

# String Matching

- **String Matching Problem?**

- 길이  $m$ 의 패턴 문자열  $P$ 가 길이  $n$ 의 텍스트 문자열  $T$  내의 어느 위치에 존재하는가?

T G C A T C **G C A G A G A G** T A T A C A G T A C G

P G C A G A G A G ?

# 단순한 접근

T G C A T C G C A G A G A G T A T A C A G T A C G  
P G C A G A G A G

*n* × *m*

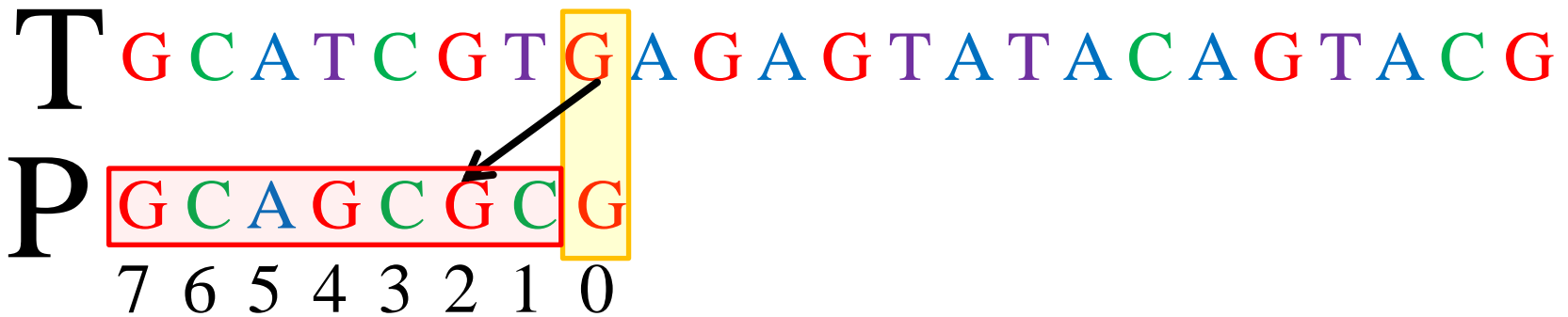
# 한 칸씩 옮겨야 할까?



미리 P내 각 문자들의  
가장 오른쪽 위치를 알아 둔다면?

A	C	G	T
5	1	2	8

# 한 칸씩 옮겨야 할까?



미리 P내 각 문자들의  
가장 오른쪽 위치를 알아 둔다면?

A	C	G	T
5	1	2	8

# Grand Example

T G C A T C G C A G A G A G T A T A C A G T A C G  
P G C A G A G A G

# Grand Example

```
21 i=m-2
22 while i>=0:
23     if d[ord(P[i])-ord('A')]==m:
24         d[ord(P[i])-ord('A')]=m-i-1
25     i-=1
```

P G C A G A G A G  
7 6 5 4 3 2 1 0

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
d	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	1	8	6	8	8	8	2	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8

