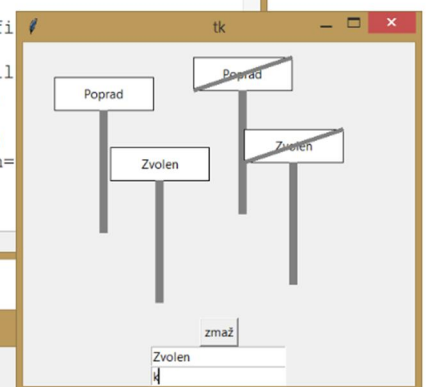
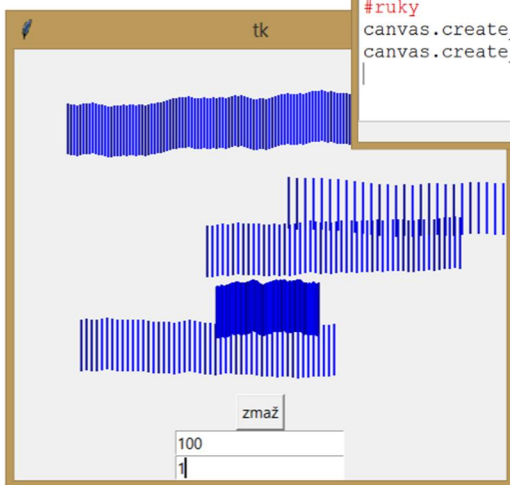
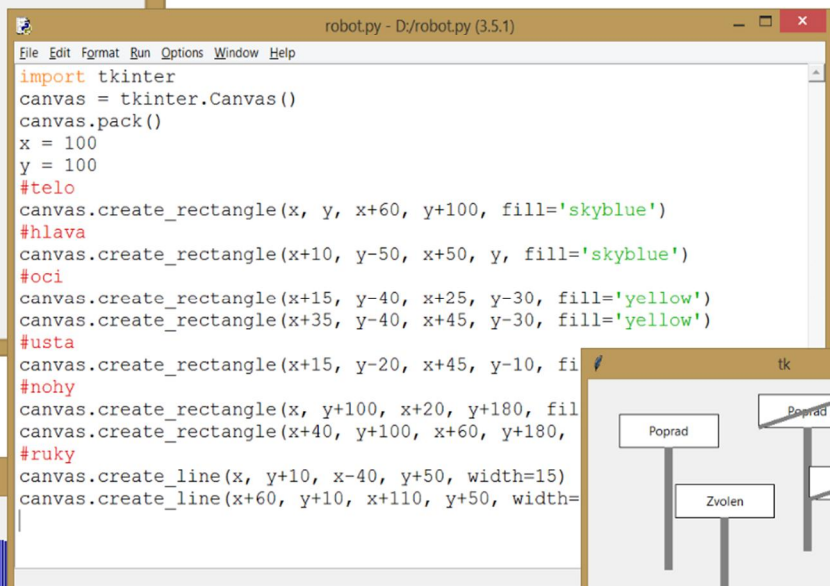
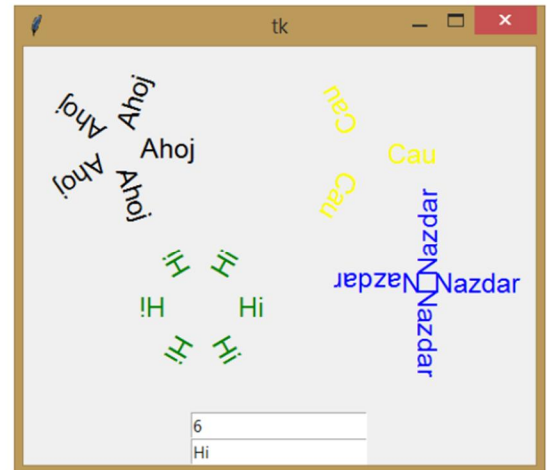
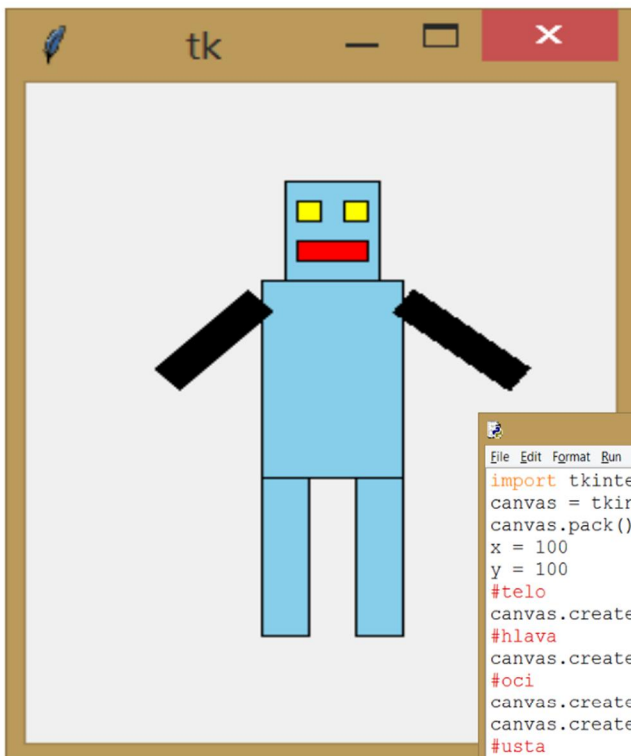


Testy k učebnici

Programujeme v Pythoně

Peter Kučera



Testy k učebnici Programujeme v Pythone

k učebnici informatiky pre stredné školy

Autor © Mgr. Peter Kučera

Design © Mgr. Peter Kučera

Prvé vydanie, 2016

Verzia číslo: 20. 8. 2016

Vydavateľ: Mgr. Peter Kučera

Údaje o nadobúdateľovi licencie:

Názov/meno: Katolícka univerzita v Ružomberku, Pedagogická fakulta, IČO: 37801279

E-mail: janka.majherova@ku.sk

Mesto/obec: Hrabovská cesta 1, 034 01 Ružomberok

Licencia číslo: 1101705003 – licencia len pre jedného používateľa

Upozorňujeme, že elektronická kniha je dielom chráneným podľa autorského zákona a je určená len pre vzdelávaciu činnosť v pôsobnosti nadobúdateľa. Kniha ako celok ani žiadna jej časť nesmie byť voľne šírená na internete, ani nijako ďalej zverejňovaná. V prípade ďalšieho šírenia neoprávnene zasiahnete do autorského práva s dôsledkami podľa platného autorského zákona a trestného zákonníka.

Veľmi si vážime, že e-knihu ďalej nešírite. Len vďaka vašim nákupom dostanú autori, vydavatelia a kníhkupci odmenu za svoju prácu. Ďakujeme, že tak prispievate k vzniku ďalších skvelých kníh.

Učebnicu a ďalšie materiály si môžete zakúpiť aj priamo na stránkach autora:

<http://www.programujemevpythone.sk/>

<https://www.facebook.com/programujemevpythone>

ISBN 978-80-972369-7-7 (pdf)

Obsah

Obsah.....	5
Úvod	7
Dôležité informácie.....	8
1. Testy s riešením a hodnotením	9
1. 1 Zadania testov	9
Test č. 1 A po kapitole 4.2	9
Test č. 1 B po kapitole 4.2	10
Test č. 1 A po kapitole 5	11
Test č. 1 B po kapitole 5	12
Polročný test č. 1 po kapitole 7	13
Polročný test č. 2 po kapitole 7	16
Test č. 1 A po kapitole 10	19
Test č. 1 B po kapitole 10	21
Test č. 1 A po kapitole 12	23
Test č. 1 B po kapitole 12	25
Test č. 2 po kapitole 11.....	27
Polročný test č. 1 po kapitole 14	29
Polročný test č. 2 po kapitole 14	33
Polročný test č. 3 po kapitole 14	36
1. 2 Riešenie testov	40
Test 5101A.....	40
Test 5101B.....	40
Test 5201A.....	41
Test 5201B.....	41
Test 5301	42
Test 5302	43
Test 6101A.....	45
Test 6101B.....	45
Test 6201A.....	46
Test 6201B.....	47
Test 6202	48
Test 6301	50
Test 6302	53
Test 6303	55
1. 3 Hodnotenie testov.....	58
Test 5101 AB.....	59
Test 5201 AB.....	60
Test 5301	61
Test 5302	63
Test 6101A.....	65
Test 6201A.....	66
Test 6202	68
Test 6301	69
Test 6302	72
Test 6303	75

2. Zadania ďalších testov	78
Test č. 2 A po kapitole 3	78
Test č. 2 B po kapitole 3	79
Test č. 2 C po kapitole 3	80
Test č. 3 A po kapitole 3	81
Test č. 3 B po kapitole 3	82
Test č. 2 A po kapitole 6	83
Test č. 2 B po kapitole 6	84
Test č. 3 A po kapitole 5	85
Test č. 3 B po kapitole 5	86
Test č. 4 A po kapitole 5	87
Test č. 4 B po kapitole 5	88
Test č. 4 C po kapitole 5	89
Test č. 4 D po kapitole 5	90
Polročný test č. 3 A po kapitole 7	91
Polročný test č. 3 B po kapitole 7	92
Polročný test č. 4 A po kapitole 7	93
Polročný test č. 4 B po kapitole 7	94
Polročný test č. 5 A po kapitole 7	95
Polročný test č. 5 B po kapitole 7	96
Polročný test č. 6 A po kapitole 7	97
Polročný test č. 6 B po kapitole 7	99
Test č. 2 po kapitole 10	101
Test č. 3 A po kapitole 11	102
Test č. 3 B po kapitole 11	104
Polročný test č. 4 A po kapitole 14	106
Polročný test č. 4 B po kapitole 14	108
Polročný test č. 5 A po kapitole 14	110
Polročný test č. 5 B po kapitole 14	112
Polročný test č. 5 C po kapitole 14	114
Polročný test č. 5 D po kapitole 14	116

Úvod

Elektronická kniha Programujeme v Pythone – Testy k učebnici je doplňujúcim materiálom k elektronickej učebnici informatiky pre stredné školy s názvom Programujeme v Pythone. Cieľom knihy je pomôcť učiteľom pri vyučovaní programovania v Pythone sadou cvičných i overovacích testov. Kniha obsahuje 44 praktických testov. Niektoré z testov sú pripravené v dvoch a niektoré aj vo viacerých variantoch. Jednotlivé varianty testu skúšajú rovnaké vedomosti a zručnosti a majú veľmi podobnú obťažnosť. V knihe je 24 testov, ak nepočítame ich jednotlivé varianty.

Testy sú odskúšané autorom na vyučovaní v 1. a 2. ročníku gymnázia. Každý test obsahuje informácie o odporúčanom čase riešenia, type testu, potrebných znalostiach, variante testu a jeho kóde.

Testy študenti riešia prakticky pri počítači v prostredí Python Idle. Svoje vypracované riešenie odovzdávajú elektronicky.

Testy možno používať na testovanie študentov, je možné ich modifikovať, prípadne sa inšpirovať úlohami pri vytváraní vlastného testu. Aj z tohto dôvodu je kniha prístupná v editovateľnej podobe. Ak však chceme, aby tento materiál bol použiteľný na testovanie, nezverejňujte samotné testy ani jednotlivé úlohy, ani úlohy, ktoré vznikli vašou modifikáciou týchto úloh.

Kniha je členená na dve časti. V prvej časti sa nachádzajú testy, ku ktorým je k dispozícii aj riešenie a tiež návrh bodového hodnotenia testu. V druhej časti sú ďalšie testy bez riešenia a hodnotenia.

Verím, že tento materiál vám pomôže pri vyučovaní programovania v Pythone. V prípade, že ešte nevyučujete Python, ma poteší, ak aj vďaka tomuto materiálu sa čoskoro rozhodnete vyučovať programovanie v tomto jazyku.

autor

Dávame do pozornosti

Stránky: <http://www.programujemevpythone.sk>

<https://www.facebook.com/programujemevpythone>.

Učiteľom dávame do pozornosti aj Klub učiteľov informatiky v Bratislave na stránke:

<http://www.1sg.sk/itklub>

Ak ešte nemáte samotnú učebnicu alebo si ju chcete objednať pre svojich študentov, môžete tak urobiť na stránke: <http://programujeme-v-pythone.webnode.sk/>

Dôležité informácie...

Nasledujúce informácie môžu byť pre vás užitočné na porozumenie štruktúry tejto knihy, značenia testov, ich účelu a pod.

Obsah učebnice vyučujeme v dvoch polrokoch pri jednohodinovej týždennej dotácii. Začíname v druhom polroku 1. ročníka gymnázia a pokračujeme v druhom polroku 2. ročníka gymnázia.

V knihe sú pripravené priebežné testy v dvoch úrovniach, ktoré zadávame v priebehu polroka a zväčša sa riešia 45-60 minút. Na záver polroka je pripravený polročný test. Tento test študent rieši 120 minút. Z týchto dôvodov sú testy označené ako priebežné a polročné. V knihe nájdeme:

- testy zadávané počas polroka po prebratí istých kapitol – kódy 51xy a 52xy pre prvý polrok s Pythonom a kódy 61xy a 62xy pre druhý polrok vyučovania Pythonu,
- polročné testy – používame ich na záver polroka (kódy 53xy a 63xy).

V značení kódu testu prvé dvojčísle označuje úroveň a posledné dvojčísle označuje číslo testu v danej úrovni (keďže je v knihe viac testov na rovnakých úrovniach). Na konci kódu môžu byť písmená A až D, ktoré označujú jednotlivé varianty rovnakých testov.

Potrebné znalosti na riešenie testu sú označené číslami kapitol, ktoré sú podľa značenia v učebnici Programujeme v Pythone. Ak teda má test označenie „po kapitole 7“, znamená to, že je potrebné mať prebraté učivo od začiatku až po kapitolu číslo 7 vrátane.

Testy hodnotíme úspešnosťou v percentách a podľa tejto stupnice určujeme známku:

85 % a viac	= 1
70 % až 84 %	= 2
50 % až 69 %	= 3
30 % až 49 %	= 4
0 % až 29 %	= 5

Samozrejme, že uvedená stupnica je len návrhom a slúži na vašu inšpiráciu. Je vyskúšaná autorom a používa ju v praxi.

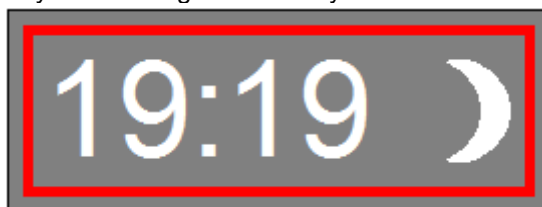
1. Testy s riešením a hodnotením

1. 1 Zadania testov

Test č. 1 A po kapitole 4.2	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5101A

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

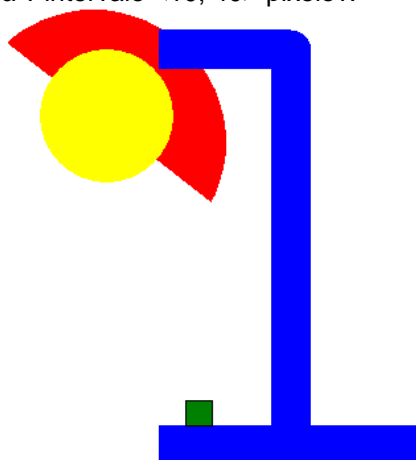
01 Vytvorte program, ktorý nakreslí digitálne hodiny.



- Čas na hodinách je náhodný v rozsahu 10 až 23 hodín (vrátane 10 aj 23),
- hodiny sú pokazené a vždy ukazujú rovnaký počet hodín aj počet minút,
- pozadie hodín je šedé, čas je biely, vnútorný rám je červený, vonkajší okraj je čierny,
- v pravej časti sa ukazuje fáza mesiaca.

02 Vytvorte program, ktorý pri hodinách nakreslí lampu.

- Stojan lampy je modrý,
- žiarovka je žltá,
- tienidlo lampy je červené,
- na podstave lampy je tlačidlo, ktorým sa ovláda intenzita svetla. Tlačidlo je zelené a jeho výška je náhodná v intervale <10; 40> pixelov.

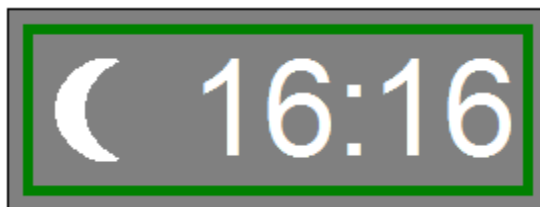


KONIEC TESTU

Test č. 1 B po kapitole 4.2	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5101B

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom ____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu ____.

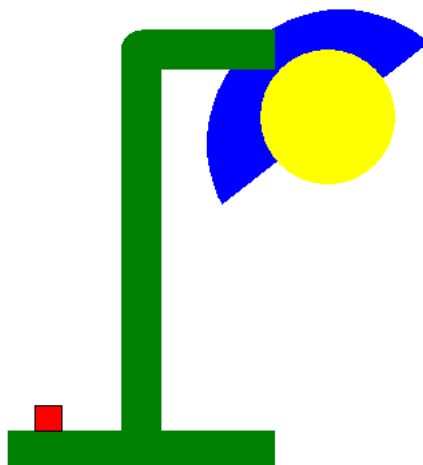
01 Vytvorte program, ktorý nakreslí digitálne hodiny.



- Čas na hodinách je náhodný v rozsahu 15 až 20 hodín (vrátane 12 aj 20),
- hodiny sú pokazené a vždy ukazujú rovnaký počet hodín aj počet minút,
- pozadie hodín je šedé, čas je biely, vnútorný rám je zelený, vonkajší okraj je čierny,
- v ľavej časti sa ukazuje fáza mesiaca.

02 Vytvorte program, ktorý pri hodinách nakreslí lampu.

- Stojan lampy je zelený,
- žiarovka je žltá,
- tienidlo lampy je modré,
- na podstave lampy je tlačidlo, ktorým sa ovláda intenzita svetla. Tlačidlo je červené a jeho výška je náhodná v intervale <10; 40> pixelov.

