

*Podpíšte tento list aj dvojhárok – meno, priezvisko a osobné číslo.*

*Odpovede píšte priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.*

*Meno a priezvisko:*

*osobné číslo:*

A	B	C	D	E	F	G	H	I	J	K	bonus
3	3	2	2	2	2	8	2	7	6	8	10

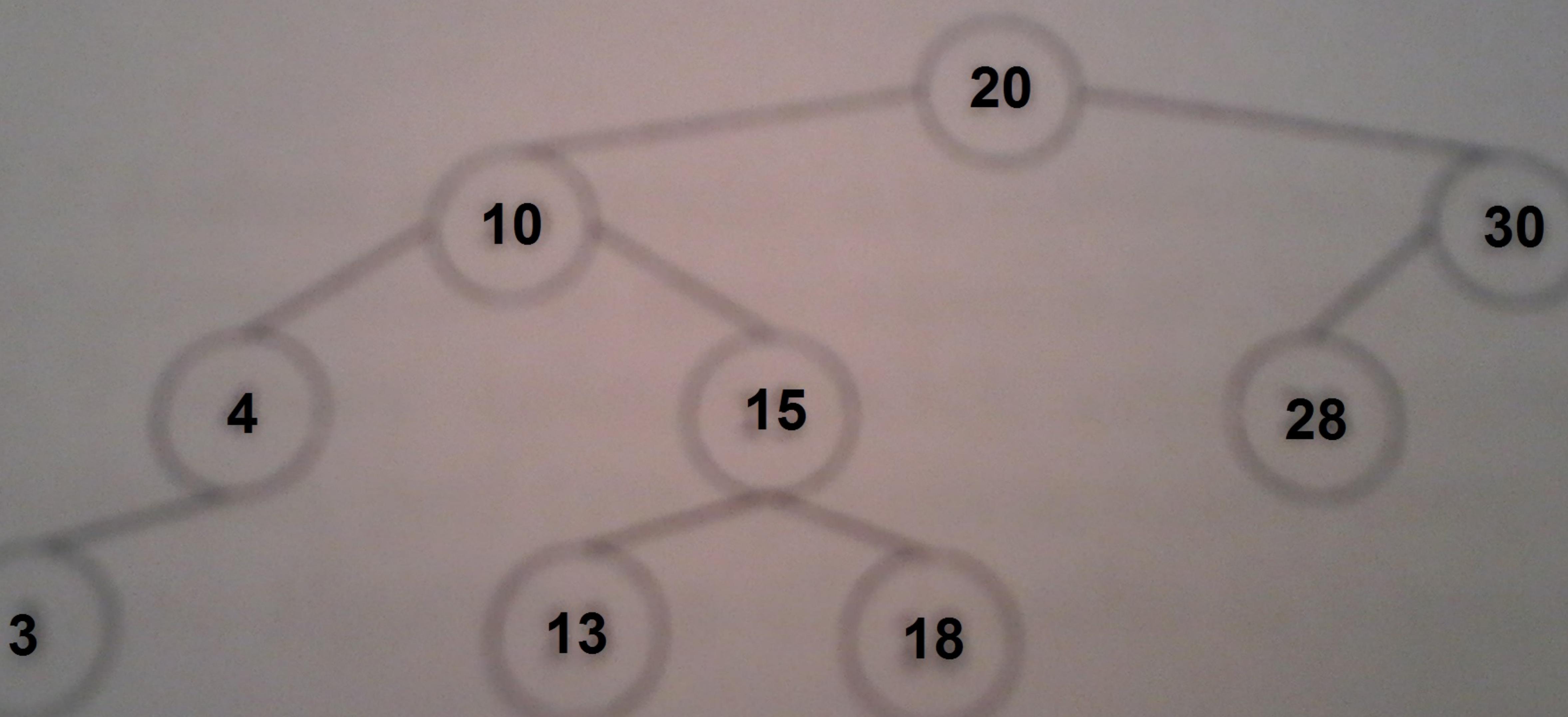
**A 3** Uvažujte prázdnú rozptylovú tabuľku, ktorej veľkosť je 6 (miest pamäti) a rozptylová funkcia je  $h(x) = x \bmod 6$ , pričom kolízie sa riešia zreťazením. Nakreslite náčrt stavu po vložení postupnosti prvkov (klúčov) 35, 2, 18, 6, 3, 10, 8, 5.  
(Nekreslite priebežné stavy.)

