TEAP, LS 2021/2022
midterm, 27. 4. 2022

1. Máme n vecí s veľkosťami $a_0, \dots, a_{n-1}$ (sú to veci, t.j. $a_i > 0$). Z ľavej strany odobreieme niekoľko (označme si ich počet i) vecí a označíme k ich súčet, t.j. $k = \sum_{\ell=0}^{i-1} a_\ell$. Zostanú nám veci $a_i, \dots, a_{n-1}$. Z pravej strany odoberieme niekoľko (označme si ich počet j) vecí tak, aby ich súčet bol opäť k. Navrhnite algoritmus, ktorý v čase $O(n \log n)$ (horší nechcem) určí najväčšie k, ktoré takto vieme dostať. Napr. pre vstup $[7, 3, 20, 5, 15, 1, 11, 8, 10]$ je odpoveď 30 (z ľava odoberieme veci $7, 3, 20$ a sprava $10, 8, 11, 1$). Pre vstup $[1, 2, 4, 8, 16]$ je odpoveď 0. Zdôvodnite korektnosť a odvoďte zložitosť.

2. Určite tesnú asymptotickú zložitosť (v závislosti od N) každého z nasledovných troch programov:

```
sum = 0;
for ( n=N; n>0; n/=2 )
  for ( i=0; i<n; i++ )
     sum++;
```

```
sum = 0;
for ( i=1; i<N; i*=2 )
  for( j=0; j<i; j++ )
     sum++;
```

```
sum = 0;
for ( i=1; i<N; i*=2 )
  for ( j=0; j<N; j++ )
     sum++;
```

3. Čo ráta volanie `f(n,n)`? Má toto volanie polynomiálnu zložitosť? Prečo?

```
int f(int a, int b) {
  if (a < 3) return a * b;
  if (b == 1) return f(a - 1, a);
  return f(a, 1) + f(a, b - 1);
}
```

4. Majme hrací plán, ktorý pozostáva z poľa `A`, ktoré má dĺžku n a obsahuje celé (kladné aj záporné) čísla. Na začiatku je robot na pozícii $i = 0$, má rýchlosť $v = 0$ a nažratosť $z = 0$. V jednom ťahu robot zožerie políčko, na ktorom stojí (t.j. $z := z + \mathtt{A}[i]$), môže upraviť svoju rýchlosť o najviac ± 1 ale tak, aby nebola záporná (t.j. $v := v$, alebo $v := \pm 1$) a napokon sa posunie o v políčok doprava (t.j. $i := i + v$). Cieľom je, aby robot pristál na poslednom políčku a bol čo najviac nažratý. Navrhnite algoritmus, ktorý zistí, koľko najviac sa robot môže nažrať. Napríklad pre vstup $A = \{1, 1, -10, 1, -2, -2, 10, -100, 1\}$ je výstup 14, lebo najlepšie, čo robot môže urobiť, je:

i	v	z	
0	0	0	stojí na začiatku, zožerie `A[0]` a zvýši rýchlosť
1	1	1	na 1ku prišiel s rýchlosťou 1, zožerie `A[1]` a zvýši rýchlosť
3	2	2	stále zrýchľuje
6	3	3	teraz zožerie `A[6]` a spomalí
8	2	13	nakoniec zožerie posledné políčko

---

5. **(bonusová úloha)** Na vstupe je mapa M rozmerov $n \times m$, pričom $M[i][j]$ reprezentuje výšku políčka v metroch ($n, m \le 1000$, $1 \le M[i][j] \le 10^9$). Čas začína bežať v roku 0, kedy nič nie je zatopené a potom každý rok voda stúpne o meter, t.j. po prvom roku sú zaplavené políčka výšky 1, po druhom roku políčka výšky ≤ 2 atď. Na vstupe ďalej nasleduje číslo $T \le 10^5$ a T otázok $t_1 \le t_2 \le \dots \le t_T \le 10^9$. Pre každú otázku máte vypísať počet ostrovov po t_i rokoch (za ostrov považujeme súvislý úsek hranou susediacich políčok). Napr. pre vstup:

```
4 5
1 2 3 3 1
1 3 2 2 1
2 1 3 4 3
1 2 2 2 2
5
1 2 3 4 5
```

je výstup `2 3 1 0 0`. Navrhnite čo najefektívnejší algoritmus, dokážte správnosť a odvoďte zložitosť.