

Na vyplnenie skúšky máte 120 minút.
Každý hárok ČITATELNE podpíšte!

Meno Priezvisko:

Boška vs. Bolan Michal

Fabrik Marek

1. (1 bod) Čo sa stane pri dereferencovaní smerníka NULL v kóde **jadra xv6**? Vyberte optimálne tvrdenie:

- vygeneruje sa výnimka výpadku stránky
- takáto dereferencia nie je možná
- dereferencia bude úspešná
- ide o nedefinovaný stav, následkom ktorého sa vygeneruje generálna výnimka ochrany

2. (1 bod) Izolácia procesov v kontexte prednášok znamená (vyberte najvhodnejšie tvrdenie):

- vynútenú separáciu z dôvodu **zapuzdrenia** dôsledkov zlyhaní
- dobrovoľnú separáciu z dôvodu **zapuzdrenia** dôsledkov zlyhaní
- dobrovoľnú integráciu z dôvodu **obnovy** dôsledkov zlyhaní
- vynútenú integráciu z dôvodu **obnovy** dôsledkov zlyhaní

3. (1 bod) Medzi hlavné nástroje **izolácie používateľského procesu v xv6** patrí (vyberte správne tvrdenie):

- vynútená kontrola toku riadenia pomocou systémových volaní
- kooperatívne plánovanie používateľských procesov
- preemptívne plánovanie používateľských procesov
- mapovanie trampolíny na tú istú virtuálnu adresu pre jadro aj používateľský proces

4. (1 bod) Kooperatívne plánovanie procesov znamená (vyberte jednu možnosť), že:

- proces sa musí sám rozhodnúť vzdať sa procesora
- sa procesu odoberie procesor v závislosti od jeho priority
- sa procesu odoberie procesor na základe prerušenia časovača
- proces na základe dočasného zvýšenia priority dokáže prevziať procesor

5. (1 bod) Režim **supervisor** procesora RISV-V v xv6 umožňuje oproti režimu **user** navyše (vyberte správne tvrdenie):

- pristupovať k VIRTIO zariadeniu
- zakázať generovanie výnimky výpadku stránky
- pristupovať k adresám, pre ktoré nie je platné mapovanie v tabuľke stránok
- zakázať generovanie výnimiek akéhokoľvek typu, pokým sa vykonáva kód jadra

6. (1 bod) Vyberte pravdivé tvrdenie **ohľadom xv6**:

- jadro OS nikdy nedokáže dereferencovať **fyzické adresy**
- jadro OS nikdy nedokáže dereferencovať **virtuálne adresy**
- jadro OS dokáže dereferencovať **aj virtuálne, aj fyzické adresy**
- jadro OS dokáže dereferencovať **bud' virtuálne, alebo fyzické adresy**, no nikdy nie oboje súčasne