**B 3** Uvažujte prázdnú rozptylovú tabuľku, ktorej veľkosť je 11 (miest pamäti) a rozptylová funkcia je  $h(x) = x \bmod 11$ , pričom kolízie sa riešia lineárnym skúšaním. Nakreslite náčrt stavu po vložení postupnosti prvkov (klúčov) 35, 2, 18, 6, 3, 10, 8, 5.

**C 2** Uvažujte AVL strom na obrázku. Nakreslite strom po vložení prvku 17.

C2

Utratuto AVL sevora na chytrém NLP



Podpíšte tento list aj dvojhárok – meno, priezvisko a osobné číslo.

Odpovede píšte priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.

Meno a priezvisko:

osobné číslo:

**D 2** Toto je pole práve po prvom rozčlenení v algoritme rýchleho usporadúvania (quicksort):

3, 0, 2, 4, 5, 8, 7, 6, 9

Ktorý z týchto prvkov mohol byť pivot? (môže ich byť viac takých!)

**E 2** Predpokladajme, že sa usporadúva pole siedmich celých čísel (pomocou haldy, heapsort). Stav po jednom z vyhaldovaní (heapify) je:

6 4 5 1 2 7 8

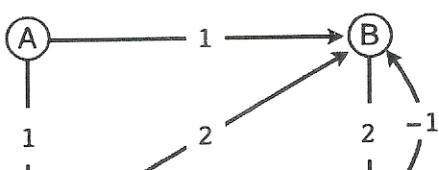
Koľko vyhaldovaní už prebehlo?

- A. 1
- B. 2
- C. 3 alebo 4
- D. 5 alebo 6

**F 2** Predpokladajme, že pri usporadúvaní 100 prvkov výberom (selection sort) práve prebehlo 42 opakovaní hlavného cyklu. Koľko prvkov je zaručene na svojom definitívnom mieste (a už sa s nimi určite nebude hýbať)?

- A. 21
- B. 41
- C. 42
- D. 43

**G 8 Bellman-Ford** Uvažujte orientovaný graf s ohodnotenými hranami, ktorý má aj záporne ohodnotené hrany, avšak nemá záporné cykly. Dokončíte stopu výpočtu Bellmanovho-Fordovho algoritmu, ktorý sa začne vykonávať vo vrchole A, ako je znázornená v tabuľke nižšie. Ako pomôcka je uvedená stopa prvej fázy relaxácie. (successful relax = úspešná relaxácia hrany, shortest known distance to = najkratšia doteraz známa vzdialenosť do). Hrany sa v každej fáze relaxujú v poradí, v akom sú uvedené pod sebou v tabuľke.



A

1

C

1

2

- 1

B

2

- 1

D

Podpište tento list aj dvojhárok – meno, priezvisko a osobné číslo.

Odpovede pište priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.

Meno a priezvisko:

osobné číslo:

		successful relax?	shortest known distance to			
			A	B	C	D
			0			
phase 1	D→B					
	C→D					
	C→B					
	B→D					
	A→C	X			1	
	A→B	X		1		
phase 2	D→B					
	C→D					
	C→B					
	B→D					
	A→C					
	A→B					
phase 3	D→B					
	C→D					
	C→B					
	B→D					
	A→C					
	A→B					
phase 4	D→B					
	C→D					
	C→B					
	B→D					
	A→C					
	A→B					

**H 2 Knuth Morris Pratt.** KMP algoritmus na hľadanie výskytu vzorového reťazca  $p$  v texte využíva predvypočítanú predponovú funkciu, ktorá je daná tabuľkou. Pre vzor  $p=ababaca$  ukážte zvyšné dva kroky počítania predponovej funkcie (doplňte hodnoty do tabuľky).