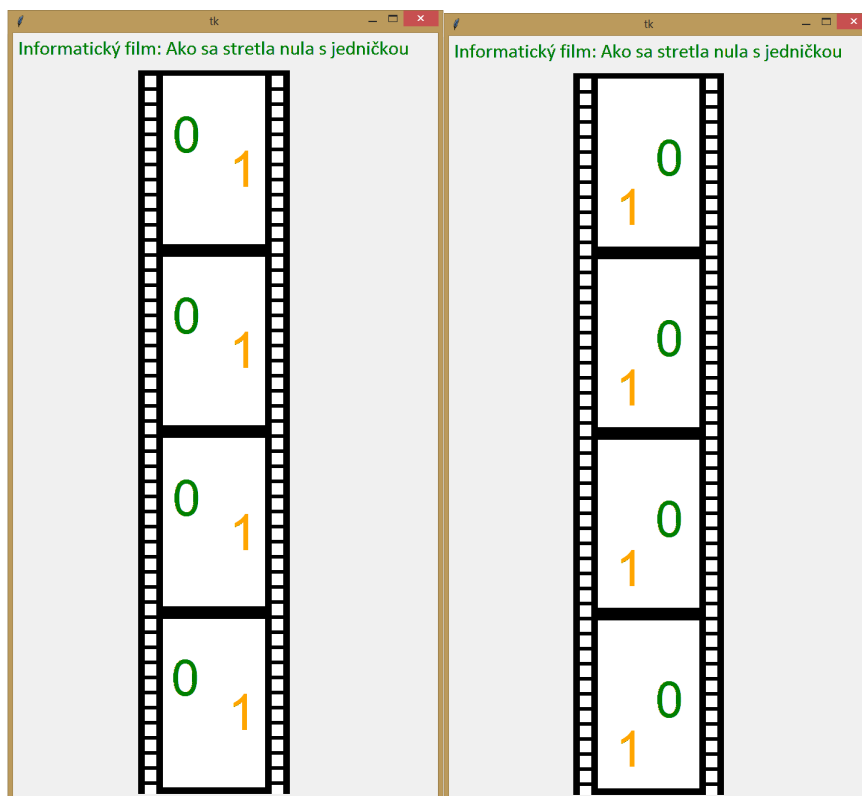


KONIEC TESTU

Test č. 1 A po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5201A

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

01 Vytvorte program, ktorý nakreslí informatický film.



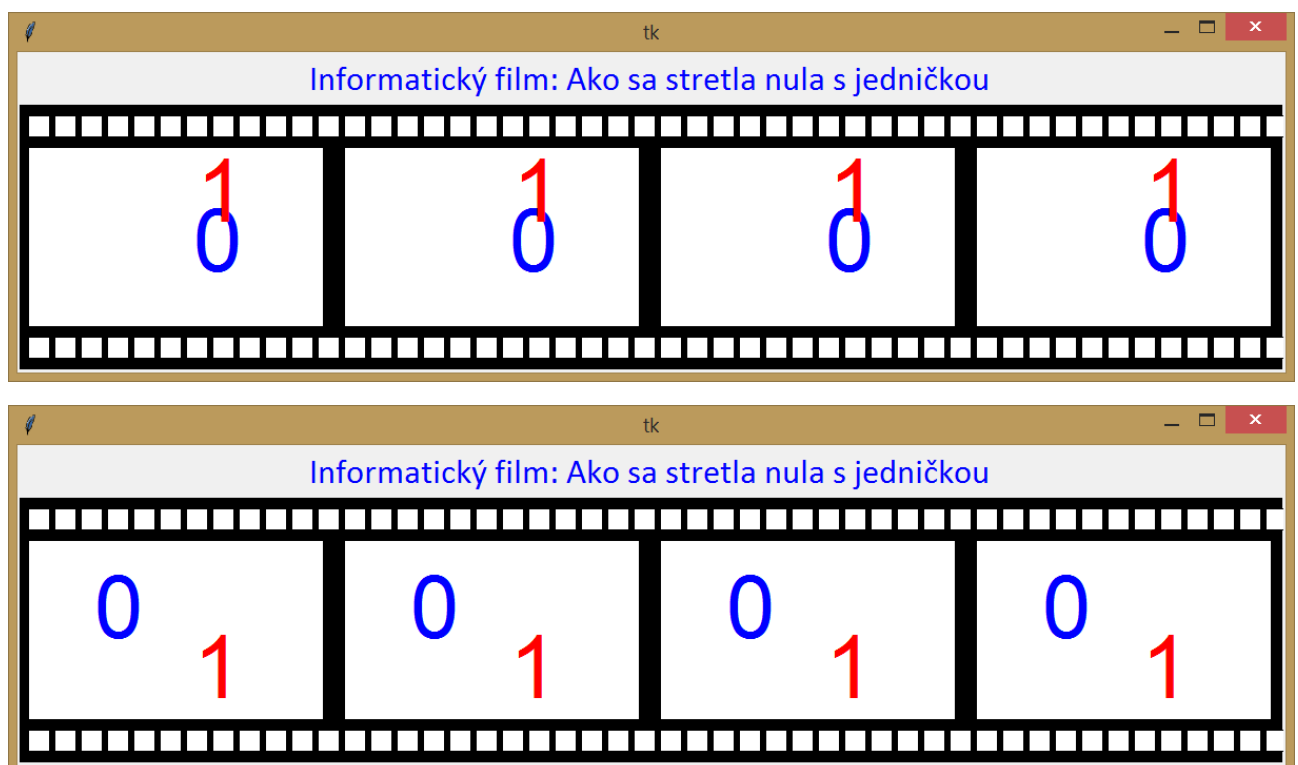
- V hornej časti je nápis „Informatický film: Ako sa stretla nula s jedničkou“, nápis je napísaný väčším písmom Arial zelenou farbou,
- pod nápisom je nakreslený film, ktorý má štyri biele políčka,
- po oboch zvislých okrajoch filmu sú nakreslené biele štvorčeky (perforácia filmu),
- na políčkach filmu sú veľkým písmom napísané nula a jednička, nula je zelená a jednička oranžová, pozícia nuly a jedničky je v rámci filmového políčka náhodná, ale ich pozície na všetkých štyroch políčkach sú rovnaké.
- Nuly a jedničky sa kreslia tak, aby neboli nakreslené mimo filmového políčka.
- Všade, kde je to vhodné použite cyklus.
- V programe vytvorte a rozumne použite aspoň jednu vlastnú funkciu.

KONIEC TESTU

Test č. 1 B po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5201B

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom ____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu ____.

01 Vytvorte program, ktorý nakreslí informatický film.



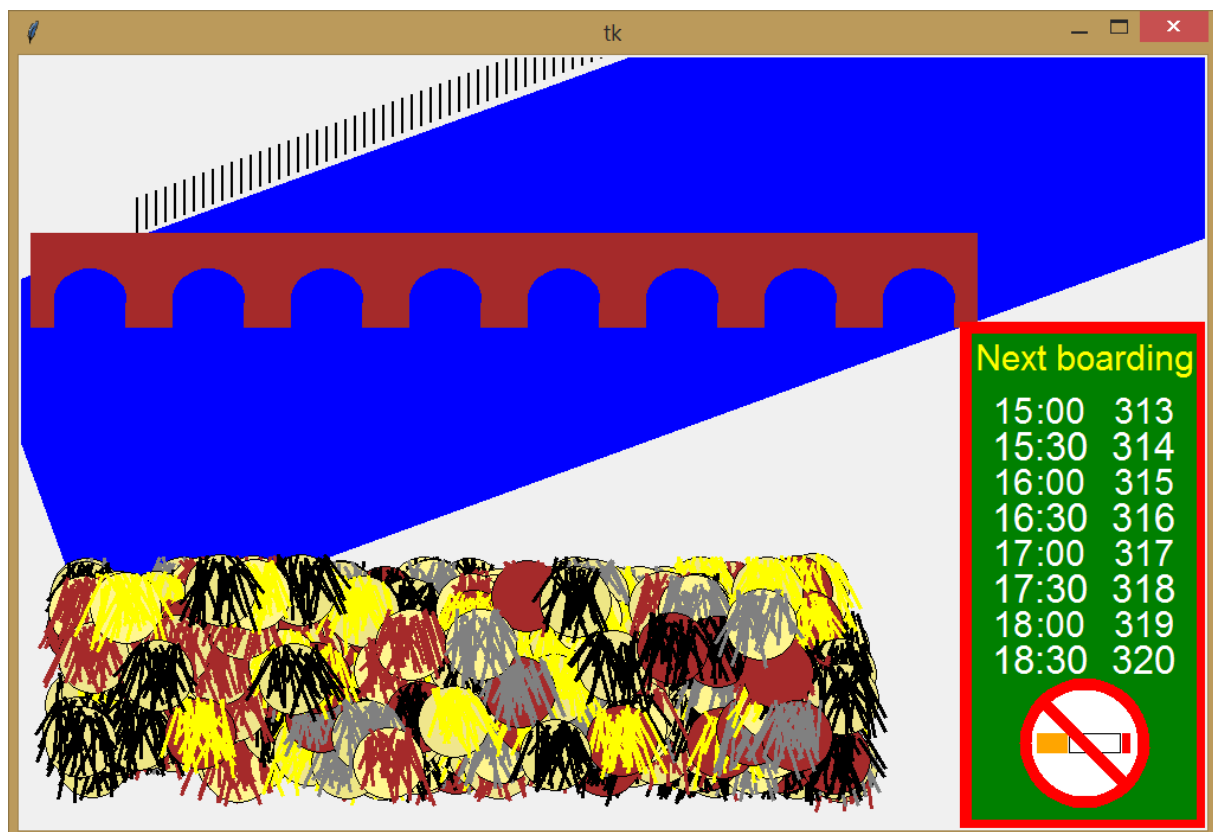
- V hornej časti je nápis „Informatický film: Ako sa stretla nula s jedničkou“, nápis je napísaný väčším písmom Arial modrou farbou,
- pod nápisom je nakreslený film, ktorý má štyri biele políčka,
- po oboch vodorovných okrajoch filmu sú nakreslené biele štvorčeky (perforácia filmu),
- na políčkach filmu sú veľkým písmom napísané nula a jednička, nula je modrá a jednička červená, pozícia nuly a jedničky je v rámci filmového políčka náhodná, ale ich pozície na všetkých štyroch políčkach sú rovnaké.
- Nuly a jedničky sa kreslia tak, aby neboli nakreslené mimo filmového políčka.
- Všade, kde je to vhodné, použite cyklus.
- V programe vytvorte a rozumne použite aspoň jednu vlastnú funkciu.

KONIEC TESTU

Polročný test č. 1 po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7 Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	-
Kód testu:	5301

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

01 V Pythone vytvorte program, ktorý nakreslí obrázok z výletu do Prahy.



- V programe sa snažte využiť for cyklus všade, kde je to možné. V programe vhodne použite aj vlastné vytvorené funkcie.
- Rozmer grafického plátna je 1000 x 650 bodov.
- V hornej časti je nakreslená rieka so zábradlím a mostom.
 - Most je hnedej farby a má osem oblúkov.
 - Rieka je šikmá, veľmi hrubá modrá čiara.
 - Severný breh rieky chráni zábradlie.
- V pravej dolnej časti je zobrazená informačná tabuľa, kde sú zobrazené najbližšie odchody vyhlídkových lodí.
 - Tabuľa je zelená a má hrubý červený okraj.
 - V hornej časti je žltý nadpis „Next boarding“.

- Na tabuli je zobrazených desať najbližších nalodení sa (odchodov). Odchody sú každú polhodinu a vždy sa na tabuli striedajú celé hodiny s polhodinami. Začína sa vždy celou hodinou. Prvá najbližšia hodina je náhodná z rozsahu 10:00 až 15:00. Všetky ďalšie časy odchodov idú chronologicky po pol hodine od prvej zobrazenej plavby.
- Vedľa časovej informácie je zobrazené číslo skupiny, ktorá sa naloží v tom čase. Číslo prvej skupiny je náhodné z rozsahu 100 až 500. Všetky ďalšie čísla sú označené nasledovným číslom.
- Informácie o vstupoch sú napísané bielou farbou.
- V spodnej časti informačnej tabule je zobrazený zákaz fajčenia.

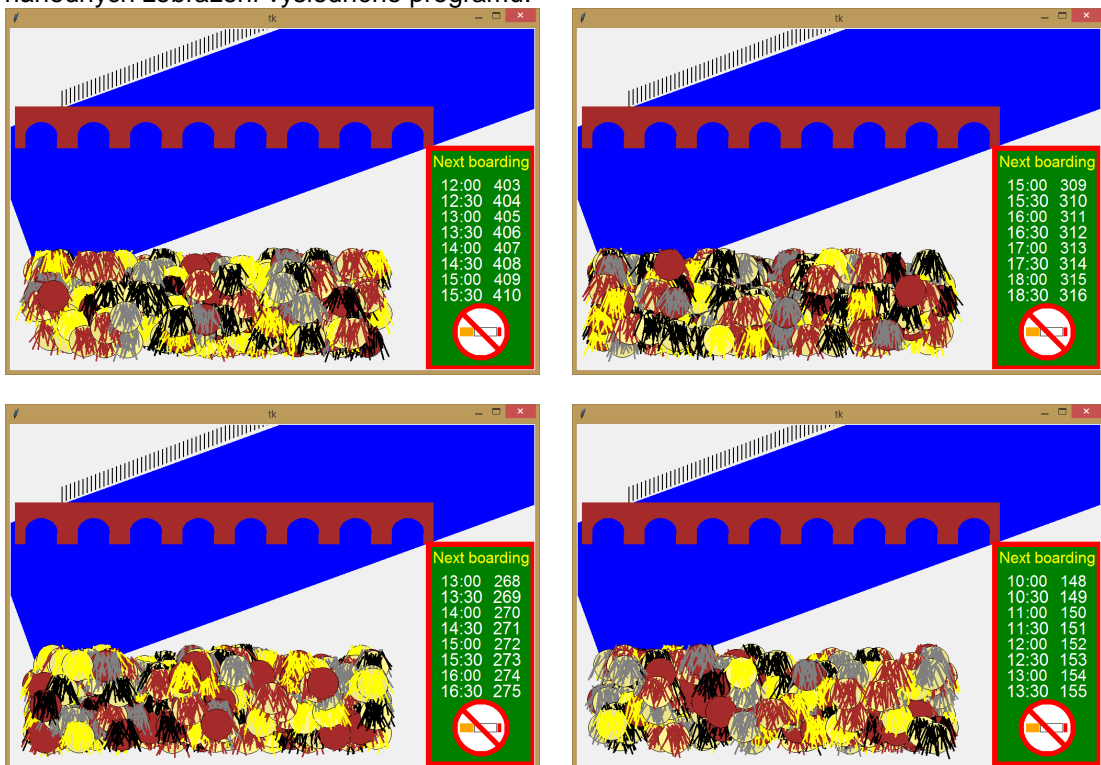


- V spodnej časti pred riekou je dav čakajúcich ľudí. Dav je v len v dolnej ohraničenej zóne. V dave je približne 300 ľudí.
- V dave sú vytŕčajúce hlavy. Hlavu môžeme nakresliť samostatnou funkciou, ktorej zadávame súradnice. Hlavy vyzerajú napríklad takto:



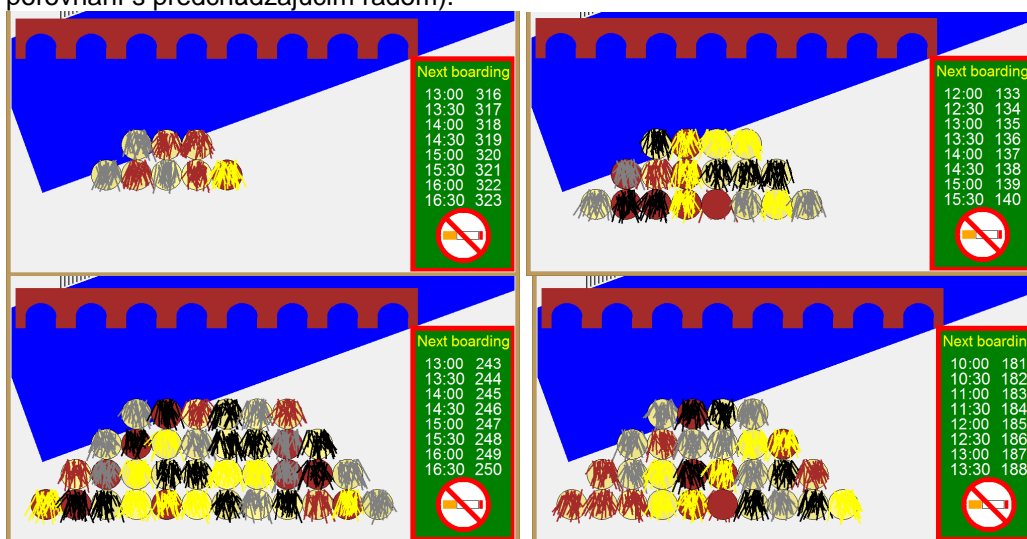
- Hlava je kruh s priemerom 60 bodov, náhodnej farby z týchto farieb: khaki, khaki1, khaki2 a brown.
- Na hlave je 30 náhodných vlasov. Vlasy začínajú v hornej časti v nejakom náhodnom úzkom obdĺžniku a končia v dolnej časti. Ich sklon je po nakreslení pod náhodným uhlom. Sklon je skôr zvislý ako vodorovný (pozrite si obrázok).
- Vlasy sú hrubšie a ich farba je náhodná z týchto farieb: žltá, hnedá, šedá a čierna.

