

Pembahasan Soal

Olimpiade Sains Nasional 2004 Bidang Informatika

Pekanbaru, 23-27 Agustus 2004

Soal 1. Membuka String Terlipat

Penulis: *Ilham Kurnia*

Nama Program: LIPAT2.PAS / C / CPP

Batas Run-time: 0.1 detik / test-case

Batas Memori: 16 MB

Nama Berkas Masukan: Standard Input

Nama Berkas Keluaran: Standard Output

Akhir-akhir ini Pak Dengklek suka bermain-main dengan string terlipat. Apa yang dimaksud dengan string terlipat? String terlipat adalah salah satu cara ringkas untuk menampilkan sebuah string. Mari kita lihat contohnya terlebih dahulu. String AAAAAAAAAABABABCCD dapat kita lipat sehingga 9(A)3(AB)CCD. Ini artinya bahwa string sesungguhnya berisi A diulang 9 kali, dilanjutkan dengan AB yang diulang 3 kali dan diakhiri dengan CCD.

Berikut ini adalah definisi formal dari string terlipat.

- ? Sebuah string yang terdiri dari sebuah karakter 'A'..'Z' dianggap sebagai string yang terlipat (jadi "A" adalah sebuah string yang terlipat). Bila lipatan dari string ini dibuka maka akan dihasilkan string yang sama yaitu sebuah karakter tadi.
- ? Jika S dan Q adalah string yang terlipat (S dan Q di sini adalah variabel), maka SQ juga adalah string yang terlipat. Jika lipatan S dibuka menjadi S' dan Q dibuka menjadi Q', maka SQ dibuka menjadi S'Q'.
- ? Jika S adalah string terlipat, maka X(S) juga adalah string terlipat, di mana X adalah bilangan bulat positif (dan X tidak akan diawali oleh angka 0). Jika S dibuka menjadi S', maka X(S) dibuka menjadi S' yang diulang sebanyak X kali.

Pak Dengklek terkadang lupa dengan bentuk asli dari string terlipat. Oleh karena itu, ia meminta bantuan kalian untuk membuka lipatan dari string yang terlipat sehingga diperoleh string aslinya.

FORMAT MASUKAN

Masukan terdiri dari hanya satu baris saja. Baris tersebut berisi 1..100 karakter yang setiap karakternya termasuk dalam himpunan { 'A', 'B', ..., 'Z', '0', '1', ..., '9', '(', ') ' }. String masukan adalah string terlipat yang benar (formatnya sesuai dengan definisi di atas).

CONTOH MASUKAN

```
9(A)3(AB)CCD
```

FORMAT KELUARAN

Keluaran terdiri dari hanya satu baris saja yang berisi string yang sudah dibuka lipatnya. String yang sudah dibuka lipatnya tidak akan berisi lebih dari 100 karakter.

CONTOH KELUARAN

```
AAAAAAAAAABABABCCD
```

CONTOH MASUKAN 2

```
2(TOKI3(YES))
```

CONTOH KELUARAN 2

```
TOKIYESYESYESTOKIYESYESYES
```

Peringatan

Jangan lupa untuk **tidak** menambahkan spasi di awal dan akhir.

Pembahasan

Cara termudah untuk menyelesaikan soal ini adalah dengan menggunakan rekursi. Algoritmanya adalah sebagai berikut:

```
1   posisi := 1
2
3   fungsi rekursi : string
4   tmp := '';
5   while (posisi = panjang(input) dan input[posisi] ? `)`
6       jika input[posisi] adalah
7           0..9: jika angka belum diinisialisasi
8               angka := input[posisi]
9               jika sudah, angka := angka * 10 + input[posisi]
10          `(`: { tambahkan tmp sebanyak angka * hasil rekursi }
11              tmp := tmp + (rekursi)angka
12              angka := belum diinisialisasi
13          A..Z: tmp := tmp + input[posisi]
14          end;
15          posisi := posisi + 1;
16  rekursi := tmp;
```

Kasus dasar dari fungsi rekursi adalah jika kita telah menemukan karakter ')' atau ketika input telah habis diiterasi. Kasus rekursifnya adalah ketika kita menemukan karakter '('. Perlu diingat, selalu ada bilangan bulat positif yang mendahului '('. Berhubung keluaran bisa terdiri dari 100 karakter, maka bisa jadi bilangan bulat positif ini memiliki lebih dari satu digit.

Dalam soal ini, disediakan 20 test case dengan rincian sebagai berikut:

N o	Deskripsi
1	Sebuah karakter 'A'
2	Dua buah karakter 'B'
3	Lima buah karakter 'C'
4	Seratus buah karakter 'D'
5	1(E)
6	9(F)
7	27(G)
8	100(H)
9	1(IJK)
10	5(IJK)
11	33(IJK)
12	1(2(3(4(L)M)N)O)
13	1(L2(M3(N4(O))))
14	Acak
15	Acak
16	Acak
17	Acak
18	Acak
19	A1(1(1(1(1...1(A1(...1(A))...))
20	1(1(1(1(2(1(1(1(1(2...1(1(1(X))...))

