Зоран Васиљевић

ОЛИМПИЈСКИ ЗАДАТАК

Једно добро решење из информатичке радионице

На дванаестој Информатичкој олимпијади одржаној у Кини 2000-те године првог такмичарског дана био је постављен овакав задатак:

Палиндром је симетричан стринг, тј. стринг који се чита идентично слева надесно и здесна налево. Написати програм који за дати стринг налази најмањи број знакова које у њега треба уметнути да се добије палиндром.

На пример, уметањем 2 знака, стринг 'Ab3bd' може се трансформисати у палиндром ('dAb3bAd' или 'Adb3bdA'). Међутим, уметањем мање од два знака не може се добити палиндром.

Улаз: Име улазног фајла је PALIN.IN. Први ред садржи један цео број: дужину улазног стринга N, $3 \leqslant N \leqslant 5000$. Други ред садржи један стринг дужине N. Стринг се састоји од великих слова од 'A' до 'Z', малих слова од 'a' до 'z' и цифара од '0' до '9'. Мала и велика слова треба сматрати различитим.

Излаз: Име излазног фајла треба да буде PALIN.OUT, у коме први ред садржи један број, који представља тражени најмањи број уметања.

Пример улаза и излаза:

PALIN.IN	PALIN.OUT
5	2
Ab3bd	

Упутство за решавање

Добар пут за решавање задатог проблема јесте проналажење најдужег палиндрома неузастопних симбола задатог стринга. Очигледно је да кад се пронађе одговарајући палиндром (нека је његова дужина L), потребан број симбола које треба уметнути у стринг са улаза да би он постао палиндром биће једнак $N - L$. Заправо кад се пронађе најдужи палиндром неузастопних симбола потребно је уочити средину тог палиндрома и одговарајуће симболе симетрично у односу на ту средину уметати у дати стринг и на тај начин ће се постићи да добијени стринг

Овај чланак преносимо из часописа „Тангента“. Уредништво је оценило да он може бити интересантан за проширени круг читалаца.

постане палиндром са минималним бројем уметања. Дакле, проблем се своди на одређивање дужине најдужег палиндрома неузастопних симбола, што се решава динамичким програмирањем.

Званично решење

Кључни део решења јесте матрица B која се формира тако да су индекси врста и колона симболи унетог стринга, с тим да су симболи у индексима колона у обрнутом редоследу. Вредност члана матрице $B[i,j]$ представља број 'преклапајућих' симбола подстринга од прве до i-те позиције и подстринга од N-те (у овом случају пете) до j-те позиције, где је $j = i+1, N$ (према матрици упоређују се стринг састављен од првог до i-тог симбола који одговарају индексима врста и стринг састављен од 1. до $(N-i)$-тог симбола који одговарају индексима колона). Очигледно је да је $1 \leqslant i \leqslant N-1$, као и да елементи на и испод споредне дијагонале нису потребни. Матрица се попуњава врсту по врсту одозго на доле. После попуњавања i-те врсте попуњава се $(i+1)$-ва, и то од прве колоне па до првог елемента изнад споредне дијагонале, тј. с лева на десно. На примеру из поставке задатка матрица B би изгледала овако:

B	1 (d)	2 (b)	3 (3)	4 (b)	5 (A)
1 (A)	0	0	0	0	
2 (b)	0	1	1		
3 (3)	0	1			
4 (b)	0				
5 (d)					

Одређивање вредности елемента $B[i,j]$ се врши према следећем правилу:

$$B[i,j] = \begin{cases} \max\{\max\{B[i,j-1], B[i-1,j]\}, B[i-1,j-1]+1\}, & a[i] = a[j] \\ \max\{B[i,j-1], B[i+1,j]\}, & a[i] \neq a[j] \end{cases}$$

Дужина траженог палиндрома се не налази међу елементима матрице, већ се одређује током попуњавања матрице тако што се након сваке израчунате врсте одређује текућа максимална дужина на следећи начин:

```
duzina:=max(B[i,N-1-i]*2+1,max(B[i,N-i]*2,duzina));
```