Ukážky náhodných zobrazení výsledného programu:

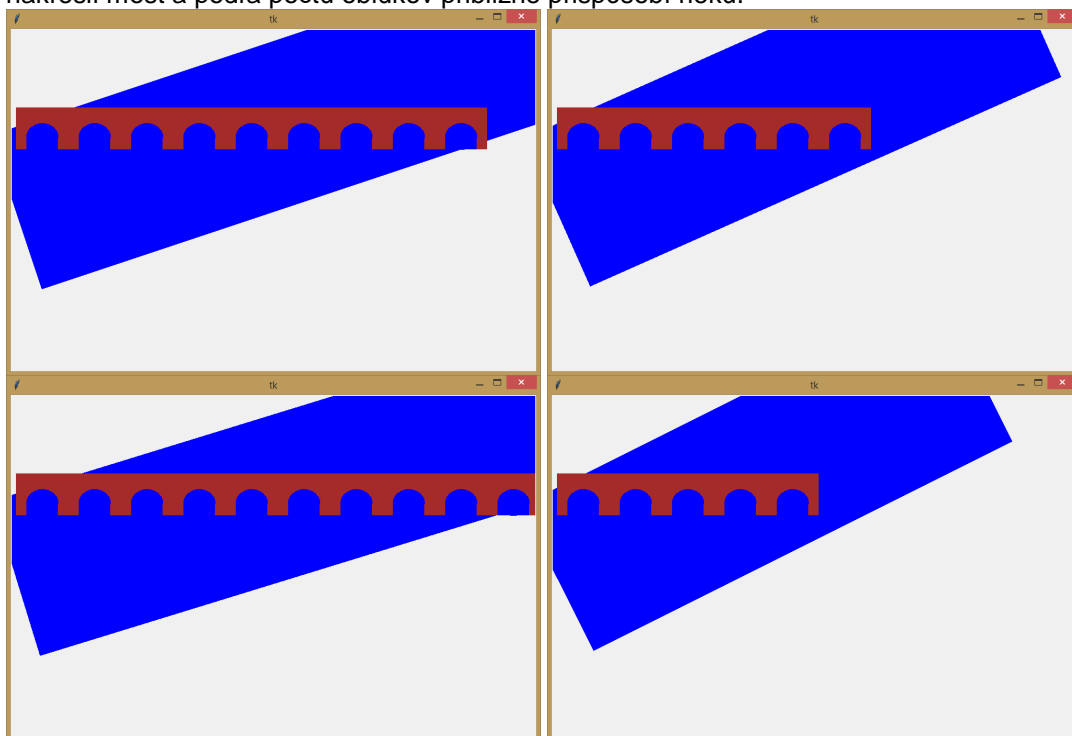


Prémia 1 Vytvorte funkciu **Prémia1**, ktorá nakreslí pred riekou skupinu ľudí (pripomínajúcu časť pyramídy). Funkcii zadáme počet ľudí v prvom rade a počet radov. V prvom rade funkcia

nakreslí zadaný počet ľudí. V každom ďalšom rade pribudnú ľudia na oboch okrajoch (v porovnaní s predchádzajúcim radom).



Prémia 2 Vytvorte funkciu **Prémia2**, ktorá na zadanom mieste a so zadaným počtom oblúkov nakreslí most a podľa počtu oblúkov približne prispôbí rieku.

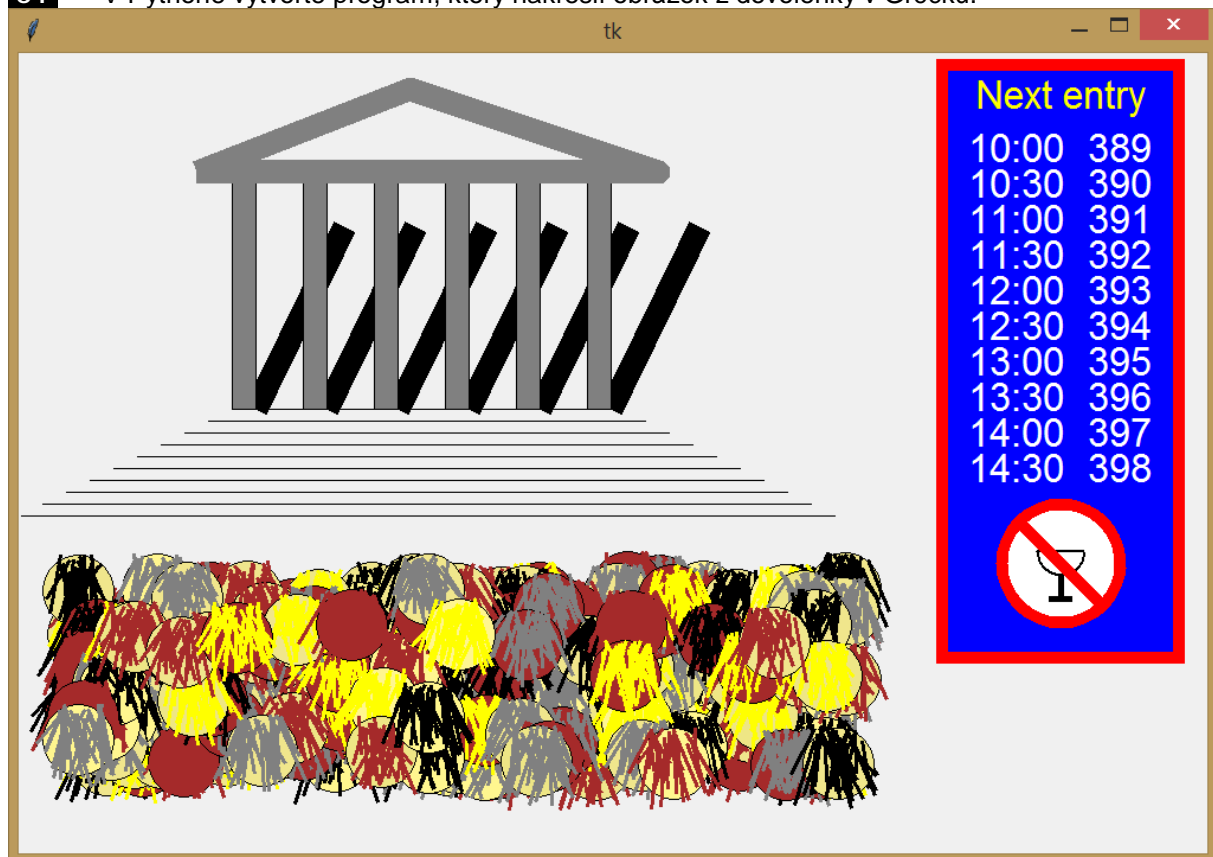


KONIEC TESTU

Polročný test č. 2 po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7 Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	-
Kód testu:	5302

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

01 V Pythone vytvorte program, ktorý nakreslí obrázok z dovolenky v Grécku.



- V programe sa snažte využiť for cyklus všade, kde je to možné. V programe vhodne použite aj vlastné vytvorené funkcie.
- Rozmer grafického plátna je 1000 x 650 bodov.
- V hornej časti je nakreslený **chrám**.
 - Chrám je šedej farby a má šesť stĺpov.
 - Stĺpy vrhajú tieň. Sklon tieňa a jeho dĺžka sú náhodné, ale v smere doprava a hore.
 - Pred chrámom je schodisko z desiatich schodov.
- V pravej časti je zobrazená informačná tabuľa, kde sú zobrazené najbližšie prehliadky chrámu.
 - Tabuľa je modrá a má hrubý červený okraj.
 - V hornej časti je žltý nadpis „Next entry“.

- Na tabuli je zobrazených desať najbližších vstupov. Vstupy sú každú polhodinu a vždy sa na tabuli striedajú celé hodiny s polhodinami. Začína sa vždy celou hodinou. Prvá najbližšia hodina je náhodná z rozsahu 10:00 až 15:00. Všetky ďalšie časy prehliadky idú chronologicky po pol hodine od prvej zobrazenej prehliadky.
- Vedľa časovej informácie je zobrazené číslo skupiny, ktorá navštívi v tom čase chrám. Číslo prvého vstupu je náhodné z rozsahu 100 až 500. Všetky ďalšie čísla sú označené nasledovným číslom.
- Informácie o vstupoch sú napísané bielou farbou.
- V spodnej časti informačnej tabule je zobrazený zákaz vstupu s nápojmi.

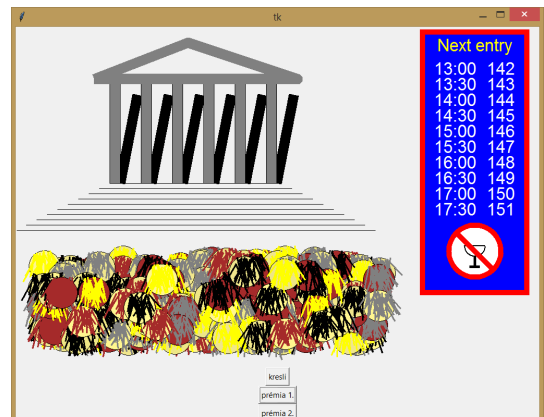
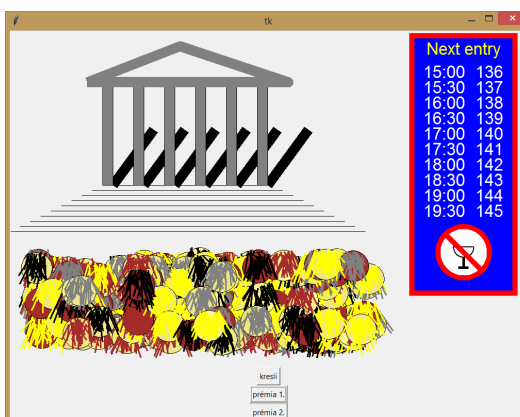
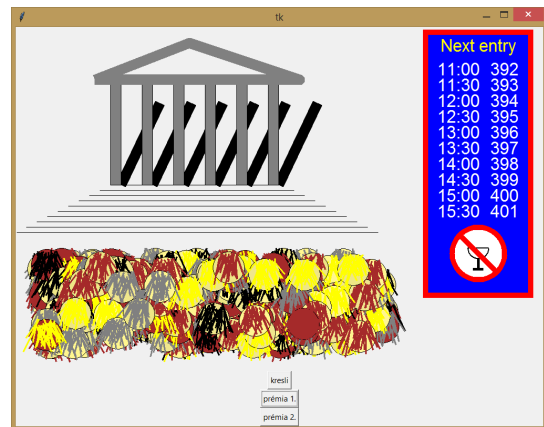
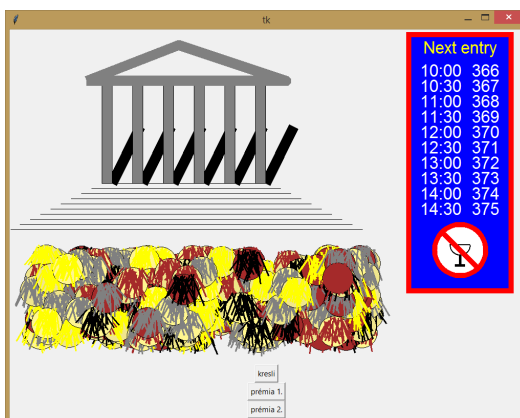


- V spodnej časti pod chrámom je dav čakajúcich ľudí. Dav je v len v dolnej ohraničenej zóne. V dave je približne 300 ľudí.
- V dave sú vytŕčajúce hlavy. Hlavu môžeme nakresliť samostatnou funkciou, ktorej zadávame súradnice. Hlavy vyzerajú napríklad takto:

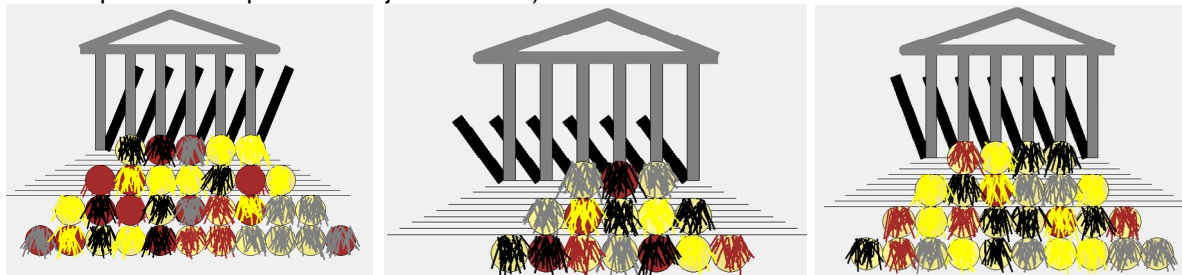


- Hlava je kruh s priemerom 60 bodov, náhodnej farby z týchto farieb: khaki, khaki1, khaki2 a brown.
- Na hlave je 30 náhodných vlasov. Vlasy začínajú v hornej časti v nejakom náhodnom úzkom obdĺžniku a končia v dolnej časti. Ich sklon je po nakreslení pod náhodným uhlom. Sklon je skôr zvislý ako vodorovný (pozrite si obrázok).
- Vlasy sú hrubšie a ich farba je náhodná z týchto farieb: žltá, hnedá, šedá a čierna.

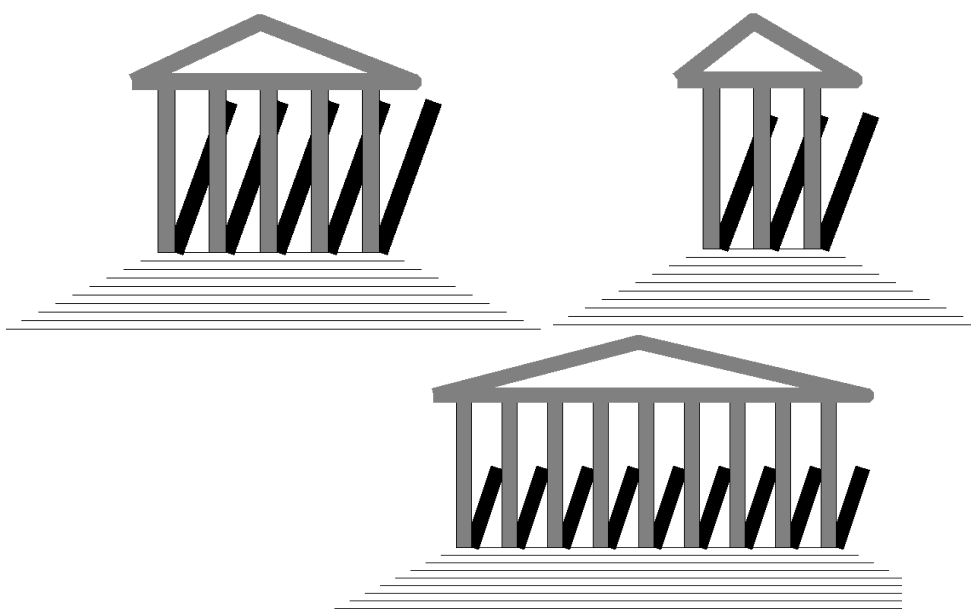
Ukážky náhodných zobrazení výsledného programu:



Prémia 1 Vytvorte funkciu **Prémia1**, ktorá nakreslí pred chrámom skupinu ľudí (pripomínajúcu časť pyramídy). Funkcii zadáme počet ľudí v prvom rade a počet radov. V prvom rade funkcia nakreslí zadaný počet ľudí. V každom ďalšom rade pribudnú ľudia na oboch okrajoch (v porovnaní s predchádzajúcim radom).



Prémia 2 Vytvorte funkciu **Prémia2**, ktorá na zadanom mieste a so zadaným počtom stĺpov nakreslí chrám.



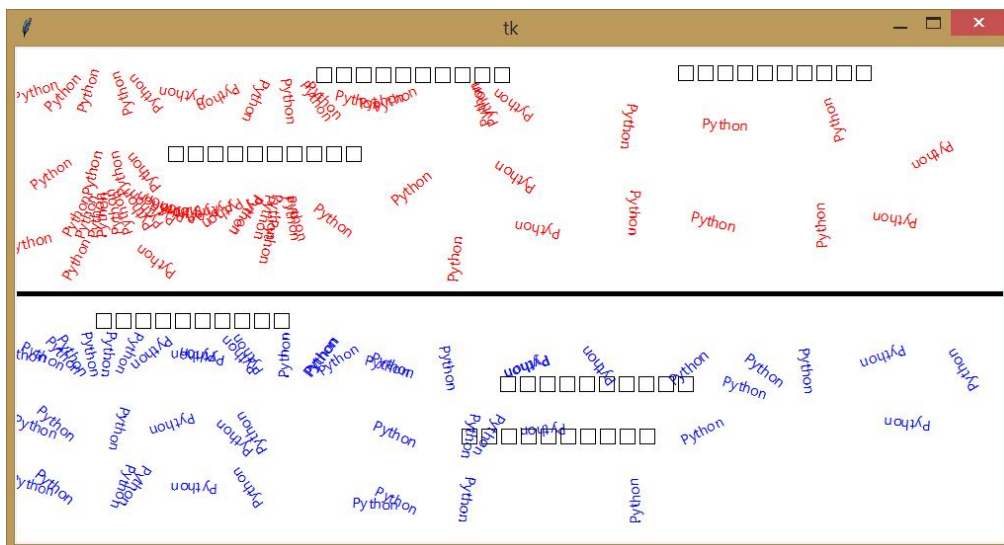
KONIEC TESTU

Test č. 1 A po kapitole 10	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	6101A

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať. Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.**

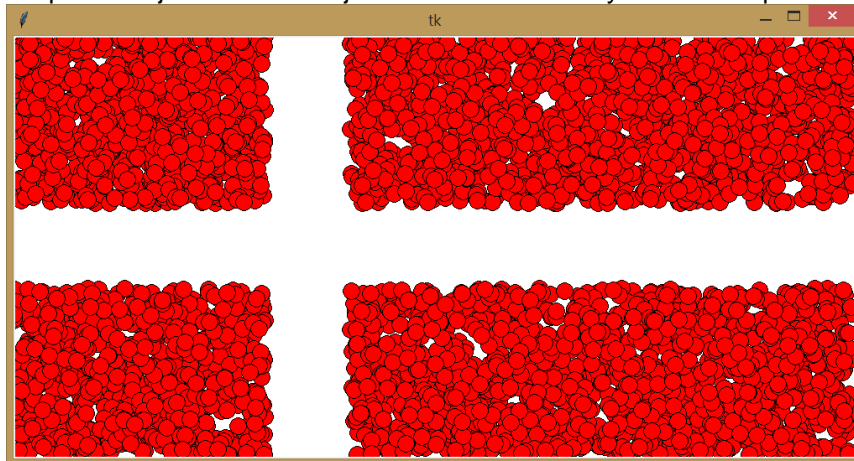
01 Vytvorte program, ktorý:

- po spustení vytvorí grafické plátno s veľkosťou 1000x500 pixelov s bielym pozadím,
- v strede plátna bude vodorovná čierna čiara s hrúbkou 5 pixelov,
- po kliknutí ľavého tlačidla myši v hornej polovici plátna sa na mieste kliknutia napíše text **Python**, ktorý spĺňa tieto parametre:
 - je otočený **v protismere** hodinových ručičiek,
 - uhol otočenia je rovnaký, ako x-ová súradnica miesta kliknutia,
 - text je červený,
- po kliknutí ľavého tlačidla myši v dolnej polovici plátna sa na mieste kliknutia napíše text **Python**, ktorý spĺňa tieto parametre:
 - je otočený **v smere** hodinových ručičiek,
 - uhol otočenia je rovnaký, ako x-ová súradnica miesta kliknutia,
 - text je modrý,
- po kliknutí pravého tlačidla myši program nakreslí na mieste kliknutia desať štvorčekov s veľkosťou 15 pixlov vedľa seba, štvorčeky budú oddelené medzerou s veľkosťou 5px.