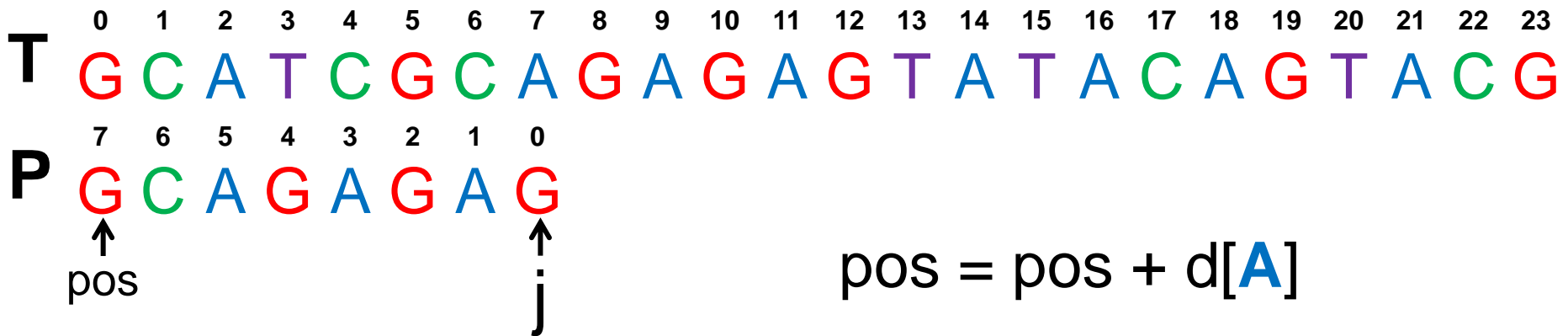
	A	C	G	T
d	1	6	2	8

# Grand Example

```

27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35         pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right

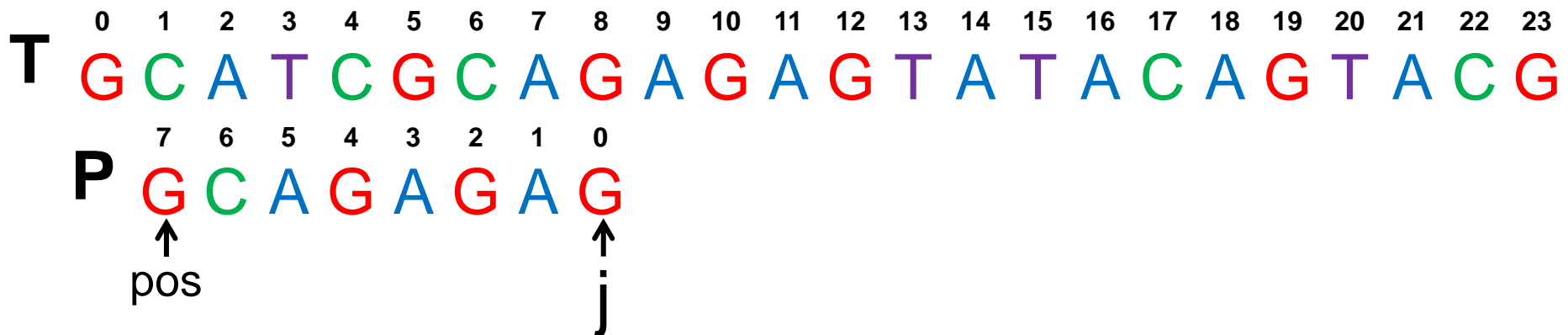
```



	A	C	G	T
d	1	6	2	8

# Grand Example

```
27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35     pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
```



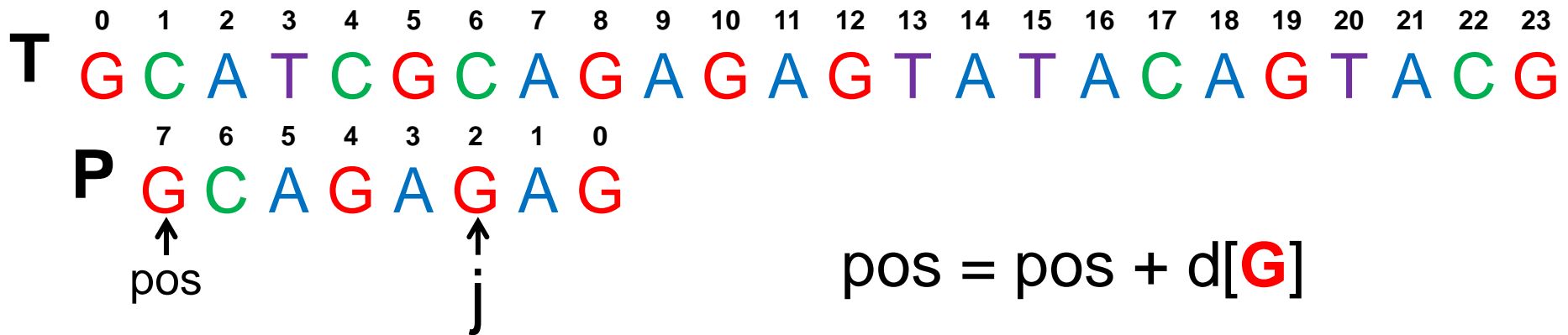
d

A	C	G	T
1	6	2	8



# Grand Example

```
27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35     pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
```