q	1	2	3	4	5	6	7
p	a	b	a	b	a	c	a
$\pi$	0	0	1	2	3		

**I 7 Sedem algoritmov usporadúvania.** Nižšie je znázornený vľavo stĺpec so vstupnou postupnosťou reťazcov, ktoré sa majú usporiadať, vpravo stĺpec s výslednou usporiadanou postupnosťou. Medzi nimi je 7 stĺpcov znázorňujúcich nejaký priebežný stav usporadúvania podľa niektorého z algoritmov, uvedených pod tým aj s písmenami, ktoré ich pre jednoduchosť označujú. Napíšte pod každý stĺpec písmeno algoritmu, podľa ktorého sa v tom stĺpci usporadúva.

Podpíšte tento list aj dvojhárok – meno, priezvisko a osobné číslo.

Odpovede píšte priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.

Meno a priezvisko:

osobné číslo:

fuzz	zoom	doze	benz	cozy	cozy	benz	cozy	benz
cozy	zest	cozy	cozy	czar	czar	buzz	fuzz	buzz
zinc	zone	czar	zinc	doze	fuzz	cozy	quiz	cozy
quiz	ritz	benz	quiz	fuzz	hazy	cruz	zinc	cruz
zero	zero	faze	zero	gaze	laze	czar	hazy	czar
suez	suez	cruz	suez	hazy	maze	daze	suez	daze
zone	zing	haze	zone	jazz	ouzo	doze	zero	doze
hazy	quiz	buzz	hazy	laze	quiz	faze	zone	faze
maze	maze	fuzz	maze	maze	suez	fuzz	czar	fuzz
ouzo	ouzo	gaze	ouzo	ouzo	zero	gaze	laze	gaze
czar	zeal	ritz	czar	quiz	zinc	haze	maze	haze
laze	raze	daze	laze	suez	zone	hazy	ouzo	hazy
doze	lazy	maze	doze	zero	doze	zone	doze	jazz
gaze	zeta	lazy	gaze	zinc	gaze	ouzo	gaze	laze
zing	zinc	whiz	zing	zing	zing	zing	jazz	lazy
jazz	jazz	hazy	jazz	zone	jazz	jazz	zing	maze
zoom	hazy	ooze	zoom	benz	zoom	zoom	buzz	ooze
cruz	cruz	ouzo	cruz	buzz	cruz	quiz	cruz	ouzo
ritz	cozy	zeal	ritz	cruz	ritz	ritz	ritz	quiz
buzz	buzz	jazz	buzz	daze	buzz	zinc	zoom	raze
faze	faze	zero	faze	faze	faze	laze	faze	ritz
zest	czar	size	zest	haze	zest	zest	raze	size
zeal	fuzz	zinc	zeal	lazy	zeal	zeal	zeal	suez
raze	laze	laze	raze	ooze	raze	raze	zest	whiz
ooze	ooze	zeta	ooze	raze	ooze	ooze	daze	zeal
lazy	doze	suez	lazy	ritz	lazy	haze	zero	
haze	haze	zing	haze	size	haze	zero	lazy	zest
daze	daze	quiz	daze	whiz	daze	suez	ooze	zeta
zeta	gaze	zoom	zeta	seal	zeta	zeta	benz	zinc
size	size	zeat	size	zest	size	size	size	zing
whiz	whiz	zone	whiz	zeta	whiz	whiz	whiz	zone
benz	benz	raze	fuzz	zoom	benz	maze	zeta	zoom

B      F      D      G      C      E      A

- A. nerekurzívne usporadúvanie zlučovaním (Bottom-up mergesort )
- B. usporadúvanie haldovaním (Heapsort )
- C. usporadúvanie vkladaním (Insertion sort )
- D. rýchle usporadúvanie (Quicksort )
- E. usporadúvanie výberom (Selection sort )
- F. Shellovo usporadúvanie (Shellsort )
- G. rekurzívne usporadúvanie zlučovaním (Top-down mergesort )

### J 6 Bitónne maximum

Postupnosť  $x_0, \dots, x_{n-1}$  je bitónna práve vtedy, ak pre nejaké  $x_k$ ,  $k \in \{0, \dots, n-1\}$  platí, že všetky prvky pred ním (vrátane jeho samotného) tvoria rastúcu postupnosť, kdežto prvky stojace za ním tvoria klesajúcu postupnosť. Formálne zapísané musí platiť, že:

$x_0 \leq x_1 \leq \dots \leq x_k \geq x_{k+1} \geq \dots \geq x_{n-1}$ . Navrhnite algoritmus, ktorý určí najväčší prvok v bitónnej postupnosti s  $N$  prvkami zapísanej v jednorozmernom poli v čase úmernom  $\log N$ .

