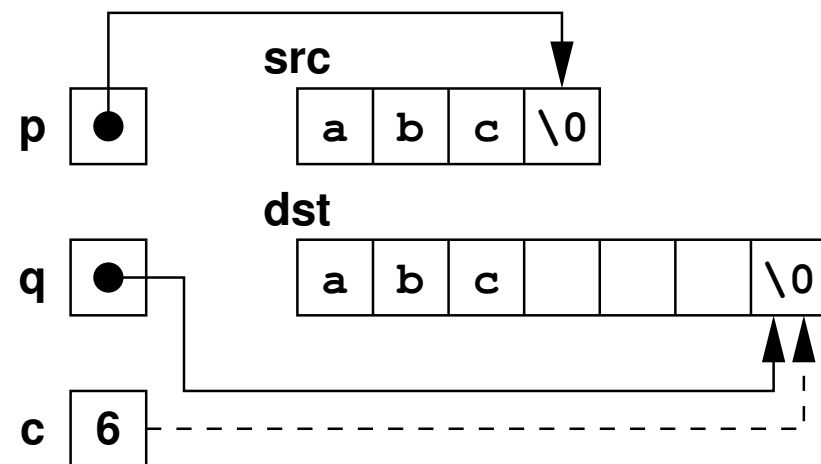
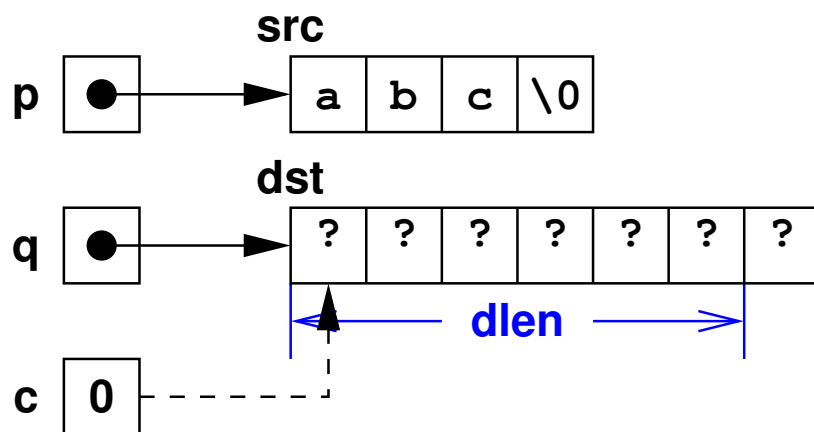
仕組みを考えてみよう

左詰め（その1） — 長さで繰り返す

```
int leftjust(char *dst, int dlen, char *src) {  
    char *p, *q;  
    int c;  
    p = src; q = dst; c = 0;  
    while(c<dlen && *p) {  
        *q++ = *p++; c++;  
    }  
    while(c<dlen) {  
        *q++ = ' '; c++;  
    }  
    *q = '\\0';  
}
```

左詰め（その1） — 長さで繰り返す (*cont.*)

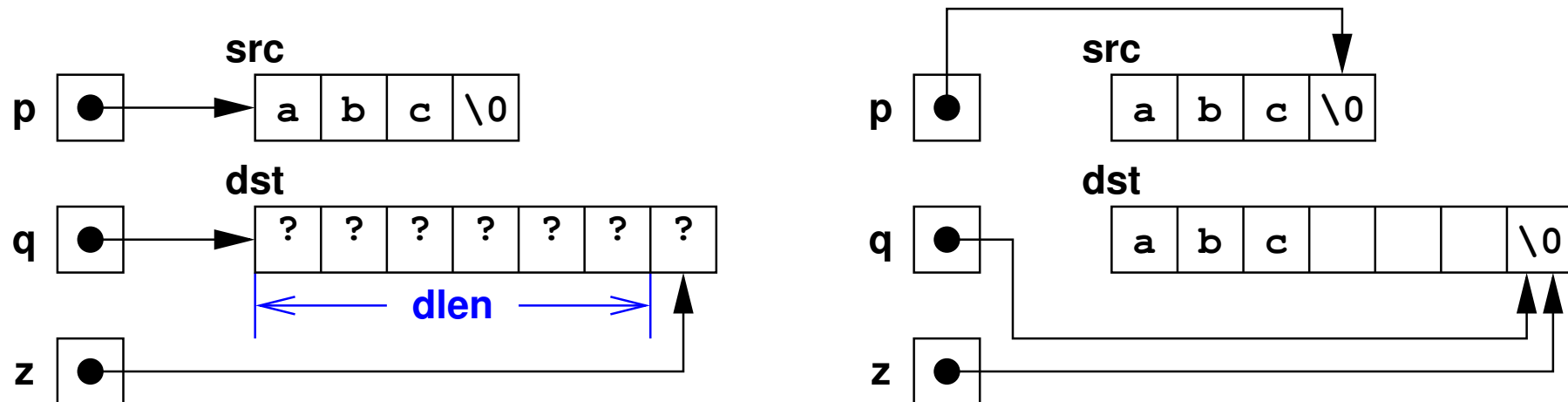
- 長さで繰り返す
 - ◇ 長さを数える
 - ◇ src を複写、残りは ' '(空白) を複写
- dst は $dlen+1$ の長さが必要



左詰め（その2） — 場所で繰り返す

```
int leftjust(char *dst, int dlen, char *src) {  
    char *p, *q, *z;  
    p = src; q = dst; z = dst+dlen;  
    while(q<z && *p) {  
        *q++ = *p++;  
    }  
    while(q<z) {  
        *q++ = ' '  
    }  
    *q = '\\0';  
}
```

左詰め（その2） — 場所で繰り返す (*cont.*)



- 長さではなく場所でループを表現
- ループ中の演算が少ない

コンマ（位どり）

入力	出力	コンマの数
1	1	0
12	12	0
123	123	0
1234	1,234	1
12345	12,345	1
123456	123,456	1
1234567	1,234,567	2

$$\text{コンマの数} = (\text{入力長} - 1) / 3$$

$$\text{出力長} = \text{入力長} + (\text{入力長} - 1) / 3$$

コンマ (位どり) (*cont.*)

```
int addcomma(char *dst, int dlen, char *src)
{
    char *p, *q;
    int iw, ow;
    iw = 0;
    p = src;
    while(*p) {
        p++; iw++;
    }
    ow = iw+(iw-1)/3;
    if(ow>dlen-1)
        return -1;
}
```

コンマ (位どり) (*cont.*)

```
p = src; q = dst;
while(iw>0) {
    if(iw%3==0 && q!=dst) {
        *q++ = ',';
    }
    *q++ = *p++;
    iw--;
}
*q = '\0';

return 0;
}
```