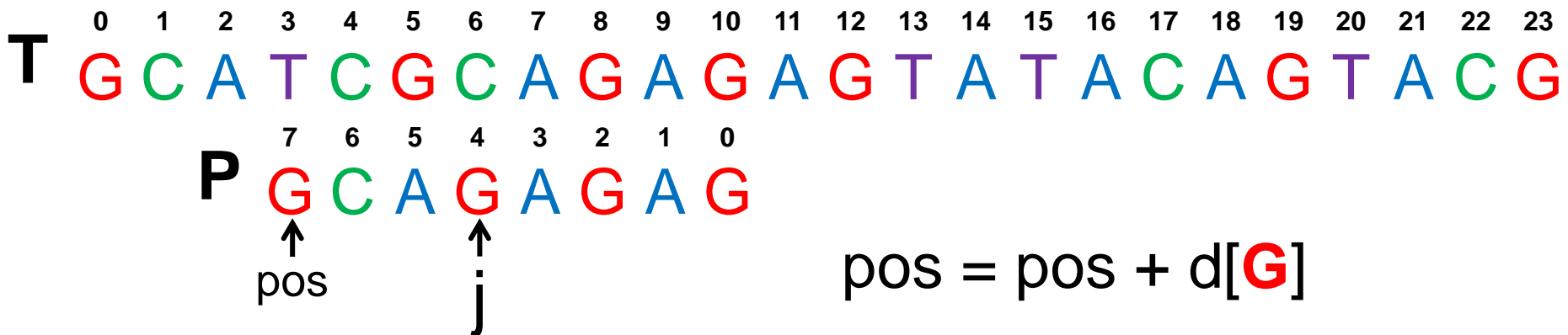
d

A	C	G	T
1	6	2	8

# Grand Example

```

27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j--
33     if j<0: # if found P in T
34         print pos
35     pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
    
```



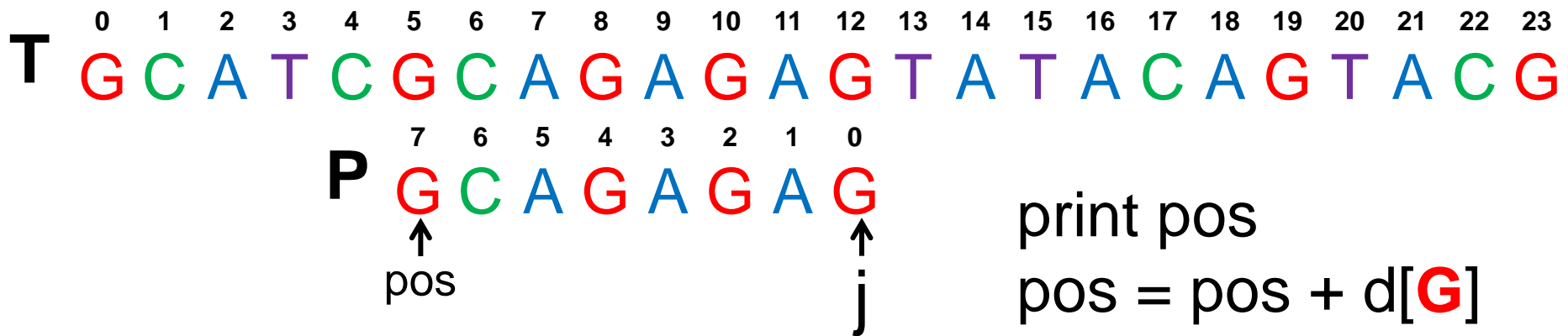
	A	C	G	T
d	1	6	2	8

# Grand Example

```

27 # Searching
28 pos=0 # from 0
29 ▾ while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31 ▾     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33 ▾     if j<0: # if found P in T
34         print pos
35         pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right

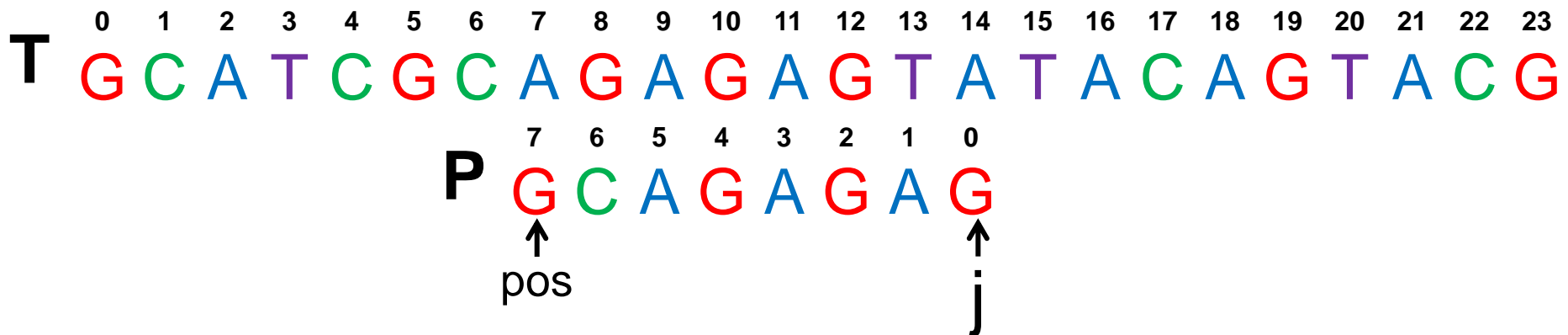
```