02

Vytvorte program, ktorý postupne pomocou časovača nakreslí v grafickom plátne s veľkosťou 1000x500 pixlov vlajku Dánska. Vlajka sa skladá z červených kruhov s priemerom 20 px.



Prémia 1

Upravte 1. úlohu tak, aby sme si čiaru v strede obrazovky mohli posúvať hore a dole stlačením šípky hore a dole. Po stlačení šípky sa obsah obrazovky zmaže, nakreslí sa čiara na novom mieste a kliknutie myši funguje podľa novej čiary.

Prémia 2

Upravte 1. úlohu tak, aby sme pomocou klávesnice mohli nastaviť, koľko štvorčekov sa nakreslí. Či 7, 8 alebo 9 (podľa toho, ktorú klávesu stlačíme – 7, 8 alebo 9). Keď sme nič nestlačili, štandardne sa nakreslí 10 štvorčekov.

Prémia 3

Upravte 2. úlohu tak, aby sme pomocou klávesu + a - mohli nastaviť polomer kruhov. Minimálna veľkosť polomeru bude 3 px a maximálna veľkosť bude 30 pixelov. Niekde na obrazovke (napríklad v bielom pruhu vlajky) budeme vidieť napísanú aktuálnu veľkosť priemeru a počet nakreslených kruhov.

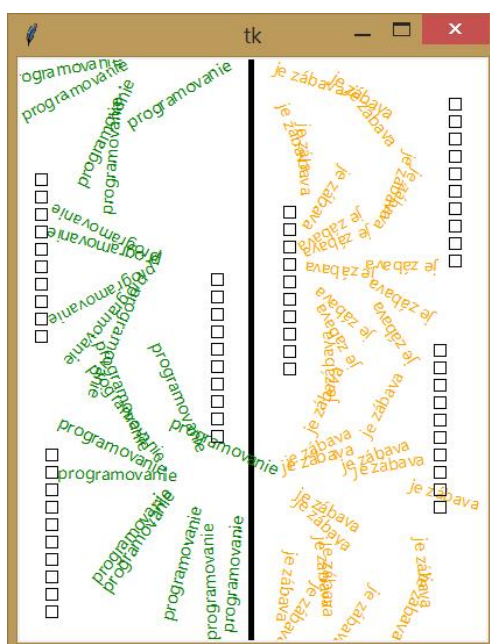
KONIEC TESTU

Test č. 1 B po kapitole 10	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	6101B

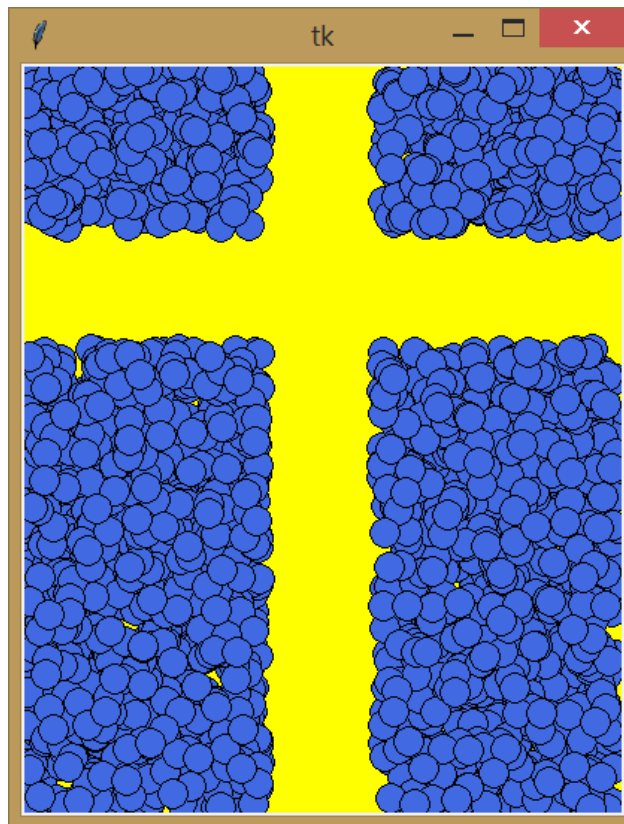
Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

01 Vytvorte program, ktorý má tieto vlastnosti:

- po spustení vytvorí grafické plátno s veľkosťou 400x500 pixelov s bielym pozadím,
- v strede plátna bude zvislá čierna čiara s hrúbkou 5 pixelov,
- po kliknutí ľavého tlačidla myši v ľavej polovici plátna sa na mieste kliknutia napíše text **programovanie**, ktorý spĺňa tieto parametre:
 - je otočený **v protismere** hodinových ručičiek,
 - uhol otočenia je rovnaký ako y-ová súradnica miesta kliknutia,
 - text je zelený,
- po kliknutí ľavého tlačidla myši v pravej polovici plátna sa na mieste kliknutia napíše text **je zábava**, ktorý spĺňa tieto parametre:
 - je otočený **v smere** hodinových ručičiek,
 - uhol otočenia je rovnaký ako y-ová súradnica miesta kliknutia,
 - text je oranžový,
- po kliknutí pravého tlačidla myši program nakreslí na mieste kliknutia desať štvorčekov s veľkosťou 10 pixlov pod seba, štvorčeky budú oddelené medzerou s veľkosťou 5px.



- 02** Vytvorte program, ktorý postupne pomocou časovača nakreslí v grafickom plátne so žltým pozadím veľkosťou 400x500 pixlov vlajku Švédska. Vlajka sa skladá z kruhov s priemerom 20 px, ktoré sa kreslia farbou royalblue.



Prémia 1 Upravte 1. úlohu tak, aby sme si čiaru v strede obrazovky mohli posúvať vľavo a vpravo stlačením šípky vľavo a vpravo. Po stlačení šípky sa obsah obrazovky zmaže, nakreslí sa čiara na novom mieste a kliknutie myši funguje podľa novej čiary.

Prémia 2 Upravte 1. úlohu tak, aby sme pomocou klávesnice mohli nastaviť, koľko štvorčekov sa nakreslí. Či 7, 8 alebo 9 (podľa toho, ktorú klávesu stlačíme – 7, 8 alebo 9). Keď sme nič nestlačili, štandardne sa nakreslí 10 štvorčekov.

Prémia 3 Upravte 2. úlohu tak, aby sme pomocou klávesu + a - mohli nastaviť polomer kruhov. Minimálna veľkosť polomeru bude 3 px a maximálna veľkosť bude 30 pixelov. Niekde na obrazovke (napríklad v žltom pruhu vlajky) budeme vidieť napísanú aktuálnu veľkosť priemeru a počet nakreslených kruhov.

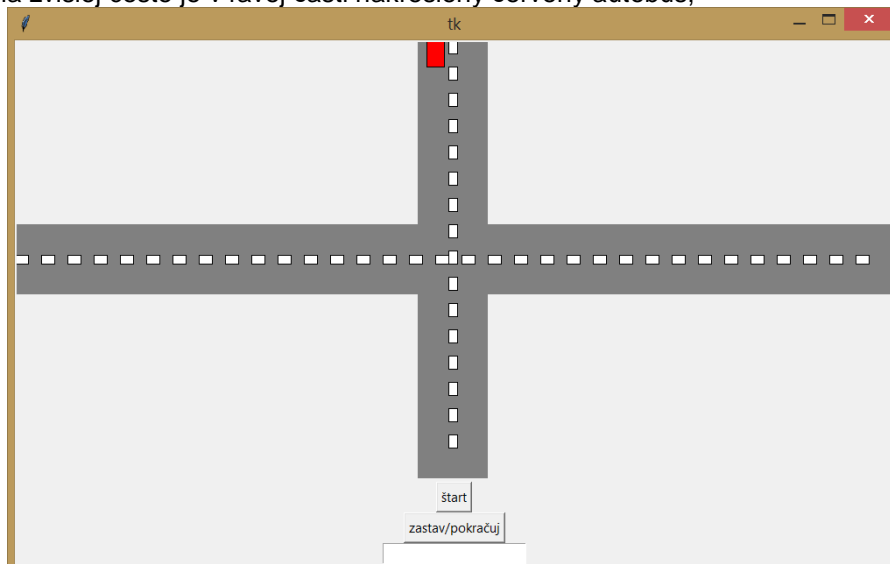
KONIEC TESTU

Test č. 1 A po kapitole 12	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu
Čas na vypracovanie:	60 minút
Variant:	A
Kód testu:	6201A

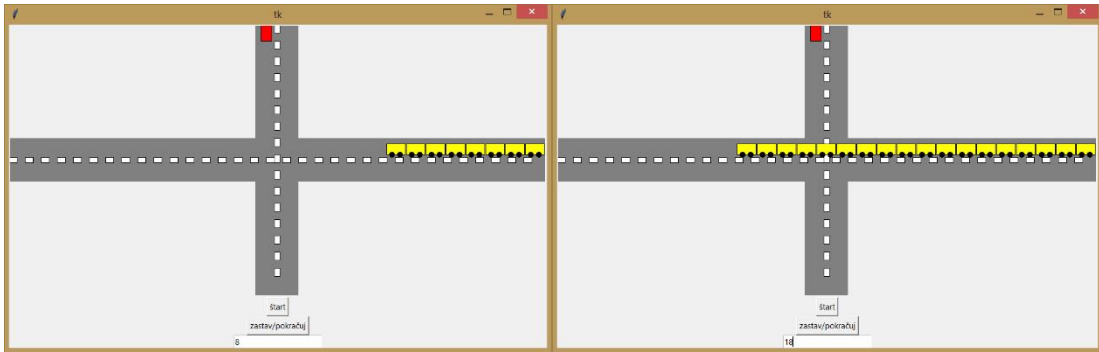
Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať. Na disku ___ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladáajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.**

01 Vytvorte program, ktorý má tieto vlastnosti:

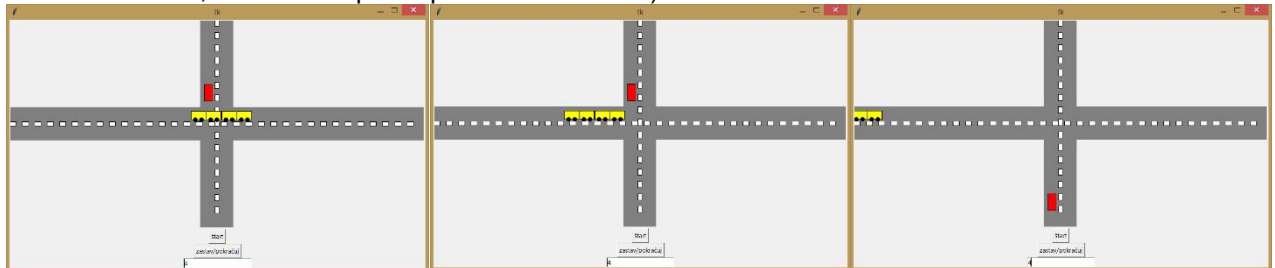
- po spustení vytvorí grafické plátno s veľkosťou 1000x500 pixelov,
- v programe budú dve tlačidlá – tlačidlo štart a tlačidlo zastav/pokračuj,
- v programe bude jedno vstupné pole (entry),
- po spustení program nakreslí presne v strede canvasu vodorovnú cestu a zvislú cestu,
- cesty sa križujú presne v strede canvasu, majú šedý podklad a v strede ciest je nakreslená prerušovaná biela čiara (vodorovné dopravné značenie), jeden dielik čiary má dĺžku presne 15 px a šírku 10 px, medzera medzi nimi je 15 px,
- na zvislej ceste je v ľavej časti nakreslený červený autobus,



- do vstupného pola môžeme zadať počet vagónov turistického vláčika, ktorý voží turistov na prehliadke mesta,
- po stlačení tlačidla štart sa na vodorovnej ceste v hornom pruhu nakreslí žltý turistický vláčik, vláčik končí úplne na pravom okraji obrazovky, počet vagónov sme zadali do vstupného pola,



- tlačidlom štart zároveň odštartujeme pohyb vláčika a autobusu,
- autobus sa pohybuje smerom dole, keď príde na okraj canvasu, ukáže sa opäť na hornom okraji,
- vláčik sa pohybuje smerom vľavo, keď posledný vozeň opustí obrazovku, ukáže sa prvý vozeň z pravého okraja canvasu,
- červený autobus nie je na hlavnej ceste, a tak zakaždým dáva prednosť v jazde vláčiku, čiže ak je to potrebné, pred križovatkou zastane a počká, kým bude môcť prejsť cez križovátku,
- v programe máme aj tlačidlo, ktoré zastaví animáciu alebo pustí jej pokračovanie (podľa toho, či animácia práve prebieha alebo nie).



Prémia 1 Doplňte do programu na kraj vodorovnej cesty pred križovatkou semafor, na ktorom svieti len červená alebo zelená farba. Kliknutím myši do canvasu môžeme prepnúť farbu na semafore. Vlak v prípade, že svieti červená, zastane pred križovatkou. V takejto situácii autobus nebude pred križovatkou čakať.

Prémia 2 Doplňte do programu zelené auto, ktoré ide po zvislej ceste zdola hore a má v križovatke rovnaké správanie ako červené auto.

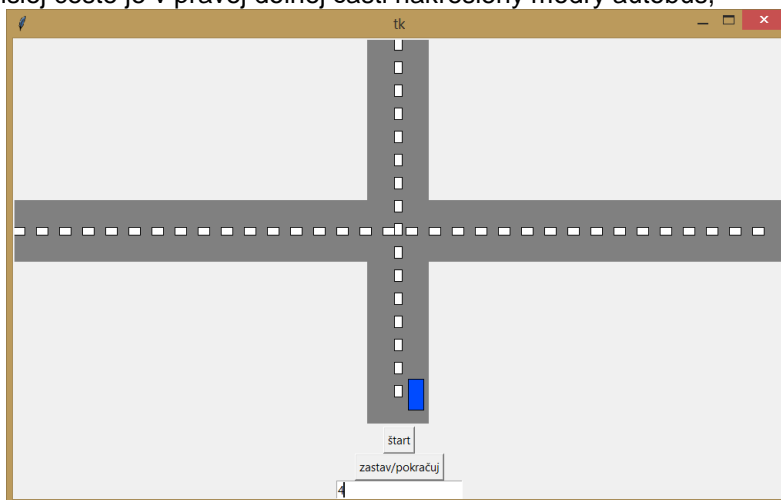
KONIEC TESTU

Test č. 1 B po kapitole 12	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu
Čas na vypracovanie:	60 minút
Variant:	B
Kód testu:	6201B

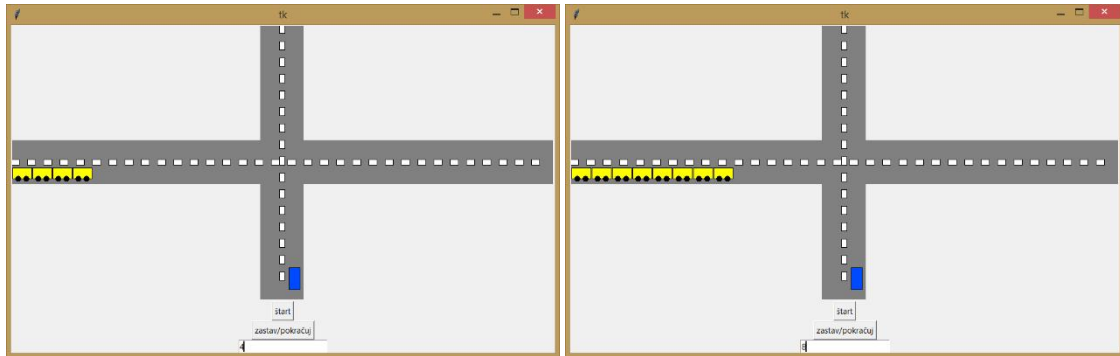
Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vaším menom. Na záver súbor pošlite mailom na adresu _____.

