

Priezvisko a meno:

1.	2.	3.	4.	5.	6.	7.	8.	9.

1. Viacvrstvový pohľad na hry. Písmenami sú označené jednotlivé vrstvy hry, herného engine. Usporiadajte ich od najnižšej po najvyššiu a odpoveď uveďte do tabuľky (písmeno označujúce najnižšiu vrstvu bude na spodku tabuľky):

- A. základný pohyb a vykresľovanie objektov
- B. hardvér
- C. vrstva nezávislá na zariadeniach a knižniciach
- D. pravidlá, logika hry, scenár hry, hrateľnosť
- E. operačný systém
- F. programovací jazyk a jeho knižnice

najvyššia	
najnižšia	

2. Urýchlenie výpočtov. Ktorý výraz počítal najrýchlejšie hodnotu  $\sin(x)$ ? Odpoveď zakrúžkujte.

- $\sin(x)$  ... zo štandardnej matematickej knižnice
- $x - 0.16605 * x * x * x + 0.00761 * x * x * x * x * x$
- $\sqrt{1 - \sin(x) * \sin(x)}$  ... zo štandardných matematických knižníc

Sem napíšte názov jazyka, v ktorom ste to skúšali .....

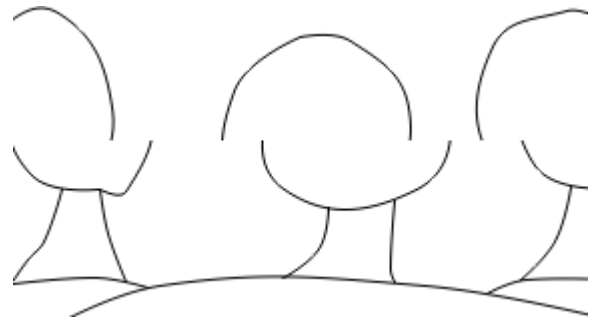
3. Ovládanie objektov. Ktorý z prístupov je vhodnejší pri tvorbe akčných hier? Z každej dvojice zakrúžkujte jeden výrok.

- Keďže je v hre veľa objektov, použijeme pre každý objekt zvlášť samostatný časovač.
- V hre zvykneme aj pri veľkom počte objektov používať jeden spoločný časovač.
- Pri stlačení klávesu si tento stav poznačíme a pohyb hráča aj objektov realizujeme až pri tiknutí časovača.
- Pri stlačení klávesu pohneme hráča a nakreslíme ho, v časovači hýbeme ostatné objekty z hry.
- Najskôr všetky objekty pohneme, až potom ich nakreslíme.
- Objekty kreslíme ihneď, ako ich pohneme.

4. Scéna so stromami sa plynulo posúva v horizontálnom smere. Prečo na obrazovke vidíme obraz, ktorý sa roztrháva v rôznych náhodných y-ových pozíciách? Správnu odpoveď zakrúžkujte.

Problém spočíva v tom, že:

- časovač tiká príliš pomaly
- obraz sa hýbe príliš pomaly
- je zlá vertikálna synchronizácia
- grafik nám taký obrázok nakreslil



5. Programujeme hru, ktorá sa odohráva vo vesmíre. Máme hviezdnu loď a 2 planéty. Chceme overiť, či program správne počíta gravitačnú silu, ktoré pôsobí na hviezdnu loď. Program sme zastavili v okamihu, keď:

Loď má:

súradnice  $[0, 0]$   
hmotnosť 1

Planéta 1 má:

súradnice  $[3, 4]$   
hmotnosť 1000

Planéta 2 má:

súradnice  $[8, 6]$   
hmotnosť 3000

Vypočítajte a doplňte odpovede:

Vzdialenosť lode od 1. planéty bude: .....

Vzdialenosť lode od 2. planéty bude: .....

... úloha pokračuje na ďalšom liste

Priezvisko a meno:

.....

Gravitačné sily počítame podľa vzorca:  $F = k * m_1 * m_2 / r^2$  ... nech je v našej hre  $k = 1$

Veľkosť gravitačnej sily medzi loďou a 1. planétou bude: .....

Veľkosť gravitačnej sily medzi loďou a 2. planétou bude: .....

Veľkosť gravitačnej sily, ktorá pôsobí na loď bude: .....

Vypočítajte vektor, teda x-ovú a y-ovú zložku gravitačnej sily, ktorá pôsobí na loď. Vektor bude:

( ....., .....

6. Simulujeme odrážajúce sa objekty (bez trenia a straty energie). V každom z nasledujúcich troch rôznych scenárov označte najpravdepodobnejšiu správnu odpoveď:

Všetky objekty sa pri zrážke postupne zrýchľujú:

- Asi sme nebrali do úvahy zákon zachovania hybnosti.
- Asi sme nebrali do úvahy zákon zachovania energie.
- Tak to má byť.