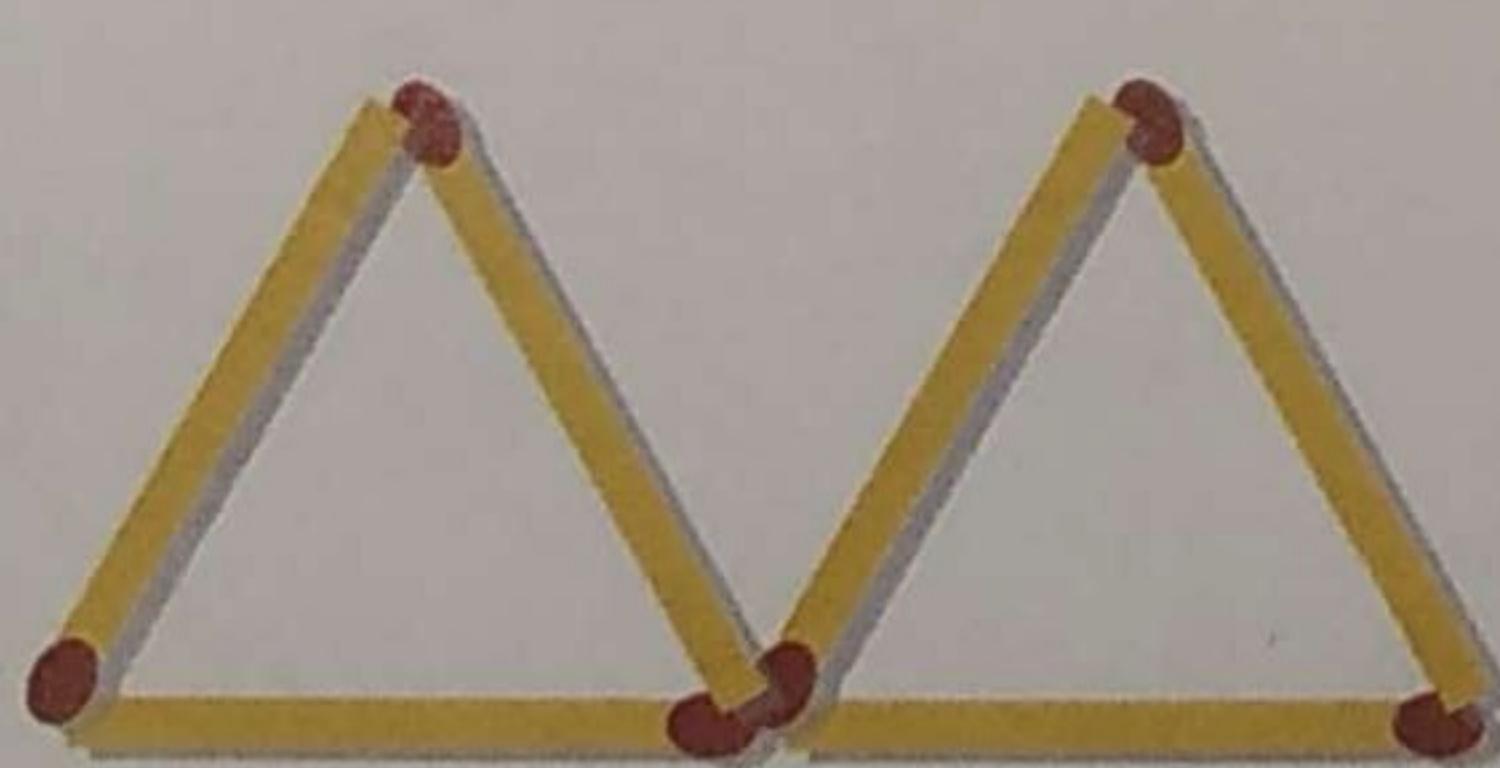
37. (1 bod) Popíšte, ako vieme rozlísiť priestor jadra a priestor používateľa podľa módu procesora. Ako príklad použite xv6.

✓
priestor jadra - ~~je to koreň mod~~ - väčšia je povolená
priestor používateľa - ~~nie mod~~ - nie väčšia je povolená, daže
môžem takto spustiť dve aplikácie
k výhodke

38. (1 bod) Prečo sa funkcie copyin() a copyout() nedajú využiť v obsluhe prerušení xv6?

✗ - v období prerušenia nesmí pôsobiť na konflikty
- nesmíto robať sa povolením load a store