01 Vytvorte program, ktorý má tieto vlastnosti:

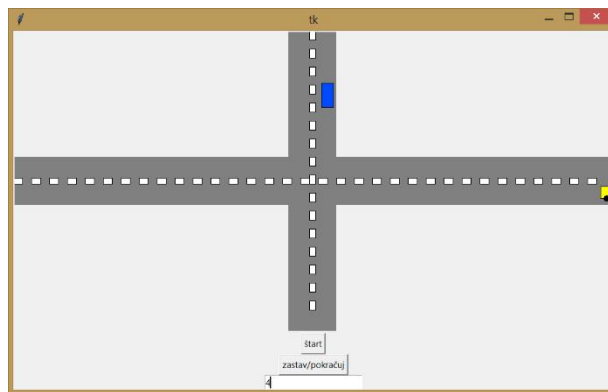
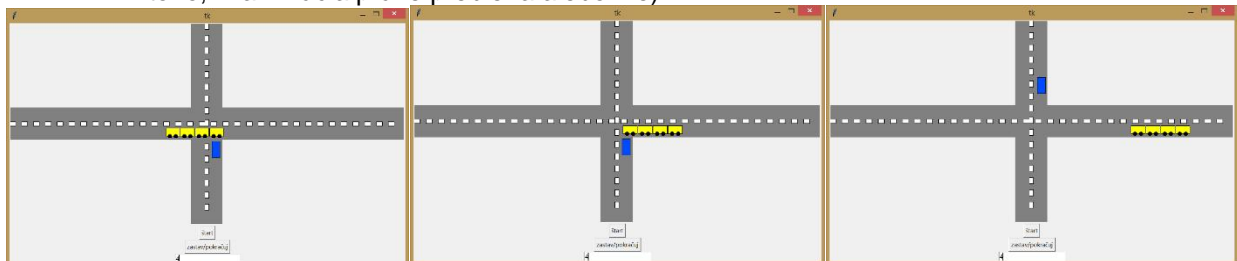
- po spustení vytvorí grafické plátno s veľkosťou 1000x500 pixelov,
- v programe budú dve tlačidlá – tlačidlo štart a tlačidlo zastav/pokračuj,
- v programe bude jedno vstupné pole (entry),
- po spustení program nakreslí presne v strede canvasu vodorovnú cestu a zvislú cestu,
- cesty sa križujú presne v strede canvasu, majú šedý podklad a v strede ciest je nakreslená prerušovaná biela čiara (vodorovné dopravné značenie), jeden dielik čiary má dĺžku presne 15 px a šírku 10 px, medzera medzi nimi je 15 px,
- na zvislej ceste je v pravej dolnej časti nakreslený modrý autobus,



- do vstupného pola môžeme zadať počet vagónov turistického vláčika, ktorý vozí turistov na prehliadku mesta,
- po stlačení tlačidla štart sa na vodorovnej ceste v dolnom pruhu nakreslí žltý turistický vláčik, vláčik začína úplne na ľavom okraji obrazovky, počet vagónov sme zadali do vstupného pola,



- tlačidlom štart zároveň odštartujeme pohyb vláčika a autobusu,
- autobus sa pohybuje smerom hore, keď príde na okraj canvasu, ukáže sa opäť na dolnom okraji,
- vláčik sa pohybuje smerom vpravo, keď posledný vozeň opustí obrazovku, ukáže sa prvý vozeň z ľavého okraja canvasu,
- modrý autobus nie je na hlavnej ceste, a tak zakaždým dáva prednosť v jazde vláčiku, čiže ak je to potrebné, pred križovatkou zastane a počká, kým bude môcť prejsť cez križovátku,
- v programe máme aj tlačidlo, ktoré zastaví animáciu alebo pustí jej pokračovanie (podľa toho, či animácia práve prebieha alebo nie).



Prémia 1 Doplňte do programu na kraj vodorovnej cesty pred križovatkou semafor, na ktorom svieti len červená alebo zelená farba. Kliknutím myši do canvasu môžeme prepnúť farbu na semafore. Vlak v prípade, že svieti červená, zastane pred križovatkou. V takejto situácii autobus nebude pred križovatkou čakať.

Prémia 2 Doplňte do programu zelené auto, ktoré ide po zvislej ceste zhora dole a má v križovátke rovnaké správanie ako modré auto.

KONIEC TESTU

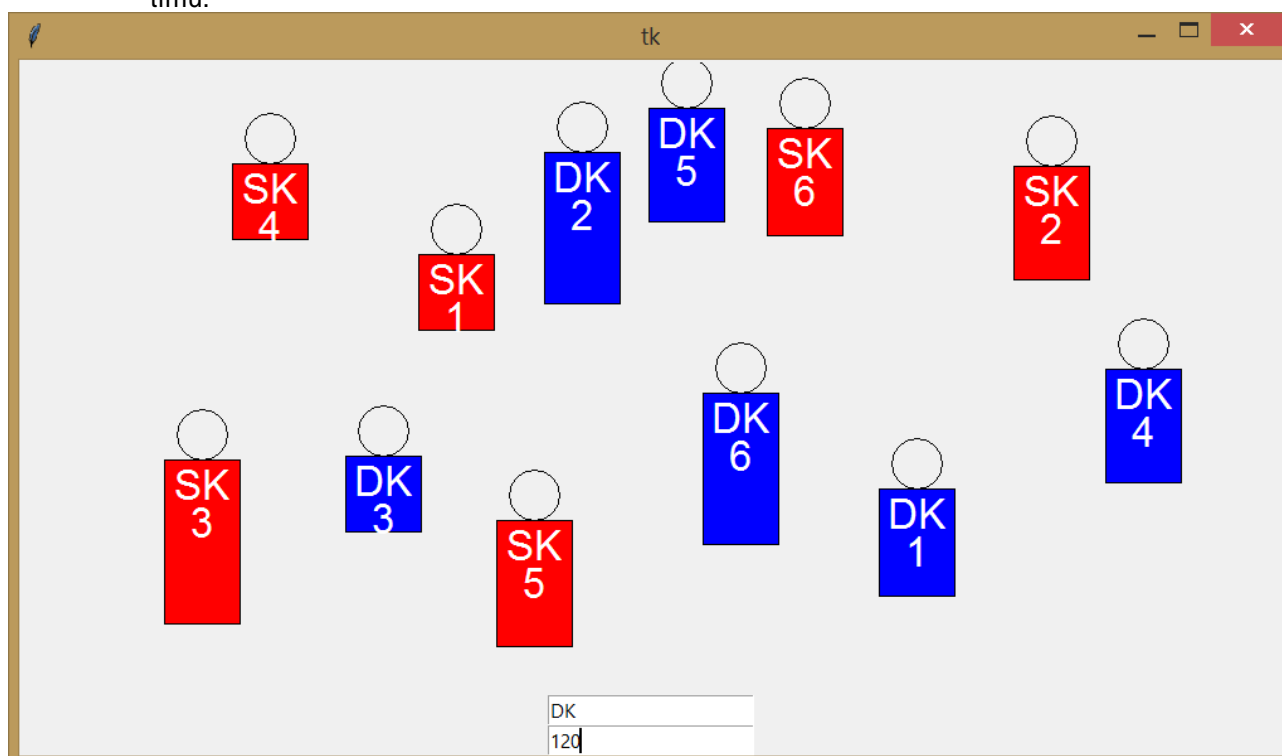
Test č. 2 po kapitole 11	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole
Čas na vypracovanie:	45 minút
Variant:	-
Kód testu:	6202

Všeobecné pokyny: Test riešite na počítači v prostredí IDLE Python 3.5. **Nezabudnite si úlohy priebežne ukladať.** Na disku ____ si vytvorte priečinok s názvom _____ a do neho si priebežne ukladajte svoj súbor s vašim menom. Na záver súbory pošlite mailom na adresu _____.

01

Vytvorte program, ktorý nám umožňuje navrhnuť rozmiestnenie hráčov hokeja:

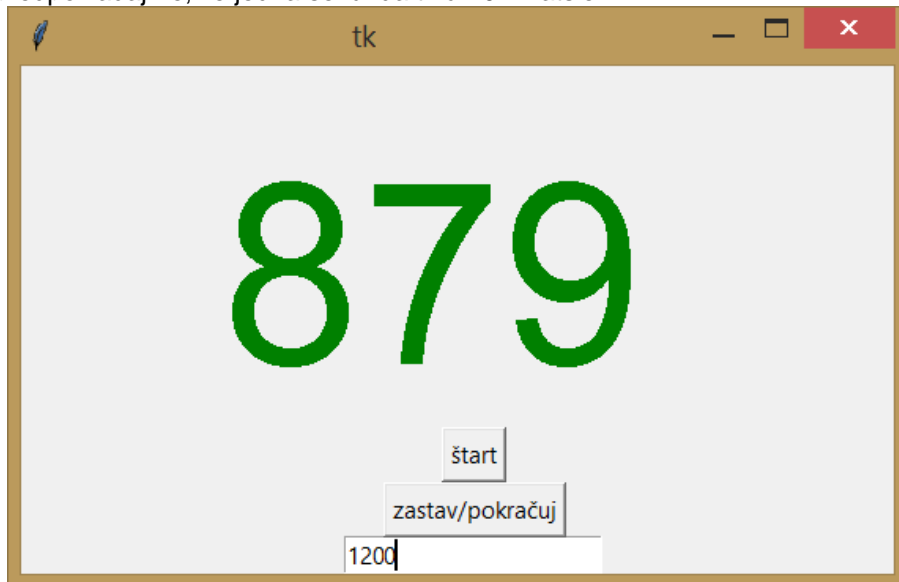
- po spustení vytvorí grafické plátno s veľkosťou 1000x500 pixelov,
- v programe budú dve vstupné polia (entry),
- v jednom vstupnom poli zadávame skratku krajiny,
- v druhom vstupnom poli zadávame dĺžku dresu hráča,
- kliknutím ľavého tlačidla myši umiestnime na mieste kliknutia hráča červeného družstva,
- kliknutím pravého tlačidla myši umiestnime na mieste kliknutia hráča modrého družstva,
- hráči majú na drese napísanú skratku krajiny, ktorá je uvedená vo vstupnom poli, skratka je nakreslená väčším písmom bielej farby,
- pod skratkou krajiny majú hráči na drese napísané svoje číslo,
- hráči sú číslovaní postupne číslami 1 až 6 (každý tím zvlášť),
- zaberte, aby nám program nedovolil na ihrisko umiestniť viac ako 6 hráčov jedného tímu.



02

Vytvorte program stopky, ktorý funguje nasledovne:

- Vo vstupnom poli (entry) zadáme číslo. Toto číslo určuje, koľko sekúnd chceme na stopkách odpočítavať. Napríklad, keď tam nastavíme číslo 1250, stopky budú odpočítavať 1250 sekúnd (postupne k nule).
- V programe sú tlačidlá (tlačidlo – vyberte si, koľko ich potrebujete), ktorými môžeme stopky naštartovať, pozastaviť a spustiť pokračovanie.
- Čas sa veľkým písmom zelenej farby zobrazuje v grafickej ploche.
- Po vypršaní času sa vypíše na obrazovku oznam „Čas vypršal“.
- Čas nemusí ísť presne v sekundách, aby ste mali možnosť ho rýchlo odskúšať, predpokladajme, že jedna sekunda trvá 10x kratšie.



KONIEC TESTU

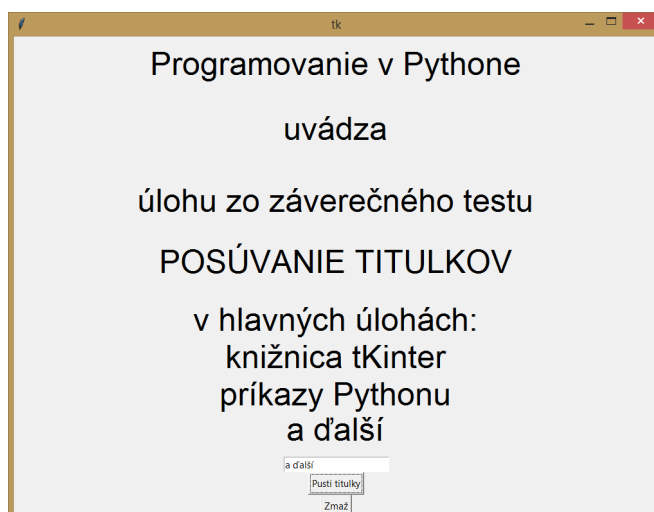
Polročný test č. 1 po kapitole 14	
Typ testu:	Polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Varianta:	-
Kód testu:	6301

Všeobecné pokyny: Test riešite na počítači v jazyku Python. Žiadne pomôcky nie sú povolené. Pracujte pokojne a s rozvahou. Svoju prácu si **priebežne ukladajte** na svoj disk _____ do priečinku _____. Súbory v tomto priečinku tiež pomenujte vašim priezviskom a číslom úlohy. V závere všetky súbory z priečinka pošlite v prílohe e-mailu na adresu _____.

01

Vytvorte program na písanie titulkov do filmu.

- Rozmer grafického plátna je 1000 x 650 bodov.
- V programe je súčiastka na zadanie vstupu a dve tlačidlá (Pusti titulky a Zmaž).
- Do vstupného okienka môžeme napísať text titulku.
- Kliknutím ľavého tlačidla myši umiestnime do grafickej plochy text, ktorý je práve napísaný v súčiastke na zadávanie vstupu. Text sa napíše vo výške kliknutia myši, ale horizontálne je vždy vycentrovaný vzhľadom na veľkosť grafického plátna.
- Text je napísaný čiernou farbou, písmom Arial s väčšou veľkosťou.
- Po naklikaní titulkov môže vyzeráť obrazovka napríklad takto (podľa toho, aké texty sme napísali):



- Po stlačení tlačidla Pusti titulky sa titulky strihom (ihneď) presunú pod spodný okraj grafickej plochy, aby boli pripravené na animáciu a následne sa pomaly presúvajú smerom hore. Keď z obrazovky odíde celá pôvodná obrazovka (aj keď nie je celá zapísaná), znovu sa tie isté titulky zobrazujú od spodného okraja.
- Priebežne môžeme myšou doklikávať ďalšie titulky, no pri doklikávaní nám to môže pokaziť ich odchod a príchod celej jednej obrazovky.

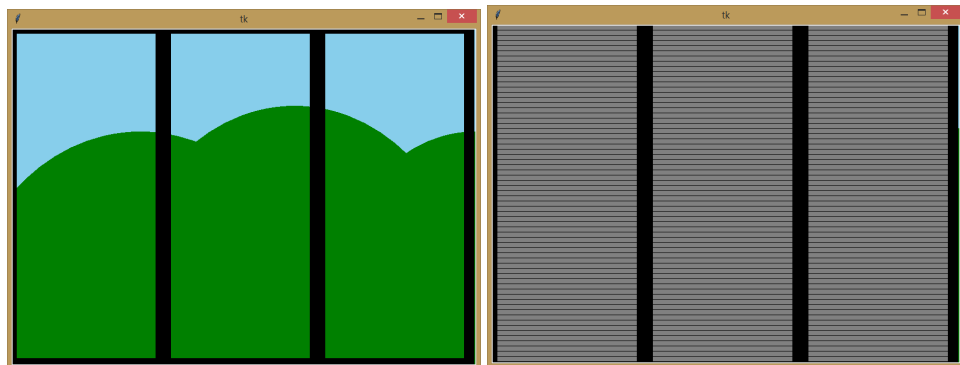
- Tlačidlom zmaž zmažeme všetky napísané titulky.



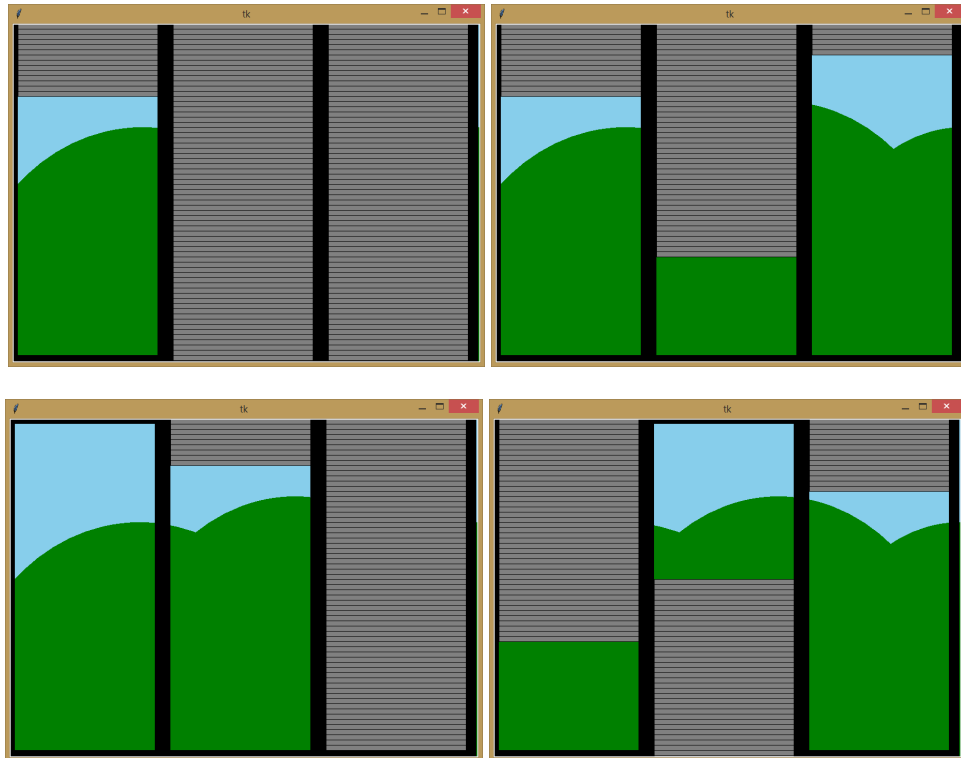
02

Vytvorte program, ktorý nám zobrazuje pohľad z okna a umožňuje nám posúvať rolety.

- Rozmer grafického plátna je 900 x 650 bodov.
- V programe sa najprv nakreslí bledomodré nebo a pred ním tri zelené kopce.
- Následne sa nakreslí rám okna rozdelený na tri rovnaké časti.



