

알고리즘 기초 세미나

Week 2

최재혁

200417

공지

- 지난번 세미나 이후에 많은 피드백이 왔습니다.
 - 난이도를 낮추고 문제 풀이 시간을 더 드릴 예정
 - 수도 코드와 개념만 알려 주는 대신 직접 구현하는 과정을 더 추가하겠습니다.
-
- 지난번 영상은 설명이 부족해보여, 오늘 강의 이후에 저번 강의 내용을 추가로 설명 드리고, 그 영상을 바로 올리겠습니다.
 - 링크는 카톡과 github에 따로 공지하겠습니다.
 - github.com/inc5025/AlgorithmSeminar

오늘 수업은?

- ASCII, String
- 지난 번 숙제 피드백

ASCII?

0	<NUL>	32	<SPC>	64	@	96	`	128	Ä	160	†	192	¿	224	‡
1	<SOH>	33	!	65	A	97	a	129	Å	161	°	193	¡	225	·
2	<STX>	34	"	66	B	98	b	130	Ç	162	¢	194	¬	226	,
3	<ETX>	35	#	67	C	99	c	131	É	163	£	195	√	227	„
4	<EOT>	36	\$	68	D	100	d	132	Ñ	164	§	196	ƒ	228	‰
5	<ENQ>	37	%	69	E	101	e	133	Ö	165	•	197	≈	229	Â
6	<ACK>	38	&	70	F	102	f	134	Ü	166	¶	198	Δ	230	Ê
7	<BEL>	39	'	71	G	103	g	135	á	167	ß	199	«	231	Á
8	<BS>	40	(72	H	104	h	136	à	168	®	200	»	232	Ë
9	<TAB>	41)	73	I	105	i	137	â	169	©	201	...	233	È
10	<LF>	42	*	74	J	106	j	138	ä	170	™	202		234	Í
11	<VT>	43	+	75	K	107	k	139	ã	171	'	203	À	235	Î
12	<FF>	44	,	76	L	108	l	140	å	172	..	204	Ã	236	Ï
13	<CR>	45	-	77	M	109	m	141	ç	173	≠	205	Ö	237	Ì
14	<SO>	46	.	78	N	110	n	142	é	174	Æ	206	Œ	238	Ó
15	<SI>	47	/	79	O	111	o	143	è	175	Ø	207	œ	239	Ô
16	<DLE>	48	0	80	P	112	p	144	ê	176	∞	208	-	240	Ⓜ
17	<DC1>	49	1	81	Q	113	q	145	ë	177	±	209	—	241	Ò
18	<DC2>	50	2	82	R	114	r	146	í	178	≤	210	"	242	Ú
19	<DC3>	51	3	83	S	115	s	147	ì	179	≥	211	"	243	Û
20	<DC4>	52	4	84	T	116	t	148	î	180	¥	212	`	244	Ü
21	<NAK>	53	5	85	U	117	u	149	ï	181	μ	213	'	245	ı
22	<SYN>	54	6	86	V	118	v	150	ñ	182	ð	214	÷	246	ˆ
23	<ETB>	55	7	87	W	119	w	151	ó	183	Σ	215	◊	247	˜
24	<CAN>	56	8	88	X	120	x	152	ò	184	Π	216	ÿ	248	—
25		57	9	89	Y	121	y	153	ô	185	π	217	Ÿ	249	˘
26	<SUB>	58	:	90	Z	122	z	154	ö	186	∫	218	/	250	·
27	<ESC>	59	;	91	[123	{	155	õ	187	ª	219	€	251	°
28	<FS>	60	<	92	\	124		156	ú	188	º	220	<	252	´
29	<GS>	61	=	93]	125	}	157	ù	189	Ω	221	>	253	”
30	<RS>	62	>	94	^	126	~	158	û	190	æ	222	fi	254	˘
31	<US>	63	?	95	_	127		159	ü	191	ø	223	fl	255	˘

char2int.cpp

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     char c;
7     cin >> c;
8     cout << (int)c;
9     return 0;
10 }
11
```

char2int.py

```
1 print(ord(input()))  
2 print(chr(int(input())))
```

11654번, 아스키 코드(acmicpc.net/problem/11654)

아스키 코드

성공 분류



5 Bronze V

구현

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	33309	26656	24222	81.853%

문제

알파벳 소문자, 대문자, 숫자 0-9 중 하나가 주어졌을 때, 주어진 글자의 아스키 코드값을 출력하는 프로그램을 작성하시오.

입력

알파벳 소문자, 대문자, 숫자 0-9 중 하나가 첫째 줄에 주어진다.

출력

입력으로 주어진 글자의 아스키 코드 값을 출력한다.

예제 입력 1 복사