Након формирања матрице резултат је $N - \texttt{duzina}$. Пошто је за попуњавање једне врсте матрице потребна само претходна, онда су за програмерску реализацију овог задатка потребна само два низа. Кључна функција којом се налази дужина најдужег палиндрома неузастопних симбола у решењу које је дала комисија изгледа овако:

```
Function MaxPub(const a,reva:ts):integer;
  Var i,j,k,now,answer:integer;
Begin
fillchar(g[0]^,sizeof(g[0]^),0);
  answer:=0; k:=len;
  for i:=1 to k-1 do
    begin
      g[i]^[0]:=0;
      for j:=1 to k-i do
        begin
          now:=max(g[i-1]^[j],g[i]^[j-1]);
          if a[i]=reva[j] then
            now:=max(now,g[i-1]^[j-1]+1);
            g[i]^[j]:=now;
          end;
         answer:=max(g[i]^[k-i]*2,answer);
         answer:=max(g[i]^[k-1-i]*2+1,answer);
        end;
       MaxPub:=answer;
      End;
```

Аргумент a функције MaxPub је унети стринг, reva његова „обрнута верзија“, len дужина унетог стринга, answer текућа максимална дужина, g је низ показивача на низ целих бројева и представља i-ту врсту матрице B, а иницијализован је тако да је $g[2s] = g[0]$, а $g[2s+1] = g[1]$, што је изведено следећим командама:

```
new(g[0]); new(g[1]); for i:=2 to maxs do g[i]:=g[i mod 2];
```

где је maxs 5000. Очигледно је да су заиста у употреби само два низа.

Мало другачије решење

Одређивање траженог броја би могло да тече и другачије. Нека је a улазни стринг, а N његова дужина. Као и у званичном решењу, на почетку се формира матрица B, али на другачији начин. Индекси врста и колона су симболи унетог стринга. Вредност члана матрице $B[i,j]$ представља дужину најдужег палиндрома неузастопних елемената стринга од i-тог до j-тог члана задатог стринга. У решавању задатка довољно је користити само 'горњу половину' матрице, па се зато испод главне дијагонале могу уписати нуле. На главној дијагонали се налазе јединице зато што је дужина најдужег палиндрома од i-тог до i-тог члана стринга увек један.

Матрица се попуњава колону по колону одоздо на горе. После попуњавања j-те колоне попуњава се $(j+1)$-ва, и то од првог поља изнад главне дијагонале па до почетног поља колоне, тј. од елемента $B[j-1,i]$ до $B[1,j]$.

Поље $B[i,j]$ добија вредност $B[i,j] = B[i-1,j-1]+2$ ако је $a[i] = a[j]$, зато што је дужина најдужег палиндрома од $(i-1)$-ог до $(j-1)$-ог симбола у стрингу (низу $a[i]$, $i = 1,N$) била $B[i-1,j-1]$ и сад је треба повећати за 2, иначе $B[i,j] = \max\{B[i,j-1], B[i+1,j]\}$, односно дужем палиндрому од i-тог до $(j-1)$-ог симбола у стрингу или од $(i+1)$-ог до j-тог.

$$B[i,j] = \begin{cases} B[i-1,j-1]+2, & a[i] = a[j] \\ \max\{B[i,j-1], B[i+1,j]\}, & a[i] \neq a[j] \end{cases}$$

Попуњена матрица B би изгледала овако:

B	1 (A)	2 (b)	3 (3)	4 (b)	5 (d)
1 (A)	1	1	1	3	3
2 (b)	0	1	1	3	3
3 (3)		0	1	1	1
4 (b)			0	1	1
5 (d)				0	1

Дужина најдужег палиндрома биће једнака члану матрице са индексима 1, N. Пошто је потребан само овај број, а за попуњавање j-те колоне се користи само $(j-1)$-ва колона, онда је и овде довољно користити само два низа.

На пример, ако се попуњава j-та колона за њу се користи један низ (низ $b2$) и један низ за претходну $(j-1)$-ву колону (низ $b1$), након попуњавања j-те колоне низ $b1$ ће добити вредности низа $b2$. Према описаном поступку кључна процедура изгледа овако:

```
Procedure Uradi;
Var i,j:integer;
Begin
  b1[-1]:=0; b1[0]:=1; b2[-1]:=0; b2[0]:=1;
  for i:=2 to n do
    begin
      for j:=1 to i-1 do
        if a[i]=a[i-j] then b2[j]:=b1[j-2]+2
        else b2[j]:=Max(b2[j-1],b1[j-1]);
    b1:=b2;
  end;
End;
```