- Ihneď po nakreslení pozadia a rámu okna sa nakreslia v každom okne rolety. Každé okno je zatiahnuté roletami. Jedna lišta rolety má výšku 10 bodov.
- Stláčaním čísiel 1, 2 alebo 3 na klávesnici určujeme, či chceme ovládať ľavú, strednú alebo pravú roletu. Po spustení programu je štandardne na ovládanie pripravená prvá roleta.
- Stlačením šípky hore alebo šípky dole posúvame príslušnú roletu (podľa toho, ktoré číslo sme naposledy stlačili) smerom hore alebo dole. Ovládaním môžeme dostať napríklad takúto situáciu:



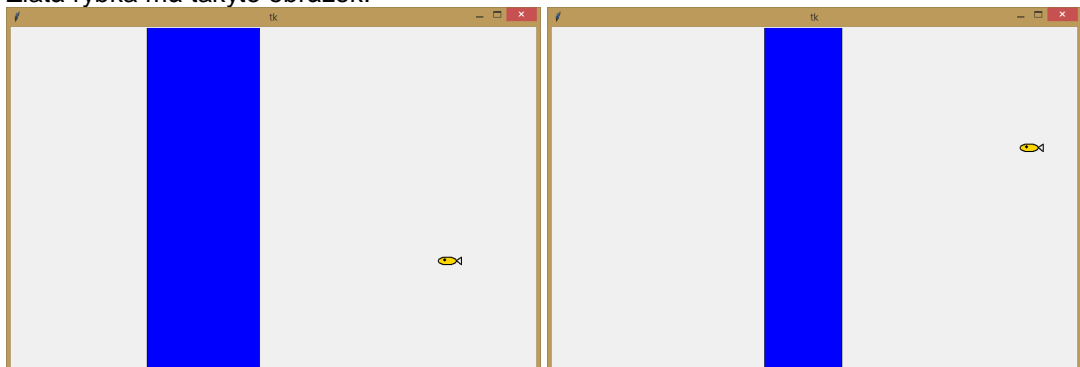
03

Vytvorte program Zachráň zlatú rybku.

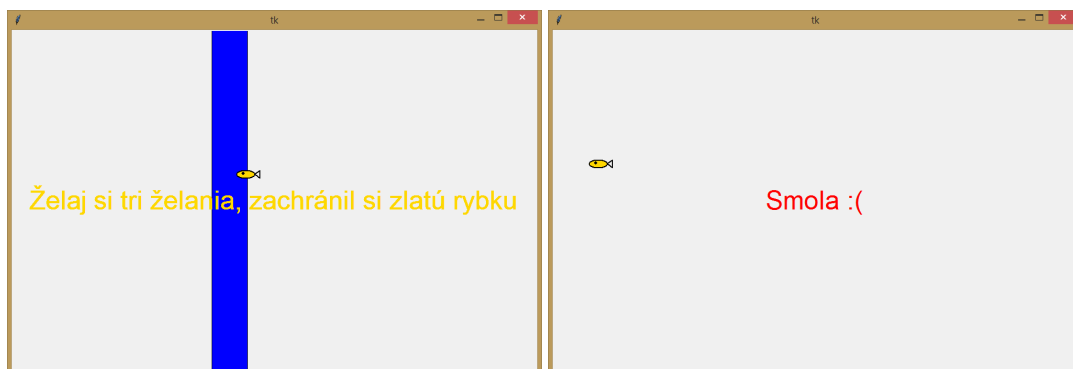
- Rozmer grafického plátna je 1000 x 650 bodov.
- Po spustení programu sa nakreslí na náhodnom mieste obrazovky zvislý rovný vodný kanál. Vodný kanál má náhodnú šírku od 100 do 300 px. Kreslí sa síce na náhodnom mieste, ale tak, aby mal z ľavého alebo pravého okraja odstup minimálne 100 px.
- Po nakreslení kanála sa na náhodnom mieste nakreslí zlatá rybka. Zlatá rybka sa však kreslí na náhodnom mieste buď vľavo od kanála alebo vpravo od kanála. Nikdy sa nenakreslí priamo vo vodnom kanáli.



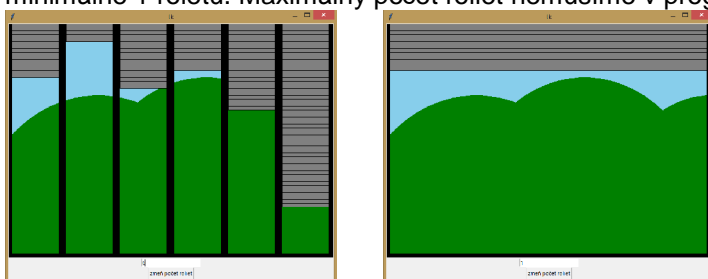
- Zlatá rybka má takýto obrázok:



- Úlohou hráča je zachrániť zlatú rybku a pomocou klávesnice ju premiestniť do vodného kanála. Rybku posúvame šípkou vľavo a vpravo. V rámci zvislej osi rybku neposúvame.
- Vodný kanál postupne vysychá, čiže prúd vody sa postupne zužuje. Hráč musí stihnúť zachrániť rybku skôr, ako vyschne kanál.
- Keď sa rybka dostane do vody, vypíše sa nám oznam: „Želaj si tri želania, zachránil si zlatú rybku.“ Oznam je napísaný v strede obrazovky väčším písmom zlatej farby.
- Ihneď po záchrane rybky je naším prvým želaním, aby prestal vysychať kanál, lebo rybka by nám nemusela stihnúť splniť ďalšie dve želania. Minimálne prvé želanie sa nám určite splní a kanál prestane vysychať.
- Ak sa nám nepodarí rybku zachrániť do vyschnutia kanála, vypíše sa nám v strede obrazovky oznam „Smola :(“. Oznam je napísaný väčším písmom a červenou farbou.



Prémia 1 Upravte program **Pohľad z okna** tak, aby v programe boli štandardne 3 rolety, ale napísaním počtu roliet do entry a stlačením tlačidla **Zmeň počet roliet** program zmení počet roliet aj s orámovaním okna. Šípkou vpravo postupne meníme, ktoré rolety sú aktívne na ovládanie. Ak sú aktívne posledné rolety, stlačením šípky vpravo sa stanú aktívne prvé rolety. Aktívne rolety posúvame hore alebo dole príslušnými šípkami. V programe môžeme mať minimálne 1 roletu. Maximálny počet roliet nemusíme v programe riešiť.



KONIEC TESTU

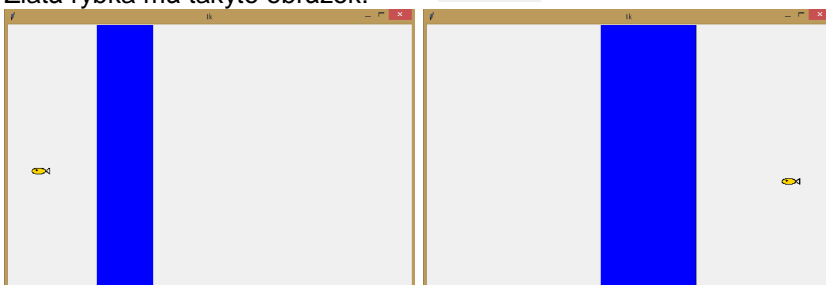
Polročný test č. 2 po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	-
Kód testu:	6302

Všeobecné pokyny: Test riešite na počítači v jazyku Python. Žiadne pomôcky nie sú povolené. Pracujte pokojne a s rozvahou. Svoju prácu si **priebežne ukladajte** na svoj disk _____ do priečinku _____. Súbory v tomto priečinku tiež pomenujte vašim priezviskom a číslom úlohy. V závere všetky súbory z priečinka pošlite v prílohe e-mailu na adresu _____.

01

Vytvorte program Zachráň zlatú ryбку.

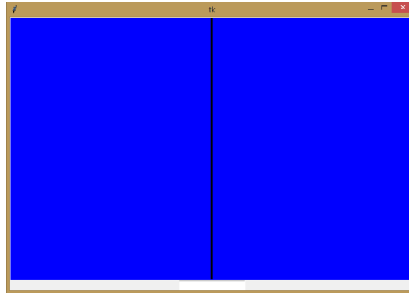
- Rozmer grafického plátna je 1000 x 650 bodov.
- Po spustení programu sa nakreslí na náhodnom mieste obrazovky zvislý rovný vodný kanál. Vodný kanál má náhodnú šírku od 100 do 300 px. Kreslí sa síce na náhodnom mieste, ale tak, aby mal z ľavého alebo pravého okraja odstup minimálne 100 px.
- Po nakreslení kanála sa na náhodnom mieste nakreslí zlatá ryбка. Zlatá ryбка sa však kreslí na náhodnom mieste buď vľavo od kanála alebo vpravo od kanála. Nikdy sa nenakreslí priamo vo vodnom kanáli.
- Zlatá ryбка má takýto obrázok:



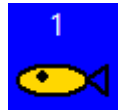
- Úlohou hráča je zachrániť zlatú ryбку a pomocou klávesnice ju premiestniť do vodného kanála. Ryбку posúvame šípkou vľavo a vpravo. V rámci zvislej osi ryбку neposúvame.
- Keď sa ryбка dostane do vody, vypíše sa nám oznam: „Želaj si tri želania, zachránil si zlatú ryбку.“ Oznam je napísaný v strede obrazovky väčším písmom zlatej farby.



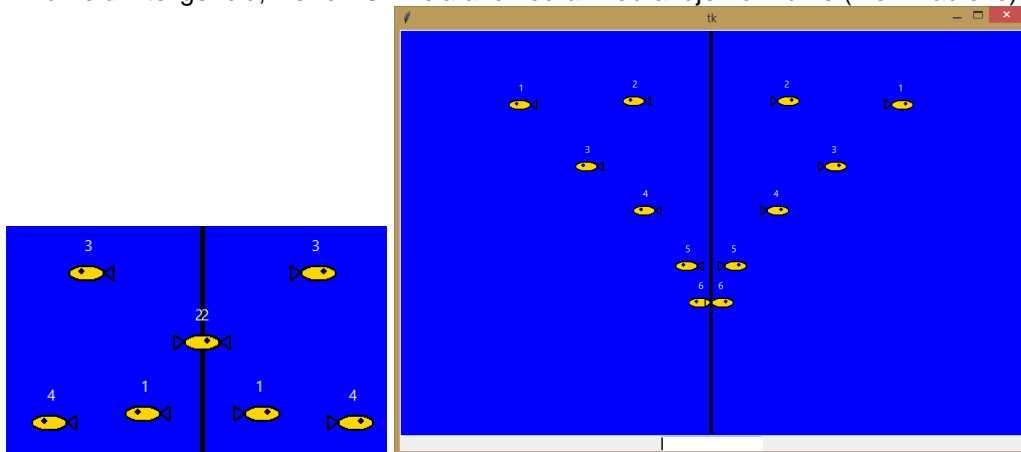
02 Vytvorte program **Akvárium**.



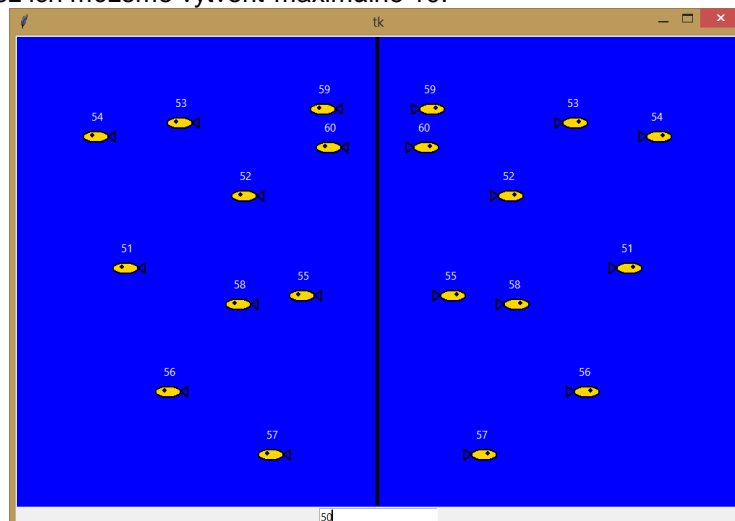
- Rozmer grafického plátna je 1000 x 650 bodov.
- Pozadie je modré a v strede je hrubá čierna čiara. Ľavá časť je akvárium s vodou a pravá časť je zrkadlo.
- Pod akváriom je súčiastka entry.
- Kliknutím myši do ľavej časti akvária na mieste kliknutia vytvoríme rybu:



- Ryby sú číslované číslami. Ich čísla sú napísané bielou farbou presne nad rybou.
- Keďže v pravej časti je zrkadlo, po vytvorení ryby v ľavej časti sa v pravej časti nakreslí jej odraz v zrkadle (na príslušnej zrkadlovej pozícii). Keďže zrkadlo má v sebe implementovanú umelú inteligenciu, vie rozlíšiť čísla a ich odraz zobrazuje normálne (nie zrkadlovo).



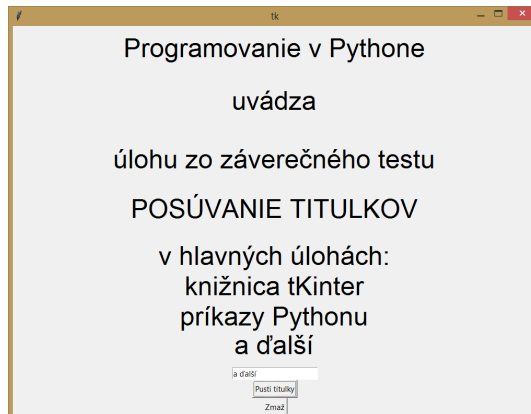
- V programe môžeme vytvoriť maximálne 10 rýb, viac rýb nám program nedovolí vytvoriť.
- Do súčiastky entry píšeme číslo a jednotlivé ryby sú číslované nie od čísla 1, ale od nasledujúceho čísla, ktoré je uvedené v entry. Ak v entry je číslo 50, ryby sú číslované od čísla 51 a tiež ich môžeme vytvoriť maximálne 10.



03

Vytvorte program na písanie titulkov do filmu.

- Rozmer grafického plátna je 1000 x 650 bodov.
- V programe je súčiastka na zadanie vstupu a dve tlačidlá (Pusti titulky a Zmaž).
- Do vstupného okienka môžeme napísať text titulu.
- Kliknutím ľavého tlačidla myši umiestnime do grafickej plochy text, ktorý je práve napísaný v súčiastke na zadávanie vstupu. Text sa napíše vo výške kliknutia myši, ale horizontálne je vždy vycentrovaný vzhľadom na veľkosť grafického plátna.
- Text je napísaný čiernou farbou, písmom Arial s väčšou veľkosťou.
- Po naklikaní titulkov môže vyzeráť obrazovka napríklad takto (podľa toho, aké texty sme napísali):



- Po stlačení tlačidla Pusti titulky sa titulky strihom (ihneď) presunú pod spodný okraj grafickej plochy, aby boli pripravené na animáciu a následne sa pomaly presúvajú smerom hore. Keď z obrazovky odíde celá pôvodná obrazovka (aj keď nie je celá zapísaná), znovu sa tie isté titulky zobrazujú od spodného okraja.
- Pribežne môžeme myšou doklikávať ďalšie titulky, no pri doklikávaní nám to môže pokaziť ich odchod a príchod celej jednej obrazovky.
- Tlačidlom zmaž zmažeme všetky napísané titulky.

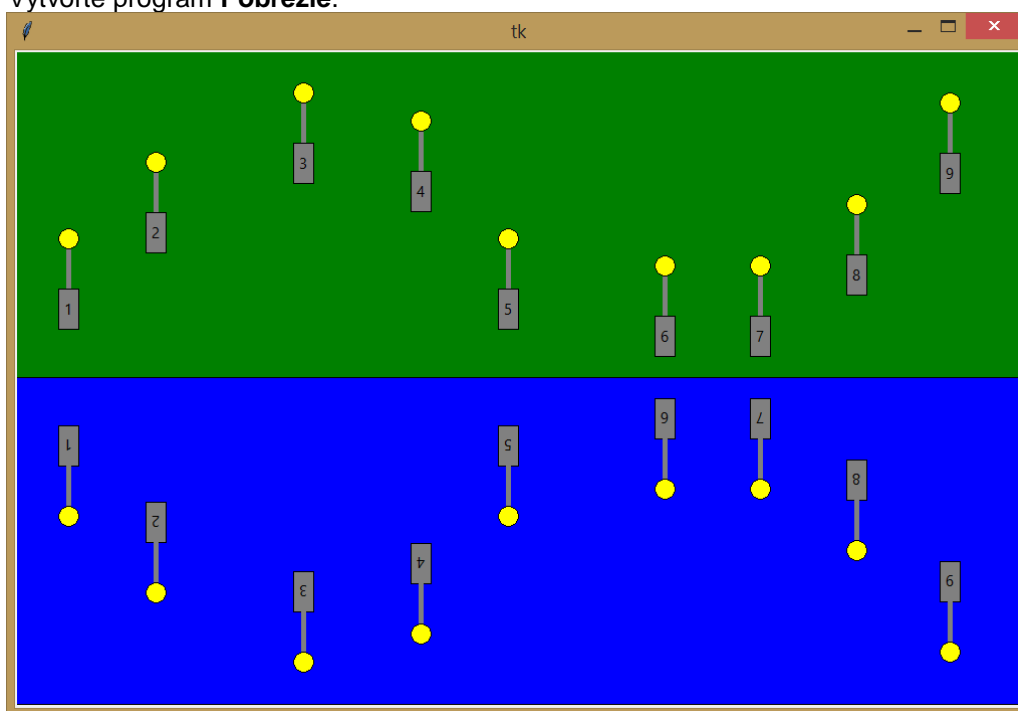


KONIEC TESTU

Polročný test č. 3 po kapitole 14	
Typ testu:	Polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	-
Kód testu:	6303

Všeobecné pokyny: Test riešite na počítači v jazyku Python. Žiadne pomôcky nie sú povolené. Pracujte pokojne a s rozvahou. Svoju prácu si **priebežne ukladajte** na svoj disk _____ do priečinku _____. Súbory v tomto priečinku tiež pomenujte vašim priezviskom a číslom úlohy. V závere všetky súbory z priečinku pošlite v prílohe e-mailu na adresu _____.