Podpíšte tento list aj dvojhárok – meno, priezvisko a osobné číslo.

Odpovede píšte priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.

Meno a priezvisko:

osobné číslo:

- (a) Opíste algoritmus, použite stručný jasný opis najlepšie v algoritmickom jazyku, nepíšte úplný kód v programovacom jazyku. Hodnotiť sa bude správnosť, efektívnosť a čistota algoritmu.

- (b) Ukážte, ako pracuje Váš algoritmus uvedením prvých štyroch porovnaní, ktoré by vykonal pri hľadaní maxima v tomto 15-prvkovom bitónnom poli:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
a[i]	10	34	56	76	87	80	70	66	56	30	28	25	20	15	11

- 1.
- 2.
- 3.
- 4.

**K 8 Dynamické určovanie mediánu.** Načítava sa veľký počet čísel, ktoré sú všetky navzájom rôzne. Treba stále udržiavať hodnotu mediánu už prečítanej množiny čísel. Napr. po prečítaní čísel 2 9 7 4 1 treba vrátiť 4. Keď sa ďalej prečítajú čísla 6 8 5 a vyžiada sa medián, treba vrátiť 5 alebo 6. Pri navrhovaní riešenia treba dodržať tieto požiadavky:

- Možno použiť pomocnú pamäť iba v konštantnom rozsahu (navyše toho, čo treba pre zapísanie samotných čísel).
- Na vrátenie mediánu máte iba konštantný čas.
- Na spracovanie n-tého prvku máte čas, najviac úmerný  $\log n$ .

Možno zmazať časť už prečítaných čísel tak, aby sa neskôr vždy dal určiť medián všetkých prečítaných čísel?

- a) áno  
b) nie

Predpokladajme, že sa načítalo n čísel a ich medián je v. Ktoré z týchto tvrdení platia o mediáne po spracovaní n+1. prvku?

Podpíšte tento list aj dvojhárok – meno, priezvisko a osobné číslo.

Odpovede píšte priamo na tento list. Pracujete samostatne a odovzdáte len výsledky vlastnej práce, dosiahnuté bez pomoci.

Meno a priezvisko:

osobné číslo:

- A. Nezmenil sa.
- B. Je to najväčšie z čísel menších ako v.
- C. Je to najmenšie z čísel väčších ako v.
- D. Buď A. alebo B. alebo C.
- E. Buď B. alebo C, ale nie A.
- F. Môže to byť ľubovoľné z doteraz prečítaných čísel.

Jednou či dvoma vety opište, ako by ste vyriešili tento problém. (naozaj 2 vety, nie viac!)

### Bonus 10

Nech je dané pole N 64-bitových dlhých celých čísel so znamienkom. Treba nájsť dve čísla x a y také, že  $x+y=0$ . (Pre jednoduchosť predpokladajte, že ani jedno z čísel nie je 0 alebo  $-2^{63}$ ).

Opíšte efektívny algoritmus v rozsahu najviac pol strany. Váš algoritmus by mal počítač v lineárnom čase v priemernom prípade prípadne aspoň čase  $N \log N$  (za menej bodov).

Vaša odpoveď sa bude hodnotiť podľa správnosti, jasnosti a úspornosti vyjadrenia.

Zakrúžkujte odhad času vykonania Vášho algoritmu v priemernom prípade:

$\log N$      $N$      $N \log N$      $N^2$      $2^N$

Zakrúžkujte odhad času vykonania Vášho algoritmu v najhoršom prípade:

$\log N$      $N$      $N \log N$      $N^2$      $2^N$