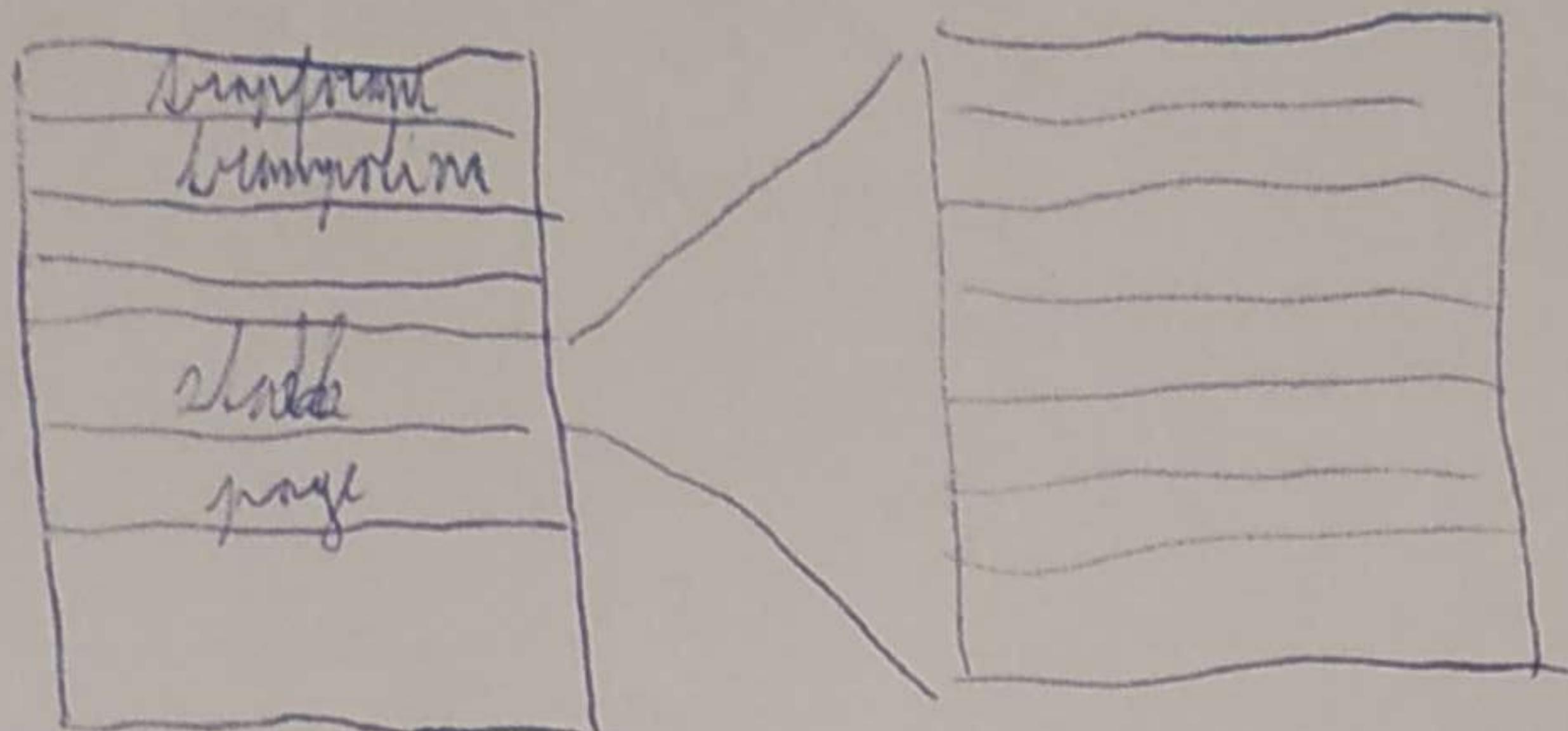
39. (2 body) BONUS: Prelož jednu zápalku tak, aby namiesto 2 \triangle vznikli štyri. Ako riešenie nakresli nový obrázok.



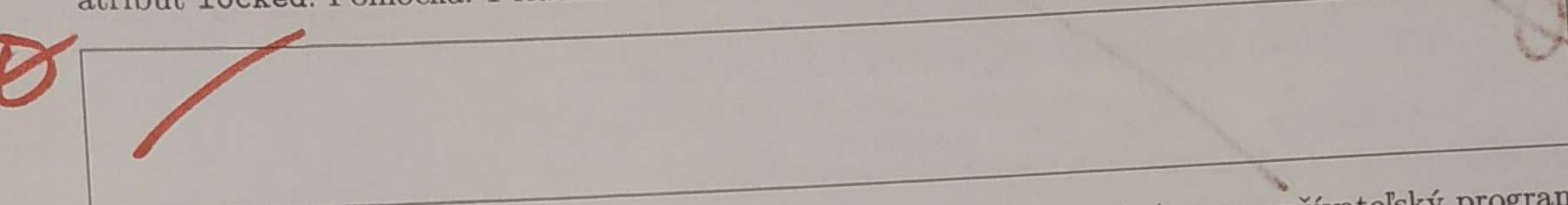
Meno Priezvisko: Patrik Novák

33. (2 body) Vysvetlite pojmom stack guard a načo sa využíva.

~~stack guard - sa využíva v poniavatelnom režime~~



34. (1 bod) Napíšte pseudokód metódy release ADT SpinLock. Interný stav zámku reprezentuje atribút locked. Pomôcka: 1 riadok.