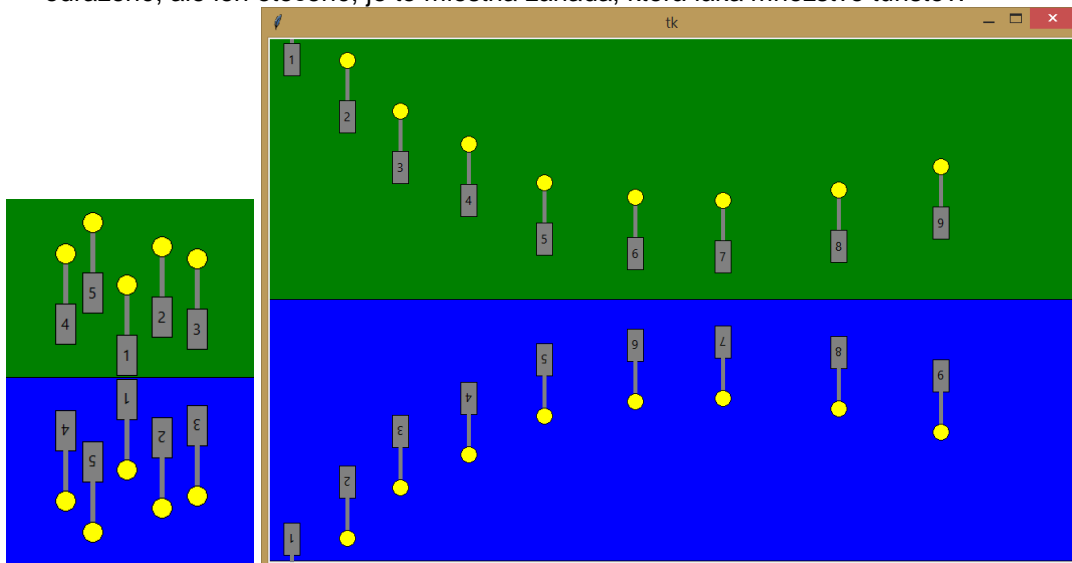
01 Vytvorte program **Pobrežie**.



- Rozmer grafického plátna je 1000 x 650 bodov.
- V hornej časti je zelená plocha pobrežia a v dolnej časti je modré more (presne v polovici). V mori sa zrkadlovo odrážajú lampy z pobrežia.
- Kliknutím myši na pobrežie vytvoríme na mieste kliknutia lampu:



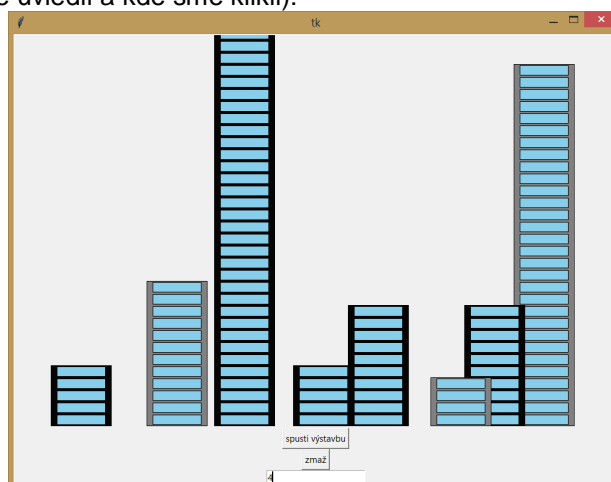
- V programe môžeme vytvoriť maximálne 9 lúč, viac lúč nám program nedovolí vytvoriť.
- Lampy sú číslované číslami postupne od 1 do 9. Ich čísla sú napísané čiernou farbou presne v strede spodnej časti.
- Keďže vodná hladina zrkadlovo odráža lampy, po vytvorení lampy na pobreží sa na mori nakreslí jej zrkadlový odraz (na príslušnej zrkadlovej pozícii). Číslo lúč nie sú zrkadlovo odrazené, ale len otočené, je to miestna záhada, ktorá láka množstvo turistov.



02

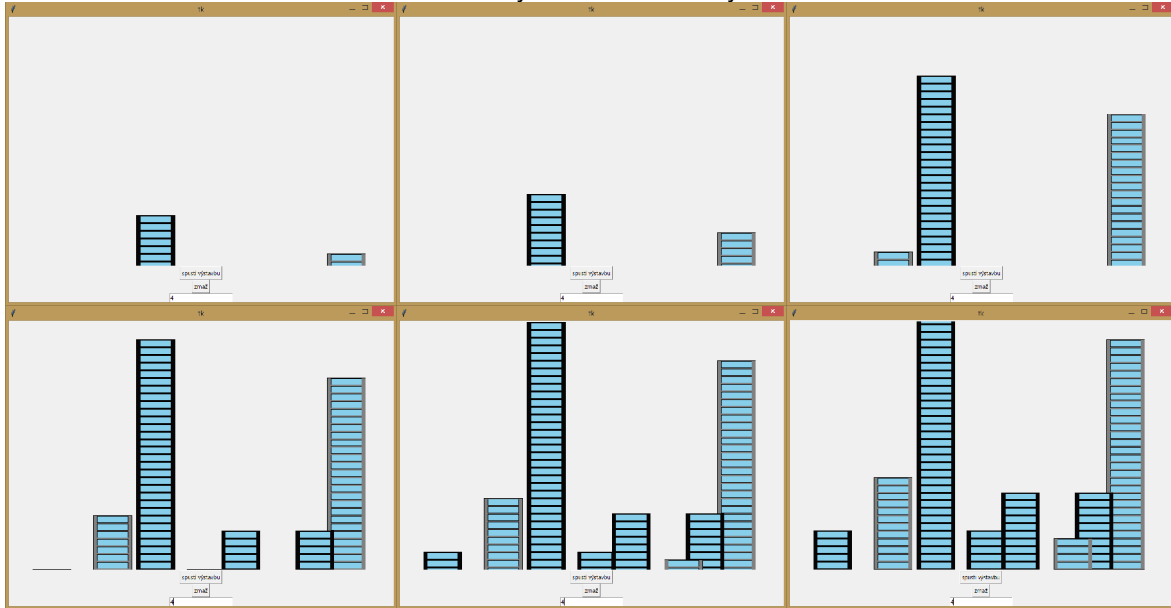
Vytvorte program na návrh mestskej výstavby a simuláciu výstavby.

- Rozmer grafického plátna je 1000 x 650 bodov.
- V programe je súčiastka na zadanie vstupu a tlačidlo (spusti výstavbu).
- Do vstupného okienka môžeme napísať výšku obytného domu (počet poschodí).
- Kliknutím ľavého tlačidla myši umiestnime do grafickej plochy obytný dom, ktorého počet poschodí je práve napísaný v súčiastke na zadávanie vstupu.
- Ľavá strana obytného domu sa nakreslí v mieste kliknutia myši, ale vertikálne je vždy zarovnaný na spodnom okraji grafického plátna.
- Obytný dom je nakreslený nejakou náhodnou farbou zo zoznamu vymenovaných (vyberte si aspoň 3 farby). Na každom poschodí je presklená stena svetlomodrej farby.
- Po naklikaní obytných domov môže vyzerat' obrazovka napríklad takto (podľa toho, aké počty poschodí sme uviedli a kde sme klikli):



- Po stlačení tlačidla spusti výstavbu sa domy strihom (ihneď) presunú pod spodný okraj grafickej plochy, aby boli pripravené na animáciu a následne sa pomaly presúvajú smerom hore – čím simulujeme postupnú výstavbu domov (postupne sa ukazujú poschodia). Keď sa na obrazovke ukážu všetky domy (aj najvyšší možný, ktorý tam práve nemusí byť), začne sa animácia výstavby opakovať odznova od spodného okraja.
- Priebežne môžeme myšou doklikávať ďalšie domy, no pri doklikávaní nám to môže pokaziť ich simuláciu a zarovnanie spodných častí (celej jednej obrazovky).

- Tlačidlom zmaž zmažeme všetky nakreslené domy.



03

Vytvorte program **Záchranár**.

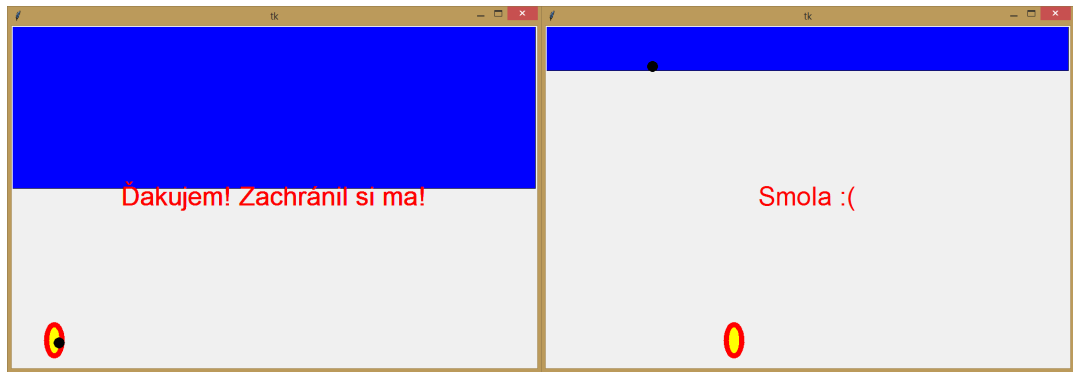
- Rozmer grafického plátna je 1000 x 650 bodov.
- Po spustení programu sa nakreslí na náhodnom mieste obrazovky osoba, ktorú treba zachrániť (čierny krúžok). V dolnej časti (vždy rovnaká vzdialenosť od spodného okraja) a náhodnom mieste zľava sa nakreslí nafukovací čln:



- Z hornej časti obrazovky sa postupne zaplavuje grafická plocha. Záplavová vlna má vodorovný okraj a je modrej farby.



- Úlohou hráča je zachrániť osobu a pomocou klávesnice ju premiestniť do záchranného člna. Osobu posúvame šípkou vľavo, vpravo, hore a dole.
- Keď sa osoba dostane do člna, vypíše sa nám oznam: „Ďakujem! Zachránil si ma!“. Oznam je napísaný v strede obrazovky väčším písmom červenej farby.
- Ihneď po záchrane šípkami presúvame osobu súčasne s člnom.
- Ak osobu zasiahne záplavová vlna, program zastaví záplavovú vlnu a napíše nám v strede obrazovky oznam „Smola :(“. Oznam je napísaný väčším písmom a červenou farbou.
- Zabezpečte, aby záplavová vlna nezakryla ani čln, ani osobu, ani oznam. Pomôcka: obrázky nakreslené neskôr zakryjú už nakreslené veci (ak ich prekryvajú).



Prémia 1 Upravte program **Pobrežie** tak, aby sme pomocou klávesnice mohli posúvať lampy (dole a hore) a podľa toho sa budú posúvať aj odrazy lúčov. Program nám ale nedovolí také posunutie smerom dole, pri ktorom by lampa opustila pobrežie a tiež nám nedovolí také posunutie smerom hore, pri ktorom by odraz na hladine opustil more (smerom na pobrežie).

KONIEC TESTU

1.2 Riešenie testov

Niektoré z testov obsahujú aj riešenia úloh, ktoré, samozrejme, nemusia byť vždy optimálnym a už vôbec nie jediným riešením. Niektoré riešené programy obsahujú aj časť kódu, ktorý nebol uvedený priamo v učebnici. Vychádzame z toho, že ukážka riešenia testu môže byť nástrojom, pomocou ktorého učíme študentov nové poznatky alebo ich motivujeme na diskusiu o spôsoboch a kvalite riešenia. Napríklad v teste 6201B nájdeme zápis `maxx=int(canvas["width"])`. Neznamená to, že bez tohto zápisu by sme úlohu nevedeli vyriešiť alebo že toto bolo potrebné v úlohe zapísať. Treba to brať ako ďalšie inšpirácie, ktoré pomáhajú aj šikovným študentom posúvať sa ďalej. Šikovní študent si totiž často môže myslieť, že z riešenia úlohy, ktorú vedel vyriešiť, sa nemá čo naučiť, čo nemusí byť pravda.

Test 5101A

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='white',width=1000,height=800)
canvas.pack()

#hodiny
canvas.create_rectangle(50,50,320,150, fill='gray')
canvas.create_rectangle(60,60,310,140, fill='', outline='red', width=5)
cas=randint(10,23)
canvas.create_text(150,100,text=':', fill='white', font='Arial 40')
canvas.create_text(105,100,text=cas, fill='white', font='Arial 40')
canvas.create_text(195,100,text=cas, fill='white', font='Arial 40')
canvas.create_oval(250,75,300,125, outline='',fill='white')
canvas.create_oval(235,75,285,125, outline='',fill='gray')

#lampa
canvas.create_oval(400,100,600,300,fill='red',outline='')
canvas.create_line(370,150,600,330,fill='white',width=120)
canvas.create_line(550,130,650,130,650,430,fill='blue',width=30)
canvas.create_line(550,430,750,430,fill='blue',width=30)
canvas.create_oval(460,130,560,230,fill='yellow',outline='')

vyska=randint(10,40)
canvas.create_rectangle(570,415-vyska,590,415,fill='green')
```

Test 5101B

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='white',width=1000,height=800)
canvas.pack()

#hodiny
canvas.create_rectangle(50,50,320,150, fill='gray')
canvas.create_rectangle(60,60,310,140, fill='', outline='green', width=5)
cas=randint(10,23)
canvas.create_text(220,100,text=':', fill='white', font='Arial 40')
canvas.create_text(175,100,text=cas, fill='white', font='Arial 40')
canvas.create_text(265,100,text=cas, fill='white', font='Arial 40')
canvas.create_oval(70,75,120,125, outline='',fill='white')
canvas.create_oval(85,75,135,125, outline='',fill='gray')

#lampa
canvas.create_oval(700,100,900,300,fill='blue',outline='')
canvas.create_line(715,320,945,140,fill='white',width=120)
canvas.create_line(750,130,650,130,650,430,fill='green',width=30)
canvas.create_line(550,430,750,430,fill='green',width=30)
canvas.create_oval(740,130,840,230,fill='yellow',outline='')

vyska=randint(10,40)
canvas.create_rectangle(570,415-vyska,590,415,fill='red')
```


Test 5201A

```
import tkinter
from random import*
canvas=tkinter.Canvas(height=1000, width=500)
canvas.pack()

canvas.create_text(250,30,text='Ako sa stretla nula s jedničkou',font='Arial
20',fill='green')

canvas.create_rectangle(50,60,240,890,fill='black')

y=65
for i in range(0,55):
    canvas.create_rectangle(55,y,65,y+10,fill='white')
    canvas.create_rectangle(225,y,235,y+10,fill='white')
    y=y+15

y=65
for i in range(0,4):
    canvas.create_rectangle(70,y,70+150,y+200,fill='white')
    y=y+205

def nulajedna():
    x0_posun=randint(15,130)
    x1_posun=randint(15,130)
    y0=randint(40,170)
    y1=randint(40,170)
    x=70
    y=65
    for i in range(4):
        canvas.create_text(x+x0_posun,y+y0,text='0',font='Arial 50',fill='green')
        canvas.create_text(x+x1_posun,y+y1,text='1',font='Arial 50',fill='orange')
        y=y+205

nulajedna()
```

Test 5201B

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1200,height=300)
canvas.pack()

maxx=int(canvas["width"])
maxy=int(canvas["height"])
vrch=50
canvas.create_rectangle(0,vrch,maxx,maxy,fill='black')

x=10
for i in range(50):
    canvas.create_rectangle(x,vrch+10,x+20,vrch+30,fill='white')
    canvas.create_rectangle(x,maxy-30,x+20,maxy-10,fill='white')
    x=x+25

x=10
for i in range(4):
    canvas.create_rectangle(x,vrch+40,x+280,maxy-40,fill='white')
    x=x+300

def policko():
    x=40
    y0=randint(80,170)
    x0_posun=randint(10,180)
    x1_posun=randint(150,240)
    y1=randint(80,170)
    for i in range(4):
        canvas.create_text(x+x0_posun,vrch+y0,text='0',font='Arial 50',fill='blue')
```

```

        canvas.create_text(x+x1_posun,vrch+y1,text='1',font='Arial 50',fill='red')
        x=x+300

canvas.create_text(maxx/2,25,text="Informatický film: Ako sa stretla nula s
jedničkou",fill='blue',font='Calibri 20')
policko()

```

Test 5301

