

Michal Korbela
septima
Gym. J.J. Bánovce nad Bebravou
úloha 8

Ja som to spravil takto:

Nakódil som si písmenkový strom a doňho som uložil všetky slová a potom pre každé políčko v reťazci (300 000) som si zistil či existuje slovo z písmenkového stromu. Ak existuje, tak si nastavím koniec pokiaľ mi tento reťazec siaha. Ak však políčko ktoré testujem neobsahuje žiadny reťazec a dané políčko sa nenachádza v už obsiahnutom intervale, tak toto písmeno už neviem nijako otlačiť. Preto ho zarátam.

Tomuto však nevyhovuje vstup 300000x'a' v reťazci a jeden z pečiatok stačí byť 4999x'a' a na konci b.

Tuto mi to nezbehne. Avšak keby som si do písmenkového stromu nahádzal pečiatky od konca a aj testoval 300000 reťazec od konca, tak potom to v momente zbehne – natrafi na nezhodu hneď na začiatku reťazca.

Preto síce môj algoritmus nieje najrýchlejší(keďže nezbehne na testovači), ale zato hádže vždy správny výsledok.

Popis štruktúry písmenkového stromu:

```
int T[4000000][27]={0};           // miesto kde sú uložené vrcholy

int nextp=2;                       //pointer na nasledujúce voľné miesto v strome

int convert(char c){               // funkcia, ktorá zakóduje znak, aby jeho hodnota bola od 0-26
return (int)c-97;                 //jednoducho
}

void add(char * s){               // funkcia na pridanie reťazca do stromu

int len=strlen(s);               // vypočítam si dĺžku reťazca
int pos=1;                      // nastavím si pointer na začiatok stromu
for(int j=0; j<len; j++){        // a zapisujem jednotlivé písmená
if(T[pos][convert(s[j])]>0)      // aj dané písmeno už existuje v strome
{
pos=T[pos][convert(s[j])];      // nastavím si pozíciu naň a idem na ďalšie písmeno slova
}
else{                          // inak si nastavím pointer na ďalšie písmeno na ďalšie voľné miesto
T[pos][convert(s[j])]=nextp;
pos=nextp;                    // presuniem sa na to miesto v strome
nextp++;                     // voľné miesto v strome bude ďalšie nasledujúce voľné miesto
}
```

```

}
}
T[pos][26]=1;           // na koniec slova pridáme do kódu 26 hodnotu 1, čo znamená,
že tu slovo končí
}

```

```

int controll(int pos){           //skontroluje a vráti vajväčší počet písmen ktoré pokryje slovo,
    ktoré sa začína na pozícii pos

```

```

    int pozicia=pos,tpos=1;           //aktuálne pokrývame 0 znakov
    int naj=0;

```

```

    while(true){           // pokiaľ máme nejaké slovo, ktoré je zhodné s reťazcom
        if(pos>slength) break;           // s length je dĺžka 300 000 reťazca, a ak by sme vyšli von, tak
        také slovo neexistuje
        if(T[tpos][26]==1) naj=pos-pozicia;           // ak sme narazili na koniec reťazca, tak potom si
        uložíme tento checkpoint – najdlhší reťazec ktorý sme doteraz našli
        if(T[tpos][convert(S[pos])>0){           // ak je ďalšie písmeno zhodné zo stromu a v reťazci
            tpos=T[tpos][convert(S[pos])];           // presuneme sa na ďalšie písmeno
            pos++;
        }
        else break;           // inak skončí a vráť najdlhší reťazec ktorý sa začína na
        danom mieste
    }

```

```

    return naj;           // vráť naj
}

```

```

for(int i=1; i<=p; i++){           // načítam reťazce
    char peciatka[5047];
    scanf("%s",&peciatka);
    add(peciatka);           // pridám ich do stromu
}

```

```

int zac=0,kon=-1;           // zatiaľ máme znaky zmatchované po -1 v reťazci – ešte ani
jeden nieje zmatchovaný
for(int i=0; i<slength; i++){           // prezrieme každé písmenko 300 000 reťazca či v ňom
    nezačína nejaké slovo zo stromu
    int temp=controll(i);           //počet znakov ktoré sme zmatchovali

```

```

    if(kon<i+temp)kon=i+temp-1;           //rozšírim interval zmatchovaných znakov

```

```

    if(kon<i) pocet++;           // ak sa dané písmenko nenachádza v nijakom slove tak ho
    pripočítame

```

}

Preto pamäťová zložitosť bude $O(k)$, kde k je počet znakov vo všetkých reťazcoch
časová – pre každé písmenko v 300 000 reťazce potrebujeme zistiť či neexistuje nejaké slovo, ktoré
je najviac h písmen dlhé, tak potom časová zložitosť bude $O(k+l*h)$ kde l je dĺžka 300 000 reťazca