WPI 2020/21 **Kolokwium 2** 17-12-2020

Zadanie 1 (autor zadania i rozwiązania: Piotr Chrząstowski)
W tym zadaniu n>1.
W n-elementowej tablicy A dla każdego i=1,...n-2 jest spełniony warunek A[i+1]-A[i]>A[i]-A[i-1]. Napisz funkcję

```
int BliscySąsiedzi(int A[], int n),
```

która wyznaczy wartość min(|A[i]-A[i+1]|) dla i=0,...n-2.

Rozwiązanie:

Ponieważ różnice między kolejnymi elementami maleją, więc wartości bezwzględne tych różnic w ogólnym przypadku najpierw rosną, osiągają wartość maksymalną (być może dla dwóch sąsiednich par), a potem maleją. Tak się stanie, jeśli różnice są z początku ujemne, a na końcu dodatnie. Jeśli znak różnic jest stały, to wartość minimalną moduł różnicy osiągnie na krańcu przedziału, czyli albo różnica między zerowym, a pierwszym elementem będzie minimalna, albo różnica między przedostatnim, a ostatnim elementem będzie minimalna.

Naszym zadaniem będzie zatem zlokalizowanie minimum wartości bezwzględnej różnic w ich ciągu prawie bitonicznym. Prawie, bo jest możliwe, że ciąg ten ma dwie równe wartości w swoim maksimum. Po zauważeniu tego faktu reszta sprowadza się do prawie standardowego wyszukiwania binarnego.

Zatem:

```
int BliscySąsiedzi(int A[], int n),
//Niech m=indeks takiego elementu, że abs(A[m+1]-A[m]) jest minimalne
// Jeśli są dwa takie indeksy, to m niech będzie mniejszym z nich.
l=0; p=n-1;
while (l<p-1){//  0<=l<=m<p<n  &
//              &  dla każdego 0<=k<l abs(A[k+1]-A[k])>abs(A[l+1]-A[l])
//              &  dla każdego  p<k<n abs(A[k+1]-A[k])>abs(A[p+1]-A[p])
  s=(l+p)/2;  // l<s<p; dozór pętli gwarantuje ostrość obu nierówności!
```

```
  if (abs(A[s]-A[s-1])>abs(A[s+1]-A[s])) l=s+1; // m > s
  else p=s; //m<=p}
return abs(A[l+1]-A[l]);
}
```

Zauważmy, że dzięki sprytnemu dozorowi pętli, który przerywa pętlę na krok przed zrównaniem się indeksów, pozbywamy się kłopotu z zakresem wewnątrz pętli: wartości A[s-1], A[s] i A[s+1] są określone, a dodatkowo po wyjściu z pętli nie musimy badać szczególnych przypadków, gdy minimum jest przyjmowane na początku lub końcu tablicy. Również, gdy są dwa minima, to l pokazuje na lewe z nich.

Zadanie 2 (autor zadania i rozwiązania: Piotr Hofman)
Napisz gramatykę generującą język słów nad alfabetem {`a,b`} takich, że różnica między liczbą liter `a` na pozycjach podzielnych przez 3, a liczbą liter `a` na pozycjach niepodzielnych przez 3 wynosi 1.
Załóż, że pierwsza litera w słowie ma pozycję równą 1.
Przykładowo do języka tego należą słowa `bba, abaababbab,` zaś słowa `a, aba, bb` nie należą.

Rozwiązanie

Symbol startowy S

Nieterminal X(p,k) dla p,k$\in$$\{0,1,2\}$ oznacza słowa których efekt jest zerowy i są rozpięte pomiędzy pozycjami o reszcie modulo 3 równej odpowiednio początek=p, koniec = k (przy czym koniec nie należy do słowa tak jak w tablicach w C), czyli jest 9 nieterminali typu X

e = epsilon

S → X(1,0) a X(1,1) | X(1,0) a X(1,2) | X(1,0) a X(1,0)

X(1,0) → b X(2,0) | a X(2,0) a X(1,0)
X(1,1) → e | b X(2,1) | a X(2,0) a X(1,1)

$X(1,2) \rightarrow\ b\ X(2,2)\ |\ a\ X(2,0)\ a\ X(1,2)$

$X(2,0) \rightarrow\ b\ X(0, 0)|\ a\ X(0,0)\ a\ X(1,0)$
$X(2,1) \rightarrow\ b\ X(0,1)\ |\ a\ X(0,0)\ a\ X(1,1)$
$X(2,2) \rightarrow e\ |\ b\ X(0,2)\ |\ a\ X(0,0)\ a\ X(1,2)$

$X(0,0) \rightarrow e\ |\ b\ X(1,0)\ |\ a\ X(1,1)\ a\ X(2,0)\ |\ a\ X(1,2)\ a\ X(0,0)$
$X(0,1) \rightarrow\ b\ X(1,1)\ |\ a\ X(1,1)\ a\ X(2,1)\ |\ a\ X(1,2)\ a\ X(0,1)$
$X(0,2) \rightarrow\ b\ X(1,2)\ |\ a\ X(1,1)\ a\ X(2,2)\ |\ a\ X(1,2)\ a\ X(0,2)$

Niezmienniki dotyczące nieterminali gwarantują, że nie wygenerujemy, żadnego niepoprawnego słowa.

Dowód, że wszystko da się zrobić, trzeba zrobić indukcyjnie poprzez wskazanie jak budować słowo. Najpierw należy wziąć słowo i narysować wykres wartości licznika jako funkcja pozycji w słowie i potem według wykresu je budować.