Berikut adalah source code dari solusi di atas:

```

program MembukaStringTerlipat;
const InFile = 'LIPAT2.IN';
      OutFile = 'LIPAT2.OUT';

var s : string;
    i : integer;

function cariString() : string;
var tmp, h : string;
    num, j : integer;

begin
tmp := ''; num := -1;
while (i <= length(s)) and (s[i] <> ')') do

```

```
begin
case s[i] of
'0'..'9': if (num = -1) then num := ord(s[i]) - ord('0')
           else num := num * 10 + (ord(s[i]) - ord('0'));
'('      :
begin
inc(i);
h := cariString();
for j := 1 to num do tmp := tmp + h;
num := -1;
end;
else tmp := tmp + s[i];
end;
inc(i);
end;
cariString := tmp;
end;

begin
assign(input, InFile);
assign(output, OutFile);
reset(input);
rewrite(output);

i := 1;
readln(s);
writeln(cariString);

close(input);
close(output);
end.
```

Soal 2. Permutasi

Penulis: *Ilham Kurnia*

Nama Program: PERM2.PAS / C / CPP
Batas Run-time: 1 detik / test-case
Batas Memori: 16 MB
Nama Berkas Masukan: Standard Input
Nama Berkas Keluaran: Standard Output

Sejumlah N ($1 \leq N \leq 1000000$) bebek Pak Dengklek sedang bermain dengan antrian mereka. Setiap bebek dinomori $1..N$. Pada awalnya, mereka antri dengan suatu urutan yang diberikan oleh sebuah array A . Jadi, untuk bebek ke- i , dia antri di urutan ke- $A[i]$. Urutan bebek bebek dinomori dari 1 sampai dengan N .

Bosan dengan antrian lamanya, bebek-bebek tersebut ingin mendapatkan urutan baru. Supaya adil, mereka membuat aturan tertentu. Misalkan urutan yang baru disimpan dalam sebuah array B . Aturannya adalah sebagai berikut: array B adalah sedemikian sehingga untuk setiap bebek, $B[A[i]] = i$, ($1 \leq i \leq N$).

Beberapa saat lamanya mereka berada pada antrian yang baru. Kemudian, Pak Dengklek datang dan terkejut melihat bebek-bebeknya berada pada antrian yang baru dan meminta mereka untuk kembali ke antriannya yang lama. Bebek-bebek tersebut lupa. Pak Dengklek meminta bantuan Anda untuk mengingatkan bebek-bebek tersebut akan antriannya yang lama.

FORMAT MASUKAN

Baris pertama dari masukan berisi sebuah bilangan bulat N . Setiap baris pada N baris berikut berisi tepat sebuah bilangan bulat $B[1]$, $B[2]$, ..., $B[N]$.

CONTOH MASUKAN

```
4
3
4
1
2
```

FORMAT KELUARAN

Jika tidak ada array A yang dapat dibuat dari array B , keluarkan angka 0. Jika tidak, keluarkan N baris, di mana di setiap barisnya terdapat sebuah bilangan bulat yang bernilai $A[i]$.

CONTOH KELUARAN

```

3
4
1
2

```

Pembahasan

Soal ini merupakan soal dengan solusi terpendek. Yang kita perlu lakukan adalah melakukan sedikit manipulasi terhadap $B[A[i]] = i$ dan menerjemahkan manipulasi tersebut ke dalam program. Pertama-tama, perlu diingat bahwa i adalah sebuah variabel, sehingga nilainya bisa berubah-ubah. Misal $x = B[j]$. Satu-satunya kemungkinan yang dapat memenuhi syarat $B[A[i]] = i$ adalah jika $j = A[x]$. Ini membuat permasalahannya menjadi $x = B[A[x]]$, yang tidak lain adalah syarat yang ditentukan.

Dalam soal ini, disediakan 10 test case dengan rincian sebagai berikut:

No	N
1	1
2	10
3	100
4	1000
5	10000
6	50000
7	100000
8	200000
9	500000
10	1000000

Berikut adalah source code dari solusi di atas:

```

program Permutasi;

const MAXN      = 1000000;
      InFile    = 'PERM2.IN';
      OutFile   = 'PERM2.OUT';

var a : array[1..MAXN] of longint;
    i, b, n : longint;

begin
assign(input, InFile);
assign(output, OutFile);
reset(input);
rewrite(output);

```

```
read(n);
for i := 1 to n do
  begin
    read(b);
    a[b] := i;
  end;

for i := 1 to n do writeln(a[i]);

close(input);
close(output);
end.
```