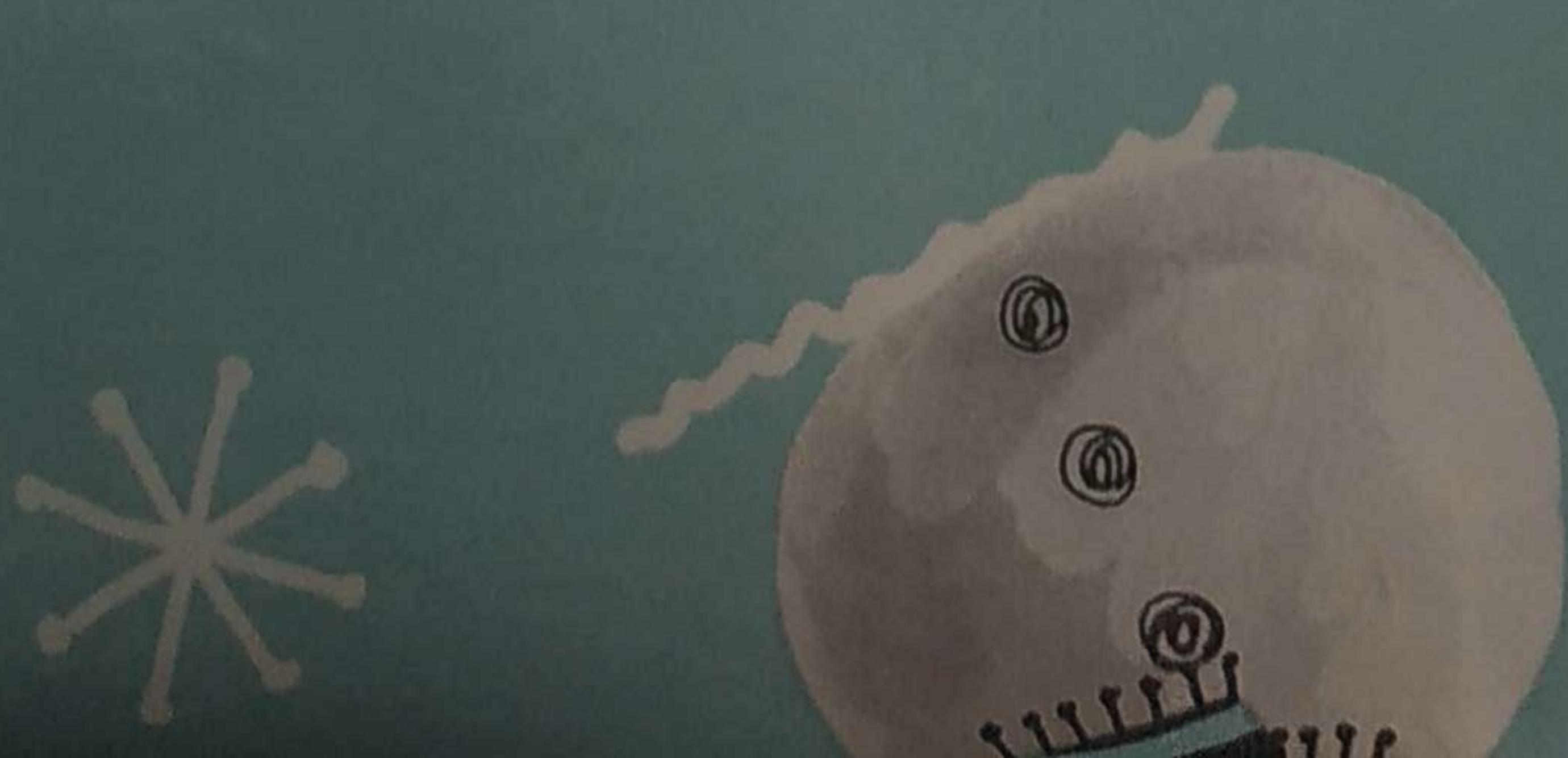


35. (2 body) Napíšte útržok programového kódu v jazyku C (dva riadky) pre používateľský program xv6, pomocou ktorého presmerujete štandardný chybový výstup do štandardného výstupu. Funkcie, ktoré môžete použiť: wait, fork, write, dup, close, exec, read, getcmd, open.

~~FD0 - štandardný výstup
FD1 - štandardný výstup
FD2 - štandardný chybový výstup~~

36. (2 body) Napíšte 4 služby, ktoré od OS očakávame.