Vidíme, že po zrážke sa objekty pohybujú nesprávnym smerom:

- Asi sme nebrali do úvahy zákon zachovania hybnosti.
- Asi sme nebrali do úvahy zákon zachovania energie.
- Tak to má byť.

Vypisujeme súčet kinetických energií všetkých objektov – vidíme stále rovnakú hodnotu:

- Asi sme nebrali do úvahy zákon zachovania hybnosti.
- Asi sme nebrali do úvahy zákon zachovania energie.
- Tak to má byť.

7. Správanie objektov. Ideme vytvoriť hru *Salaš*. V nej bača stráži ovce. Bača má správanie:

- ak je ráno, vyženie ovce na lúku
- ak je večer, zaženie ovce do košiara
- v noci stráži ovce tak, že keď príde vlk, snaží sa ho odohnať.

Programujeme automat, ktorý uvedené správanie realizuje. Jednotlivé stavy sú objekty s metódami *Zmena*, *Začni*, *Vykonaaj*, *Skonči*.

Ak treba doplňte v metódach pre jednotlivé stavy chýbajúce časti. Použite „príkazy“ z nasledovnej ponuky:

- vyžeň ovce na lúku
- zažeň ovce do košiara
- zažeň vlka
- stav *deň*
- stav *noc*
- stav *vlk*
- ak je večer
  
- resp. napíšte slovo *nič*, ak má časť ostať prázdna.

... úloha pokračuje na ďalšom liste

Priezvisko a meno:

.....  
objekt pre stav *noc*:

metóda Zmena:  
ak príde vlk, výsledok je:  
.....  
ak je ráno, výsledok je:  
.....  
metóda Začni:  
.....  
metóda Skonči:  
.....

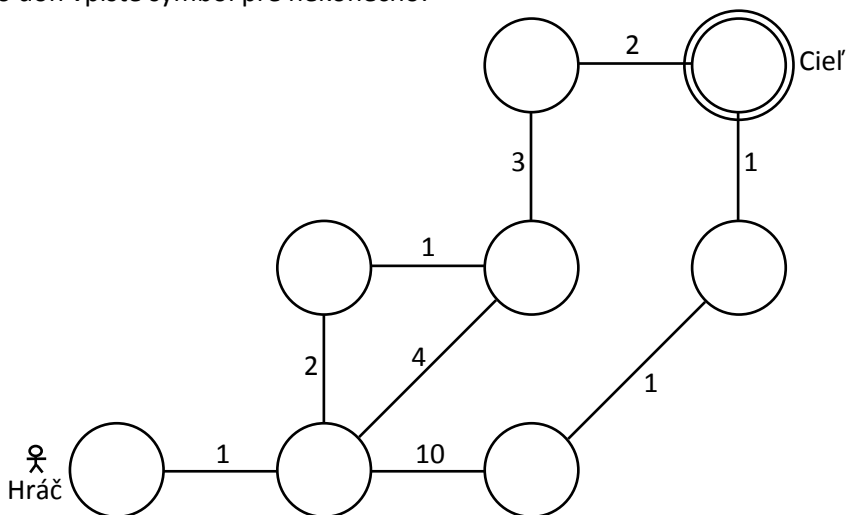
objekt pre stav *vlk*

metóda Zmena:  
ak je vlk preč, výsledok je  
.....  
metóda Vykonaj:  
.....

objekt pre stav *deň*:

metóda Zmena:  
....., výsledok je:  
.....  
metóda Začni:  
.....  
metóda Vykonaj:  
.....  
metóda Skonči:  
.....

8. Hernú plochu reprezentujeme pomocou neorientovaného grafu s hranami rôznej ceny. Program hľadá cestu pre postavičku hráča do cieľového vrcholu pomocou Dijkstrovho algoritmu. Do jednotlivých vrcholov vpíšte, aká bude cena cesty do vrcholu v okamihu, keď tento algoritmus skončí prehľadávanie grafu. Ak niektorý vrchol vôbec nebude navštívený, nechajte ho prázdny alebo doň vpíšte symbol pre nekonečno:



9. Nasledujúca, funkcia má realizovať algoritmus NEGAMAX. Doplňte do chýbajúcich častí čísla, výrazy s premennými *kto*, *hĺbka*, *najlepšie* a volanie *negamax* tak, aby funkcia správne pracovala:

*funkcia negamax(kto, hĺbka):*

*ak je hra vyhnaná, skonči funkciu s výsledkom (-100 + hĺbka)*

*ak je hracia plocha plná, skonči funkciu s výsledkom .....*

*najlepšie = .....*

*pre všetky neobsadené pozície:*

*polož(pozícia, kto)*

*v = .....*

*ak najlepšie < v:*

*najlepšie = .....*

*odlož(pozícia, kto)*

*skonči funkciu s výsledkom .....*