```
A
```

예제 입력 2 복사

```
C
```

예제 출력 1 복사

```
65
```

예제 출력 2 복사

```
67
```

15813번, 너의 이름은 몇 점이니?

(acmicpc.net/problem/10809)

너의 이름은 몇 점이니? 원본 출처



Bronze II

문자열

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	128 MB	686	596	564	89.524%

문제

소운이는 성필이에게 단단히 화가 났다. 성필이가 자꾸 소운이의 이름을 놀리는 것이다.

극대노한 소운이는 이름에 대해 많은 검색을 하던 도중 "이름점수"라는 것을 발견하게 된다. 이름 점수란, 알파벳 하나하나에 점수가 있고 이름에 들어가는 모든 알파벳 점수를 합한 것이라고 한다. 예를 들어 이름이 **SUNG PIL** 이라면,

- A = 1점
- B = 2점
- C = 3점
- ...
- Z = 26점

인 점수판에 따라 $S(19)+U(21) + N(14) + G(7) + P(16) + I(9) + L(12) = 98$ 점이다. (즉, 점수는 알파벳 순서이다)

소운이는 **SO YOON**이므로 $S(19) + O(15) + Y(25) + O(15) + N(14) = 103$ 점으로 성필이보다 "이름점수"가 높았다! 그 사실을 알아챈 소운이는 성필이에게 자신이 "이름점수"가 더 높다는 것을 전했고 성필이는 아직 충격에서 헤어나오지 못했다고 한다.

이제 소운이는 사람의 이름을 볼 때 마다 "이름점수"를 계산해본다. 하지만 너무나 많은 사람을 만나기 때문에 계산하기가 귀찮다! 귀찮아진 소운이를 위해 "이름점수"를 계산하는 프로그램을 만들어 주자.

입력

첫 번째 줄에 이름의 길이가 주어진다. (단, 길이는 100자 이하이다)

두 번째 줄에 이름이 띄어쓰기 없이 대문자로 주어진다.

출력

주어진 이름에 대한 "이름점수"를 출력해주자.

예제 입력 1 복사

```
7
SUNGPII
```

예제 출력 1 복사

```
98
```


어떻게 풀어야 할까?

- 길이, 합, 글자 변수를 선언한다.
- 길이를 입력 받는다
 - 한글자를 입력 받는다
 - 합에 글자에 해당하는 점수를 더한다
 - 이를 길이 만큼 반복한다
- 합을 출력한다.

풀이 (15813.cpp)

```
#include <iostream>
using namespace std;

int main()
{
    int n, sum = 0;
    char c;

    cin >> n;
    for(int i = 0; i < n; i++){
        cin >> c;
        sum += (int)c - 'A' + 1;
    }

    cout << sum;

    return 0;
}
```

String input in python

```
A = input()  
# python
```

String input in Cpp (egloos.zum.com/hanjw8410/v/4056009)

```
#include <iostream>
#include <string>
using namespace std;

const int LENGTH = 100;

int main()
{
    char c[LENGTH];
    cin >> c;

    string s1;
    cin >> s1;

    string s2;
    getline(cin, s2);

    return 0;
}
```

```

#include <iostream>
#include <string>
using namespace std;

int main()
{
    string s;
    cin >> s;

    for(int i = 0; i < s.size(); i++){
        cout << i << " : " << s[i] << "\n";
    }

    return 0;
}

```

index	0	1	2	3	4	5	6	7	8	9	10	11
Value	H	e	l	l	o	,	W	o	r	l	d	!

10809번, 알파벳 찾기 (acmicpc.net/problem/10809)

알파벳 찾기 성공 분류



Bronze II

구현

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	45800	24677	21137	55.505%

문제

알파벳 소문자로만 이루어진 단어 S가 주어진다. 각각의 알파벳에 대해서, 단어에 포함되어 있는 경우에는 처음 등장하는 위치를, 포함되어 있지 않은 경우에는 -1을 출력하는 프로그램을 작성하시오.

입력

첫째 줄에 단어 S가 주어진다. 단어의 길이는 100을 넘지 않으며, 알파벳 소문자로만 이루어져 있다.

출력

각각의 알파벳에 대해서, a가 처음 등장하는 위치, b가 처음 등장하는 위치, ... z가 처음 등장하는 위치를 공백으로 구분해서 출력한다.

만약, 어떤 알파벳이 단어에 포함되어 있지 않다면 -1을 출력한다. 단어의 첫 번째 글자는 0번째 위치이고, 두 번째 글자는 1번째 위치이다.

예제 입력 1 복사

```
baekjoon
```

예제 출력 1 복사

```
1 0 -1 -1 2 -1 -1 -1 -1 4 3 -1 -1 7 5 -1 -1 -1 -1 -1 -1 -1 -1 -1
```

어떻게 풀어야 할까?

- String, int[26] 배열 선언
- 배열을 -1로 초기화
- String 입력 받음
 - 한 글자를 봄
 - 그 글자에 해당하는 배열의 값이 -1이면 배열에 현재 인덱스 저장
 - 아니면 넘어 감
- 배열의 값을 모두 출력함

풀이 (10809.cpp)

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    int n[26];
    string s;
    cin >> s;

    for(int i = 0; i < 26; i++)
        n[i] = -1;

    for(int i = 0; i < s.size(); i++){
        int idx = s[i] - 'a';
        if(n[idx] == -1)
            n[idx] = i;
    }

    for(int i = 0; i < 26; i++)
        cout << n[i] << " ";

    return 0;
}
```


1871번, 좋은 자동차 번호판 (acmicpc.net/problem/10809)

좋은 자동차 번호판

성공

출처

다국어

☆

한국어 ▾

2 Bronze II

구현 문자열

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	128 MB	1555	1008	925	65.930%

문제

앨버타의 자동차 번호판은 ABC-0123 (세 글자, 네 숫자)와 같이 두 부분으로 나누어져 있다.