~~efektivita, rýchlosť, jednoduchosť, plnosť programu, bezpečnosť~~ ✓



30. (2 body) Napíšte pseudokód **metódy acquire ADT SleepLock**. Interný stav zámku reprezentujú atribúty `locked` a FIFO fronta čakajúcich procesov `queue`. Metóda má k dispozícii odkaz na aktuálny proces, ktorý ju vyvolal: `current`. Zmenu stavu procesu robte pomocou metódy `setState`. Preplánovanie vyvolajte pomocou funkcie `yield()`.

~~8~~

~~ADT sleeplock~~
~~- vymenovanie na ďalší čas~~
~~- určenie preplánovania~~

31. (2 body) **Opíšte** stratégiu plánovania, ktorú používa jadro xv6. K opisu algoritmu použíte **pseudokód**.

~~6~~

32. (3 body) V niekoľkých bodech schematicky opíšte kroky, ktoré musí robiť obsluha výpadku stránky COW forku.

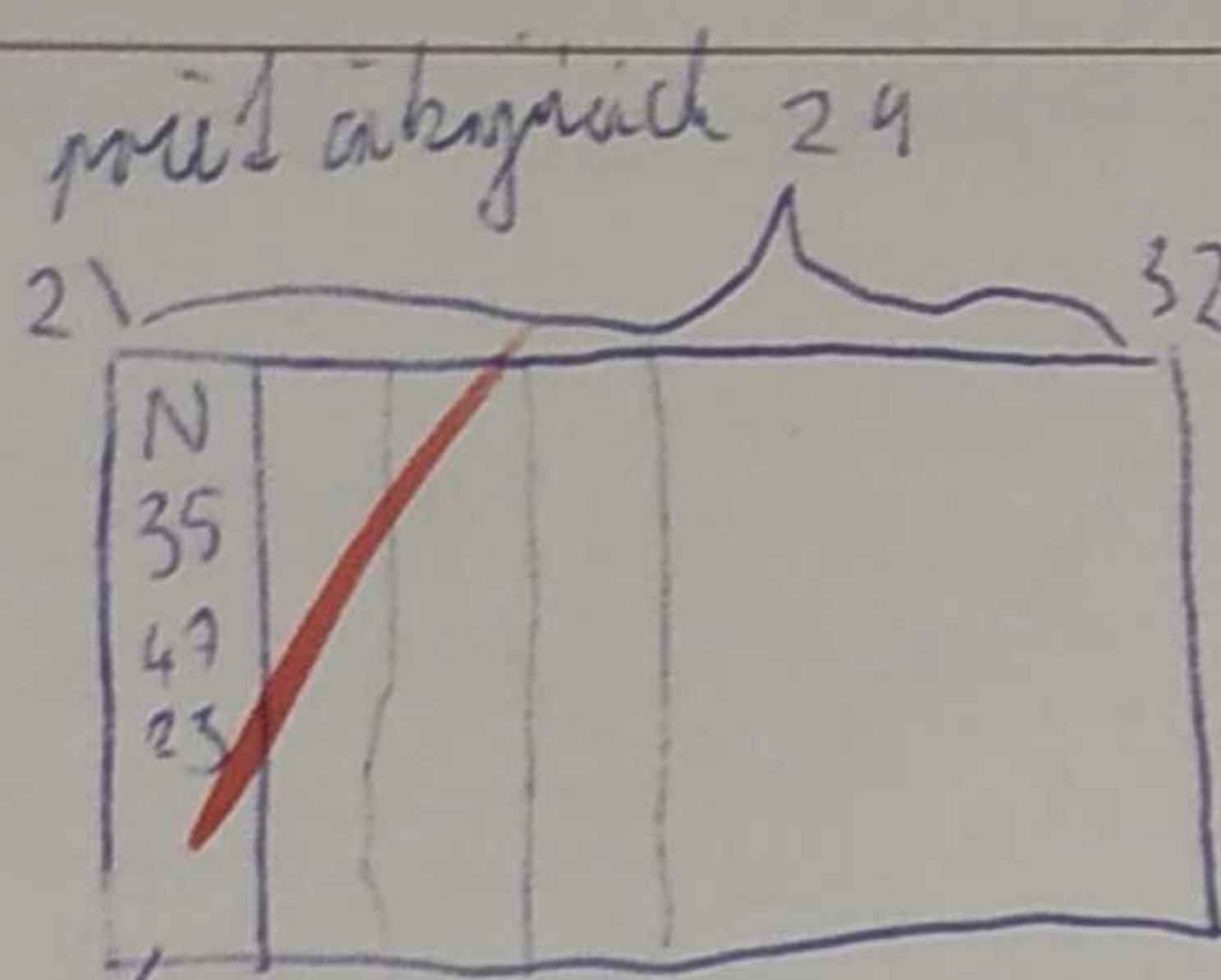
~~7~~

getaddr()
 ↓
 fork()
 ↓
 wait()
 ↓
 iun()