<b>d</b>	A	C	G	T
	1	6	2	8

# Grand Example

```
27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35         pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
```

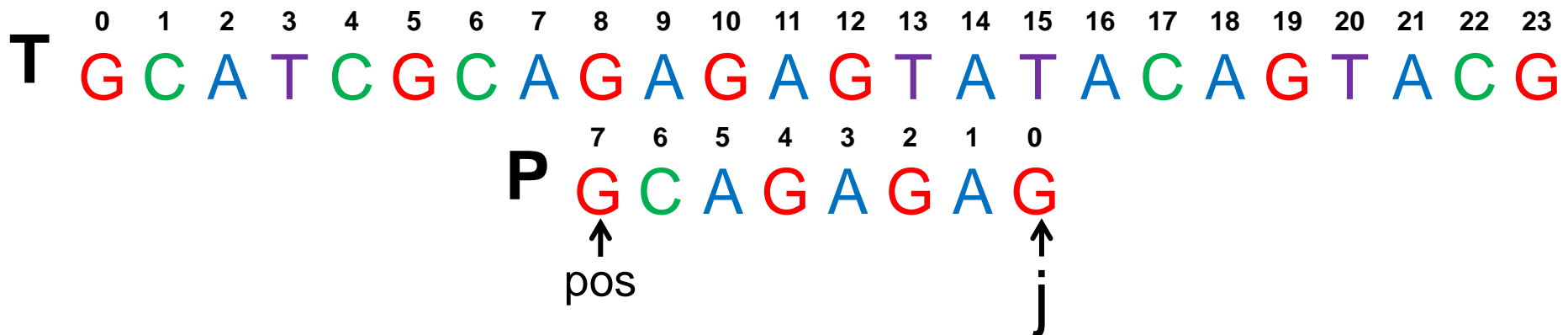


$$\text{pos} = \text{pos} + d[\text{A}]$$

d	A	C	G	T
	1	6	2	8

# Grand Example

```
27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35     pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
```

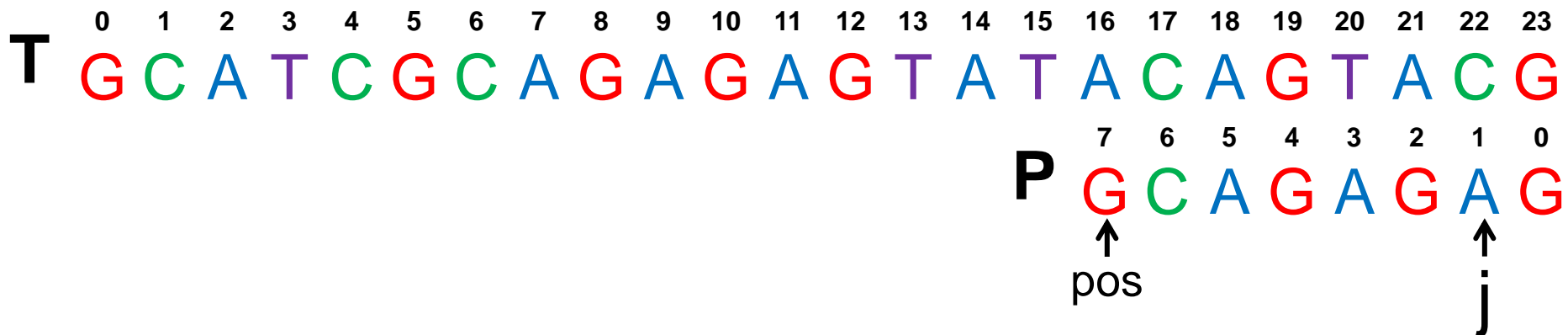


$$\text{pos} = \text{pos} + d[\text{T}]$$

d	A	C	G	T
	1	6	2	8

# Grand Example

```
27 # Searching
28 pos=0 # from 0
29 while pos<=n-m: # to n-m
30     j=m-1 # pattern index
31     while j>=0 and T[pos+j]==P[j]: # match T and P
32         j-=1
33     if j<0: # if found P in T
34         print pos
35     pos+=d[ord(T[pos+m-1])-ord('A')] # shift Pattern to right
```



$$\text{pos} = \text{pos} + d[\text{C}]$$

d	A	C	G	T
	1	6	2	8