Maksimalna suma nesusednih u nizu

Postavka : Dat je niz *a* prirodnih brojeva dužine *n*. Odrediti podniz datog niza čiji je zbir elemenata maksimalan a u kome ***nema susednih*** elemenata.

Rešenje :

Pristup rešavanju ovog problema u kome bi se tražili svi mogući podnizovi nesusednih elemenata nije dobar, pre svega zbog složenosti. Zbog toga ćemo polazni problem razložiti na nalaženje traženog podniza (podniza koji ima maksimalnu sumu) na nekom **delu niza *a* ,** tj. definisaćemo niz *d* na sledeći način :

$$d[k] = maksimalni\ zbir\ nesusednih\ elemenata\ niza\ (a_1, a_2, \dots, a_k), za\ k \in [1 \dots n]$$

Rešenje polaznog problema (tražena suma) će biti $d[n]$.

Vrednosti elemenata niza *d* mogu se popunjavati rekurzivno. Pretpostavimo da smo izračunali vrednosti elemenata niza *d* do *(k-1)*-og elementa (dakle, našli smo maksimalnu sumu nesusednih elemenata niza $(a_1, a_2, \dots, a_{k-1})$) I hoćemo da izračunamo vrednost elementa $d[k]$. Prilikom računanja ove vrednosti posmatramo element $a[k]$ jer on do sada nije učestvovao u formiranju prethodnih suma.

Ukoliko $a[k]$ učestvuje u formiranju vrednosti $d[k]$ (maksimalne sume nesusednih elementa $(a_1, a_2, \dots, a_k)$), tada element $a[k-1]$ ne sme ući u istu sumu. Element $a[k-1]$ je potencijalno mogao biti uključen samo u sumu $d[k-1]$ pa se element $a[k]$ može dodati na sumu $d[k-2]$ jer tu sigurno ne učestvuje ni jedan element koji je susedan sa $a[k]$.

Ukoliko $a[k]$ ne učestvuje u formiranju vrednosti $d[k]$, tada je $d[k] = d[k-1]$ (svejedno je da li posmatramo podniz $(a_1, a_2, \dots, a_{k-1})$ ili $(a_1, a_2, \dots, a_k)$).

Ovo se može formalno zapisati kao :

$$d[k] = \max\{\ d[k-1], a_k + d[k-2])\ \}\ , k \geq 3$$

Baza rekurzije je :

$$d[1] = \max\{0, a_1\}$$
$$d[2] = \max\{0, a_1, a_2\}$$

Nakon izračunavanja maksimalne sume, potrebno je **rekonstruisati** koji to elementi niza čine maksimalnu sumu. Ako se pogleda rekurentna formula, vidi se da ukoliko element $a[k]$ ne učestvuje u formiranju elementa $d[k]$, tada je $d[k] = d[k-1]$. Polazeći od poslednjeg elementa niza *d*, I koristeći ovo zapažanje, vrlo jednostavno se može rekonstruisati koji elementi čine traženu sumu.

Napomena : Na ovakav način će se dobiti elementi u obrnutom poretku.

```
Input: Niz a prirodnih brojeva dužine n
Output: subsequence - podniz nesusednih elemenata sa najvećom sumom

if n = 1 then
    return subsequence = a;
end
d[1] = max{0,a1}
d[2] = max{0,a1,a2}
for k = 3 to n do
    if d[k-1] > d[k-2] + a[k] then
          d[k] = d[k-1];
    else
          d[k] = d[k-2] + a[k];
end
subsequence = [];
currentIndex = n;
while (currentIndex > 0) do
    if d[currentIndex] <> d[currentIndex - 1] then
          add a[currentIndex] to subsequence;
          currentIndex = currentIndex - 2;
    else
          currentIndex = currentIndex - 1;
    end
end
return subsequence;
```

Podniz date sume

Postavka : Dat je niz prirodnih brojeva *a* dužine *n* i prirodan broj S≤ 10^5. Pronaći podniz niza *a* čija je suma jednaka S ili ustanoviti da takav niz ne postoji.

Rešenje :

Jedan od mogućih pristupa u rešavanju ovog problema bi mogao da bude formiranje svih mogućih podnizova i ispitivanje da li neki od njih ima sumu S. Niz dužine *n* ima 2^n podnizova što efikasnost pristupa u kome bi se formirali svi podnizovi odmah dovodi u pitanje.

Pošto za broj S postoji ograničenje koje nije toliko strogo, drugi pristup bi mogao da bude rešavanje polaznog problema na način da se ustanovi postojanje svih suma manjih od S. Definišimo niz *sum* na sledeći način :

$$sum[k] = \begin{cases} true, & ukoliko\ postoji\ podniz\ sa\ sumom\ k \\ false, & ukoliko\ ne\ postoji\ podniz\ sa\ sumom\ k \end{cases}$$

Niz *sum* se može računati rekurzivno I to na sledeći način :

1. Na početku se svi elementi niza *sum* postave na *false* osim elementa *sum[0].*
2. Pretpostavimo da smo dodali sve elemente $a[1], a[2], \dots a[k-1]$ (dakle, podniz a^{k-1}) da bi se napravile neke sume, tj. da bi se neki elementi niza *sum* promenili na *true.*
3. Posmatrajmo element $a[k]$. Njegovim uključivanjem u podniz postiže se formiranje svih suma $s + a[k]$ za svaku sumu *s* koja je do tada formirana I manja od S.

Odgovor na pitanje da li postoji podniz polaznog niza čija je suma S se dobija ispitivanjem vrednosti elementa *sum[S].*

Rekonstrukcija elemenata podniza koji ima sumu S se može postići ako se za svaki element niza *sum* pamti element niza *a* preko koga smo vrednost tog elementa postavili na *true.* To je, dakle, još jedan dodatni niz koji se na početku inicijalizuje npr. sa -1. Na taj način se, jednostavnom iteracijom po „prethodnicima“ počevši od „prethodnika“ elementa *sum[S]* može rekonstruisati podniz.

Nađeni podniz nije jedinstven. Ukoliko bismo želeli sve podnizove sume S, onda bi se za svaki element niza *sum* pamtili svi prethodnici. Drugi način je da se svi prethonici računaju pri samoj rekonstrukciji (*k-ti* element je prethodnik elementu *sum[s]* ukoliko je $sum[s - a[k]] = true$).

Napomena: Niz *sum* se obilazi od nazad! Ukoliko bi se obilazio od napred, onda bi svaki element niza mogli iskoristiti proizvoljan broj puta. Recimo, ako posmatramo element $a[k]$ tada će se (najkasnije) element $sum[a[k]]$ postaviti na *true.* Uzimajući u obzir da se u sledećim koracima posmatraju elementi $sum[a[k]+1], sum[a[k]+2] \dots$ doći ćemo u situaciju da se element $sum[a[k]+a[k]]$ postavi na *true* jer je $sum[a[k]]$= *true*.

```
Input: Niz a dužine n; broj  S
Output: subArray - podniz sa sumom S (prazan ukoliko takav ne postoji)

for = 0 to S do
    sum[i] = false;
    prior[i] = -1;
end
sum[0] = true;
for k = 1 to n do
for s = S downTo 0 do
    if sum[s] and s + a[k] ≤ S then
          sum[s + a[k]] = true;
          prior[s + a[k]] = k;
    end
end
end
subArray = [];
if sum[S] then
    currentSum = S;
    while prior[currentSum] <> -1 do
          add prior[currentSum] to subArray;
          currentSum = currentSum - a[prior[currentSum]];
    end
end
return subArray;
```