fork() - zjednotí celý VAP
 iun() - vymená celý VAP a aktualizáciu súboru
 fork(COW) - VA diského procesu pride do RAM

Meno Priezvisko: Patrik Novák

27. (2 body) Nakreslite štruktúru logu súborového systému xv6 na disku v situácii, keď žiadna disková transakcia nie je potvrdená. Nákres **vhodne** a **čitateľne** komentujte.



informácia 35 nie je potvrdená ďalej čísla

- ✓ 28. (2 body) Schematicky znázornite (nakreslite graf volaní alebo vývojový diagram), ako sa realizuje spustenie programu z príkazového riadku aplikácie *shell*. Funkcie, ktoré **môžete** (nie všetky sú vhodné!) v grafe použiť: *wait*, *fork*, *write*, *dup*, *close*, *exec*, *read*, *getcmd*, *open*. Neuvažujte presmerovania ani chybové stavy.

make option
ls
mail
fork
write
dup
close
read
getcmd
open

29. (1 bod) Prečo obsahuje i-uzol na disku informáciu o type súborového objektu (napr. T_DIR)?

~~aby i-uzol vedel o ktorom i-čile sa jedná~~

Meno Priezvisko: Petrík Norak

14. (2 body) Neupravená verzia xv6 obsahuje tieto typy súborových objektov (vyberte správnu odpoveď):

- 2
1 T_DEVICE, T_DIR, T_FILE
 T_DEVICE, T_DIR, T_FILE, T_LINK
 T_DIR, T_FILE, T_LINK
 T_DIR, T_FILE

15. (1 bod) Pomocou číslovania od 1 zoradte vykonávanie pseudo inštrukcií emulátora:

- 1 • for True:
2 • read next instruction
3 • decode instruction
4 • execute instruction (updating CPU state)

16. (1 bod) Čo je prvotným cieľom žurnálu súborového systému? Zvoľte najšpecifickejšiu odpoveď.

- 1 bezpečnosť dát
2 spokojnosť používateľa
3 efektivita využitia hardvéru
4 integrita súborového systému

17. (1 bod) Očíslovaním (od 1) označte poradie vykonávanie operácií logovacieho systému v xv6:

- 1,4 zápis N=0 do hlavičky logu na disk
2 zápis obsahu blokov na ich cieľové miesta na disk
3 zápis hlavičky logu na disk
4 synchrónne čakanie na dokončenie zápisov blokov
5 zapísanie blokov z bcache do logovacej oblasti na disk

18. (1 bod) Nenulová hodnota N v hlavičke logu na disku znamená (zvoľte správnu odpoveď):

- 1 počet úspešne inštalovaných blokov logu
2 počet blokov logu čakajúcich na inštaláciu
3 počet neúspešne inštalovaných blokov logu
4 počet pokusov o obnovu súborového systému z logu

19. (1 bod) CPU pri vykonávaní kódu jadra xv6 (zvoľte správnu odpoveď):

- 1 musí mať vypnuté prerušenia
2 môže mať vypnuté prerušenia
3 nesmie mať vypnuté prerušenia
4 síce môže, ale nemá vypnuté prerušenia

20. (1 bod) Z pohľadu kódu vykonávaného na CPU sú prerušenia (zvoľte správnu odpoveď):

- 1 symetrické
2 synchrónne
3 asymetrické
4 asynchónne

7. (1 bod) Návratová hodnota systémového volania xv6 sa vracia používateľskému procesu pomocou (vyberte správne tvrdenie):

- všeobecne dohodnutého registra a1
- špeciálneho registra procesora RISC-V
- rovnakého registra, ktorý sa používa na návratovú hodnotu funkcie
- ľubovoľného registra, ktorý sa definuje v hlavičkovom súbore `kernel/riscv.h`
- toho registra, ktorý sa používa na prechod z priestoru používateľa do priestoru jadra