좋은 번호판은 첫 번째 부분의 가치와 두 번째 부분의 가치의 차이가 100을 넘지 않는 번호판이다.

글자로 이루어진 첫 번째 부분의 가치는 글자를 26진법 수처럼 계산한다. (각 자리가 [A..Z]) 예를 들어, "ABC"의 가치는 28 ($0 \times 26^2 + 1 \times 26^1 + 2 \times 26^0$)이 된다. "ABC-0123"은 $|28 - 123| \leq 100$ 이기 때문에, 좋은 번호판이다.

자동차 번호판이 주어졌을 때, 좋은 번호판인지 아닌지를 구하는 프로그램을 작성하시오.

입력

첫째 줄에 번호판의 수 N ($1 \leq N \leq 100$)이 주어진다. 다음 N 개 줄에는 자동차 번호판이 LLL-DDDD 형식으로 주어진다.

출력

각각의 자동차 번호판에 대해서, 좋은 번호판이면 "nice"를, 아니면 "not nice"를 출력한다.

예제 입력 1 복사

```
2
ABC-0123
AAA-9999
```

예제 출력 1 복사

```
nice
not nice
```

어떻게 풀어야 할까?

- 번호판의 수를 입력 받음
 - 번호판을 입력 받음
 - (알파벳 점수) = $(s[0] - 'A') * 26 * 26 + (s[1] - 'A') * 26 + s[2] - 'A'$
 - (숫자 점수) = $(s[4] - '0') * 1000 + (s[5] - '0') * 100 + (s[6] - '0') * 10 + s[7] - '0'$
- 알파벳 점수와 숫자 점수의 차이가 100 이하이면 nice 출력
- 아니라면 not nice 출력

풀이 (1871.cpp)

```
#include <iostream>
#include <string>
using namespace std;

int abs(int a)
{
    if (a > 0)
        return a;

    return -a;
}

int main()
{
    int n, ls, rs;
    string s;

    cin >> n;

    for (int i = 0; i < n; i++)
    {
        cin >> s;
        ls = (s[0] - 'A') * 26 * 26 + (s[1] - 'A') * 26 + s[2] - 'A';
        rs = (s[4] - '0') * 1000 + (s[5] - '0') * 100 + (s[6] - '0') * 10 + s[7] - '0';
        if (abs(ls - rs) <= 100)
            cout << "nice\n";
        else
            cout << "not nice\n";
    }

    return 0;
}
```

2675번, 문자열 반복 (acmicpc.net/problem/2675)

문자열 반복 성공 출처 다국어

☆ 한국어 ▾

2 Bronze II

구현

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	128 MB	39472	20542	18241	53.826%

문제

문자열 S 를 입력받은 후에, 각 문자를 R 번 반복해 새 문자열 P 를 만든 후 출력하는 프로그램을 작성하시오. 즉, 첫 번째 문자를 R 번 반복하고, 두 번째 문자를 R 번 반복하는 식으로 P 를 만들면 된다. S 에는 QR Code "alphanumeric" 문자만 들어있다.

QR Code "alphanumeric" 문자는 0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ\\$_%*+-./: 이다.

입력

첫째 줄에 테스트 케이스의 개수 T ($1 \leq T \leq 1,000$)가 주어진다. 각 테스트 케이스는 반복 횟수 R ($1 \leq R \leq 8$), 문자열 S 가 공백으로 구분되어 주어진다. S 의 길이는 적어도 1이며, 20글자를 넘지 않는다.

출력

각 테스트 케이스에 대해 P 를 출력한다.

예제 입력 1 복사

```
2
3 ABC
5 /HTP
```

예제 출력 1 복사

```
AAABBBCCCC
/////HHHHHTTTTTPPPPP
```

1157번, 단어 공부(acmicpc.net/problem/1157)

단어 공부 성공 분류



1 Bronze I

메드 록 구현

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
2 초	128 MB	56782	21434	17369	38.360%

문제

알파벳 대소문자로 된 단어가 주어지면, 이 단어에서 가장 많이 사용된 알파벳이 무엇인지 알아내는 프로그램을 작성하시오. 단, 대문자와 소문자를 구분하지 않는다.

입력

첫째 줄에 알파벳 대소문자로 이루어진 단어가 주어진다. 주어지는 단어의 길이는 1,000,000을 넘지 않는다.

출력

첫째 줄에 이 단어에서 가장 많이 사용된 알파벳을 대문자로 출력한다. 단, 가장 많이 사용된 알파벳이 여러 개 존재하는 경우에는 ?를 출력한다.

예제 입력 1 복사

Mississippi

예제 출력 1 복사

?

예제 입력 2 복사

zZa

예제 출력 2 복사

Z

QnA