8. (1 bod) Ako reaguje **nemodifikovaná** verzia xv6 na **výpadok stránky** v režime **jadra** (vyberte optimálne tvrdenie):

- ukončením** behu **xv6**
- ukončením procesu** používateľa, v kontexte ktorého prišlo v jadre ku výnimke
- doinštaluje** sa chýbajúce mapovanie a **reštartuje** sa kód, ktorý výpadok spôsobil
- ak ide o **legitímnú** žiadost jadra o prístup ku pamäti, **doinštaluje** sa chýbajúce mapovanie a **reštartuje** sa kód, ktorý výpadok spôsobil

9. (1 bod) OS xv6 na preemptívne plánovanie využíva: (označte očakávanú odpoveď)

- systémové volania
- hardvérové prerušenia
- špecifické systémové volanie *ecall*
- špecifické hardvérové prerušenie *časovača*

10. (1 bod) Odpovedajte ohľadom vlákien v jadre **nemodifikovaného** xv6. Označte jednu odpoved.

- jadro nepoužíva koncept vlákien
- kód jadra beží v jedinom vlákne
- kód jadra beží v dvoch vláknach: jednom pre používateľské procesy, druhom pre plánovač
- kód jadra beží v $N+M$ vláknach: N je počet používateľských procesov, M počet plánovačov

11. (1 bod) Označte optimálne tvrdenie.

- Zámky **veľmi zriedkavo zvyšujú** efektivitu paralelného vykonávania.
- Zámky **veľmi zriedkavo znižujú** efektivitu paralelného vykonávania.
- Zámky **vždy zvyšujú** efektivitu paralelného vykonávania.
- Zámky **vždy znižujú** efektivitu paralelného vykonávania.

12. (1 bod) Veľkosť najmenšej hardvérovej adresovateľnej jednotky pri práci s diskom je:

- 1 bajt
- 512 bajtov
- 1024 bajtov
- 2048 bajtov

13. (1 bod) Zvoľte správnu odpoveď vzhľadom na **nemodifikovanú** verziu xv6. Mapovanie vlákien užívateľského programu je vzhľadom na užívateľský priestor a priestor jadra v pomere (uveďte číselný pomer, napr. 0:9, M:-1, atď.):

8



21. (1 bod) Procesor RISC-V má (vyberte správne tvrdenie):

- 4 módy činnosti: user, machine, hypervisor a supervisor
 3 módy činnosti: user, machine a supervisor
 2 módy činnosti: user a supervisor

22. (1 bod) Kód obsluhy prerušenia v jadre xv6 (zvolte správnu odpoveď):

- musí bežať v kontexte používateľského procesu
 môže bežať v kontexte používateľského procesu
 nesmie bežať v kontexte používateľského procesu
 sice môže, ale nebeží v kontexte používateľského procesu

23. (2 body) Správne (čiarami) spojte výrazy z ľavého stĺpca s výrazmi v pravom stĺpci. Hľadajte odpoveď, ktoré systémové volanie je ktorou abstrakciou HW.

1	exec	zdieľaná fyzická pamäť	2
2	pipe	fyzická pamäť	1
3	fork	diskové bloky	4
4	open, read, write	jadrá CPU	3

24. (1 bod) Zvolte pravdivé tvrdenie:

- IRQ číslo reprezentuje **zdroj** prerušenia
 IRQ číslo reprezentuje **chybový kód** zariadenia
 výrobcovia hardvéru rešpektujú dohodu, že IRQ číslo **zariadenia** je na každej platforme **rovnaké**
 výrobcovia hardvéru rešpektujú dohodu, že IRQ číslo **UART zariadenia** je na každej platforme **rovnaké**

25. (2 body) Majme diskový priestor veľkosti 4 bloky vyhradený na i-uzly. Veľkosť bloku je 512 bajtov a veľkosť i-uzla 64 bajtov. Maximálne koľko objektov súborového systému môže v tejto konfigurácii koexistovať? Uveďte výpočet.

26. (1 bod) Koľko bitov potrebujeme na indexovanie (adresovanie) v rámci jednej stránky veľkosti 4 KiB? Uveďte výpočet.

$$2^{12} = 4096$$

12