```

import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def infotabula(x,y):
    cas = randint(10,15)
    canvas.create_rectangle(x,y,x+200,y+420,width=10,outline='red',fill='green')
    canvas.create_text(x+100,y+25,text='Next boarding',fill='yellow',font='Arial 18')
    y1= y+70
    skupina = randint(100,500)
    for i in range(0,4):
        canvas.create_text(x+40,y1,text=cas,fill='white',font='Arial 18')
        canvas.create_text(x+80,y1,text=':00',fill='white',font='Arial 18')
        canvas.create_text(x+150,y1,text=skupina,fill='white',font='Arial 18')
        skupina=skupina+1
        y1 = y1+30
        canvas.create_text(x+40,y1,text=cas,fill='white',font='Arial 20')
        canvas.create_text(x+80,y1,text=':30',fill='white',font='Arial 20')
        canvas.create_text(x+150,y1,text=skupina,fill='white',font='Arial 20')
        skupina=skupina+1
        y1 = y1+30
        cas=cas+1
    nosmoking(x+100,y+350)

def nosmoking(x,y):
    canvas.create_oval(x-50,y-50,x+50,y+50,width=10,outline='red',fill='white')
    canvas.create_rectangle(x-40,y-8,x-15,y+8,fill='orange',outline='orange')
    canvas.create_rectangle(x-13,y-8,x+30,y+8,fill='white')
    canvas.create_rectangle(x+32,y-8,x+38,y+8,fill='red',outline='red')
    canvas.create_line(x-38,y-38,x+38,y+38,width=10,fill='red')

def hlava(x,y):
    f=choice(('khaki','khaki1','khaki2','brown'))
    canvas.create_oval(x-30,y-30,x+30,y+30,fill=f)
    f=choice(('brown','black','yellow','gray'))
    for i in range(30):
        zacx = x+randint(-20,20)
        zacy = y-30+randint(0,15)
        canvas.create_line(zacx,zacy,zacx+randint(-25,25),zacy+40+randint(-
15,15),width=3,fill=f)

def davludi():
    for i in range(300):
        hlava(randint(50,700),randint(450,600))

def most(x,y,pocet):
    canvas.create_line(x,y+200,x+pocet*100+300,y-200,fill='blue',width=310)
    canvas.create_rectangle(x,y,x+pocet*100,y+80,fill='brown',outline='')
    x1=x+20
    for i in range(pocet):
        canvas.create_oval(x1,y+30,x1+60,y+80,fill='blue',outline='')
        canvas.create_rectangle(x1,y+55,x1+60,y+80,fill='blue',outline='')
        x1 = x1+100

def rieka(x,y,sirka,vyska):
    canvas.create_rectangle(x,y,x+sirka,y+vyska,fill='blue')

def zabradlie(x,y,pocet):

```

```

x1 = x
y1 = y
for i in range(pocet):
    canvas.create_line(x1,y1,x1,y1-30,width=2)
    x1=x1+8
    y1=y1-3

def kresli():
    most(10,150,8)
    zabradlie(100,150,50)
    infotabula(800,230)
    davludi()

def premia2():
    most(10,150,randint(3,10))

def premial():
    most(10,150,8)
    zabradlie(100,150,50)
    infotabula(800,230)
    pocet_riadok=randint(2,4)
    pocet_stlpec=randint(3,6)
    x=250
    y=400
    for j in range(pocet_riadok):
        for i in range(pocet_stlpec):
            hlava(x+i*60,y+j*60)
            x=x-60
            pocet_stlpec=pocet_stlpec+2

kresli()
#premiat()
#premia2()

```

Test 5302

```

import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def chram(x,y,pocet):
    posunx = randint(50,100)
    posuny = randint(20,100)
    x1 = x
    for i in range(pocet):
        canvas.create_line(x1+posunx,y+posuny,x1+20,y+200,fill='black',width=20)
        canvas.create_rectangle(x1,y,x1+20,y+200,fill='gray')
        x1=x1+60
        canvas.create_line(x-30,y,x+pocet*60,y,x+pocet*60/2-30,y-70,x-
30,y,width=20,fill='gray')
        x2 = x
        y2 = y + 200
        posunx = 0
        for i in range(10):
            canvas.create_line(x2-posunx,y2,x+pocet*60-30+posunx,y2)
            y2 = y2 + 10
            posunx = posunx+20

def infotabula(x,y):
    cas = randint(10,15)
    canvas.create_rectangle(x,y,x+200,y+500,width=10,outline='red',fill='blue')
    canvas.create_text(x+100,y+25,text='Next entry',fill='yellow',font='Arial 20')
    y1= y+70
    skupina = randint(100,500)
    for i in range(0,5):
        canvas.create_text(x+40,y1,text=cas,fill='white',font='Arial 20')
        canvas.create_text(x+80,y1,text=':00',fill='white',font='Arial 20')

```

```

        canvas.create_text(x+150,y1,text=skupina,fill='white',font='Arial 20')
        skupina=skupina+1
        y1 = y1+30
        canvas.create_text(x+40,y1,text=cas,fill='white',font='Arial 20')
        canvas.create_text(x+80,y1,text=':30',fill='white',font='Arial 20')
        canvas.create_text(x+150,y1,text=skupina,fill='white',font='Arial 20')
        skupina=skupina+1
        y1 = y1+30
        cas=cas+1
    nodrink(x+100,y+420)

def nodrink(x,y):
    canvas.create_oval(x-50,y-50,x+50,y+50,width=10,outline='red',fill='white')
    canvas.create_line(x,y+10,x,y+30,width=5,fill='black')
    canvas.create_line(x-10,y+30,x+10,y+30,width=5,fill='black')
    canvas.create_oval(x-20,y-30,x+20,y+10,width=2,fill='white')
    canvas.create_rectangle(x-20,y-30,x+20,y-10,fill='white',outline='white')
    canvas.create_line(x-20,y-10,x+20,y-10,fill='black',width=2)
    canvas.create_line(x-38,y-38,x+38,y+38,width=10,fill='red')

def hlava(x,y):
    f=choice(('khaki','khaki1','khaki2','brown'))
    canvas.create_oval(x-30,y-30,x+30,y+30,fill=f)
    f=choice(('brown','black','yellow','gray'))
    for i in range(30):
        zacx = x+randint(-20,20)
        zacy = y-30+randint(0,15)
        canvas.create_line(zacx,zacy,zacx+randint(-25,25),zacy+40+randint(-
15,15),width=3,fill=f)

def davludi():
    for i in range(300):
        hlava(randint(50,700),randint(450,600))

def kresli():
    chram(180,100,6)
    infotabula(780,10)
    davludi()

def premia2():
    chram(180,100,randint(3,10))
    infotabula(780,10)

def premial():
    chram(180,100,6)
    infotabula(780,10)
    pocet_riadok=randint(2,4)
    pocet_stlpec=randint(3,6)
    x=250
    y=300
    for j in range(pocet_riadok):
        for i in range(pocet_stlpec):
            hlava(x+i*60,y+j*60)
            x=x-60
            pocet_stlpec=pocet_stlpec+2

kresli()
#premi1()
#premia2()

```

Test 6101A

1. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='white', width=1000, height=500)
canvas.pack()

def pis_text(suradnice):
    x = suradnice.x
    y = suradnice.y
    uhol = x
    if y<250:
        canvas.create_text(x,y, text='Python',fill='red', angle=uhol)
    else:
        canvas.create_text(x,y, text='Python',fill='blue', angle=-uhol)

def stvorce(suradnice):
    x = suradnice.x
    y = suradnice.y
    for i in range(0,10):
        canvas.create_rectangle(x+i*20,y,x+i*20+15,y+15)

canvas.create_line(0,250,1000,250,width=5)
canvas.bind('<Button-1>', pis_text)
canvas.bind('<Button-3>', stvorce)
```

2. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='white', width=1000, height=500)
canvas.pack()

def vlajka_dansko():
    x = randrange(1000)
    y = randrange(500)
    if x<300 and y<200:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'red')
    if x>400 and y<200:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'red')
    if x>400 and y>300:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'red')
    if x<300 and y>300:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'red')
    canvas.after(2, vlajka_dansko)

vlajka_dansko()
```

Test 6101B

1. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='white', width=400, height=500)
canvas.pack()

def pis_text(suradnice):
    x = suradnice.x
    y = suradnice.y
    uhol = y
    if x<200:
        canvas.create_text(x,y, text='programovanie',fill='green', angle=uhol)
    else:
```

```
        canvas.create_text(x,y, text='je zábava',fill='orange', angle=-uhol)

def stvorce(suradnice):
    x = suradnice.x
    y = suradnice.y
    for i in range(0,10):
        canvas.create_rectangle(x,y+i*15,x+10,y+i*15+10)

canvas.create_line(200,0,200,600,width=5)
canvas.bind('<Button-1>', pis_text)
canvas.bind('<Button-3>', stvorce)
```

2. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(bg='yellow', width=400, height=500)
canvas.pack()

def vlajka_svedsko():
    x = randrange(400)
    y = randrange(500)
    if x<160 and y<110:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'royalblue')
    if x>240 and y<110:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'royalblue')
    if x>240 and y>190:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'royalblue')
    if x<160 and y>190:
        canvas.create_oval(x-10, y-10, x+10, y+10, fill = 'royalblue')
    canvas.update()
    canvas.after(2, vlajka_svedsko)

vlajka_svedsko()
```

Test 6201A

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=500)
canvas.pack()

maxx=int(canvas["width"])
maxy=int(canvas["height"])

def cesty():
    canvas.create_rectangle(0,maxy/2-40,maxx,maxy/2+40, fill='gray',outline='')
    canvas.create_rectangle(maxx/2-40,0,maxx/2+40,maxy, fill='gray',outline='')
    pocet = int(maxx/30)
    for i in range(pocet):
        canvas.create_rectangle(i*30,maxy/2-5,i*30+15,maxy/2+5, fill='white')
    pocet = int(maxy/30)
    for i in range(pocet):
        canvas.create_rectangle(maxx/2-5,i*30,maxx/2+5,i*30+15, fill='white')

def vlak(x,y,pocet):
    for i in range(pocet):
        canvas.create_rectangle(x,y,x+35,y+20, fill='yellow',tags='vlak')
        canvas.create_oval(x+5,y+15,x+15,y+25, fill='black',tags='vlak')
        canvas.create_oval(x+20,y+15,x+30,y+25, fill='black',tags='vlak')
        x=x+37

def start():
    global pocet, vlak_x
    pocet = int(entry1.get())
    vlak_x=maxx-37*pocet
```

```

    vlak(vlak_x,maxy/2-30,pocet)
    animuj()

def bus(x,y):
    canvas.create_rectangle(x-10,y-40,x+10,y, fill='red',tags='bus')

def animuj():
    global vlak_x,bus_y
    canvas.move('vlak',-5,0)
    vlak_x = vlak_x-5
    if not (maxy/2-60<bus_y<maxy/2 and vlak_x<maxx/2+30 and vlak_x+37*pocet>maxx/2-30):
        canvas.move('bus',0,5)
        bus_y = bus_y+5
    if vlak_x+pocet*37<=0:
        vlak_x=maxx
        canvas.move('vlak',maxx+pocet*37,0)
    if bus_y>=maxy:
        bus_y=0
        canvas.move('bus',0,-maxy)
    if pokračovat==1:
        canvas.after(10,animuj)

def pokračuj():
    global pokračovat
    if pokračovat==0:
        pokračovat=1
        animuj()
    else:
        pokračovat=0

pokracovat=1
vlak_x=0
bus_x=maxx/2-20
bus_y=30

cesty()
bus(bus_x,bus_y)
canvas.bind_all('<space>',pokracuj)
button1=tkinter.Button(text='štart',command=start)
button1.pack()
button2=tkinter.Button(text='zastav/pokračuj',command=pokracuj)
button2.pack()
entry1=tkinter.Entry()
entry1.pack()

```

Test 6201B

```

import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=500)
canvas.pack()

maxx=int(canvas["width"])
maxy=int(canvas["height"])

def cesty():
    canvas.create_rectangle(0,maxy/2-40,maxx,maxy/2+40, fill='gray',outline='')
    canvas.create_rectangle(maxx/2-40,0,maxx/2+40,maxy, fill='gray',outline='')
    pocet = int(maxx/30)
    for i in range(pocet):
        canvas.create_rectangle(i*30,maxy/2-5,i*30+15,maxy/2+5, fill='white')
    pocet = int(maxy/30)
    for i in range(pocet):
        canvas.create_rectangle(maxx/2-5,i*30,maxx/2+5,i*30+15, fill='white')

def vlak(x,y,pocet):
    for i in range(pocet):
        canvas.create_rectangle(x,y,x+35,y+20, fill='yellow',tags='vlak')

```

```

        canvas.create_oval(x+5,y+15,x+15,y+25, fill='black',tags='vlak')
        canvas.create_oval(x+20,y+15,x+30,y+25, fill='black',tags='vlak')
        x=x+37

def start():
    global pocet, vlak_x
    pocet = int(entry1.get())
    vlak_x=0 #maxx-37*pocet
    vlak(vlak_x,maxy/2+10,pocet)
    animuj()

def bus(x,y):
    canvas.create_rectangle(x-10,y,x+10,y+40, fill='red',tags='bus')

def animuj():
    global vlak_x,bus_y
    canvas.move('vlak',+5,0)
    vlak_x = vlak_x+5
    if not (maxy/2+30<bus_y<maxy/2+60 and vlak_x<maxx/2+30 and vlak_x+37*pocet>maxx/2-
30):
        canvas.move('bus',0,-5)
        bus_y = bus_y-5
    if vlak_x>=maxx:
        vlak_x=0-37*pocet
        canvas.move('vlak',-maxx-pocet*37,0)
    if bus_y<=0:
        bus_y=maxy
        canvas.move('bus',0,maxy)
    if pokracovat==1:
        canvas.after(10,animuj)

def pokracuj():
    global pokracovat
    if pokracovat==0:
        pokracovat=1
        animuj()
    else:
        pokracovat=0

pokracovat=1
vlak_x=0
bus_x=maxx/2+20
bus_y=maxy-30

cesty()
bus(bus_x,bus_y)
button1=tkinter.Button(text='Štart',command=start)
button1.pack()
button2=tkinter.Button(text='zastav/pokračuj',command=pokracuj)
button2.pack()
entry1=tkinter.Entry()
entry1.pack()

```

Test 6202

1. úloha

```

import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=500)
canvas.pack()

maxx=int(canvas["width"])
maxy=int(canvas["height"])

def postava(x,y,krajina,velkost,farba,cislo):
    canvas.create_oval(x-20,y-20,x+20,y+20)
    canvas.create_rectangle(x-30,y+20,x+30,y+20+velkost,fill=farba)

```



```

        canvas.create_text(x,y+40, text=krajina,font='Arial 20',fill='white')
        canvas.create_text(x,y+70, text=cislo,font='Arial 20',fill='white')

def klik1(suradnice):
    global pocet1
    x = suradnice.x
    y = suradnice.y
    pocet1 = pocet1+1
    krajina = entry1.get()
    velkost = int(entry2.get())
    if pocet1<7:
        postava(x,y,krajina,velkost,'red',pocet1)

def klik2(suradnice):
    global pocet2
    x = suradnice.x
    y = suradnice.y
    pocet2 = pocet2+1
    krajina = entry1.get()
    velkost = int(entry2.get())
    if pocet2<7:
        postava(x,y,krajina,velkost,'blue',pocet2)

pocet1 = 0
pocet2 = 0
entry1=tkinter.Entry()
entry1.pack()
entry2=tkinter.Entry()
entry2.pack()
canvas.bind('<Button-1>',klik1)
canvas.bind('<Button-3>',klik2)

```

2. úloha

```

import tkinter
canvas = tkinter.Canvas(width=1000,height=500)
canvas.pack()

cas=0
pokracovat=1

def odpocitavanie():
    global cas
    cas = cas - 1
    canvas.delete('all')
    canvas.create_text(500,250, text=cas, font='Arial 50', fill='green')
    if cas>0 and pokracovat==1:
        canvas.after(100,odpocitavanie)
    else:
        if cas==0:
            canvas.create_text(200,200,text='Čas vypršal')

def pokracuj():
    global pokracovat
    if pokracovat==0:
        if cas>0:
            pokracovat=1
            odpocitavanie()
    else:
        pokracovat=0

def start():
    global cas
    cas=int(entry1.get())
    odpocitavanie()

entry1 = tkinter.Entry()
entry1.pack()

```

```
button1 = tkinter.Button(text='Štart',command=start)
button1.pack()
button2 = tkinter.Button(text='zastav/pokracuj',command=pokracuj)
button2.pack()
```

Test 6301

1. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def klik(suradnice):
    x = 500
    y = suradnice.y
    canvas.create_text(x,y,text=entry1.get(), font='Arial 30',tags='titulky')

def pusti():
    global pocet_posunov
    canvas.move('titulky',0,-10)
    pocet_posunov=pocet_posunov+1
    if pocet_posunov>2*65:
        pocet_posunov=0
        canvas.move('titulky',0,2*650)
    canvas.after(100,pusti)

def start():
    canvas.move('titulky',0,650)
    pusti()

def zmaz():
    canvas.delete('all')

pocet_posunov=0
entry1 = tkinter.Entry()
entry1.pack()
button1 = tkinter.Button(text='Pusti titulky',command=start)
button1.pack()
button2 = tkinter.Button(text='Zmaž',command=zmaz)
button2.pack()

canvas.bind('<Button-1',klik)
```

2. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=900,height=650, bg='skyblue')
canvas.pack()

canvas.create_oval(-100,200,600,1000,fill='green',outline='')
canvas.create_oval(200,150,900,1000,fill='green',outline='')
canvas.create_oval(600,200,1200,1000,fill='green',outline='')

for i in range(3):
    canvas.create_rectangle(i*300,0,i*300+290,650,width=20)

for i in range(65):
    canvas.create_rectangle(10,i*10,280,i*10+10,fill='grey',tags='lava_roleta')
    canvas.create_rectangle(310,i*10,580,i*10+10,fill='grey',tags='stredna_roleta')
    canvas.create_rectangle(610,i*10,880,i*10+10,fill='grey',tags='prava_roleta')

def lava(event):
```

```
global posuvam
posuvam=1

def stredna(event):
    global posuvam
    posuvam=2

def prava(event):
    global posuvam
    posuvam=3

def hore(event):
    if posuvam==1:
        canvas.move('lava_roleta',0,-10)
    elif posuvam==2:
        canvas.move('stredna_roleta',0,-10)
    else:
        canvas.move('prava_roleta',0,-10)

def dole(event):
    if posuvam==1:
        canvas.move('lava_roleta',0,10)
    elif posuvam==2:
        canvas.move('stredna_roleta',0,10)
    else:
        canvas.move('prava_roleta',0,10)

posuvam=1
canvas.bind_all('1',lava)
canvas.bind_all('2',stredna)
canvas.bind_all('3',prava)
canvas.bind_all('<Up>',hore)
canvas.bind_all('<Down>',dole)
```

3. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def rieka(x,sirka):
    canvas.create_rectangle(x,0,x+sirka,650,fill='blue',tags='rieka')

def kresli_rybu(x,y):
    canvas.create_oval(x,y,x+35,y+15,fill='gold',width=2, tags='ryba')
    canvas.create_oval(x+10,y+3,x+15,y+8,fill='black', tags='ryba')
    canvas.create_line(x+35,y+7,x+45,y,x+45,y+15,x+35,y+7,width=2, tags='ryba')

def zisti_ci_zachranil():
    global zachranil
    if x<ryba_x<x+sirka:
        canvas.create_text(500,325,text='Želaj si tri želania, zachránil si zlatú rybku',
font='Arial 30',fill='gold')
        zachranil=1

def vlavo(event):
    global ryba_x
    ryba_x = ryba_x-5
    canvas.delete('ryba')
    kresli_rybu(ryba_x,ryba_y)
    zisti_ci_zachranil()

def vpravo(event):
    global ryba_x
    ryba_x = ryba_x+5
    canvas.delete('ryba')
    kresli_rybu(ryba_x,ryba_y)
```

```
zisti_ci_zachranil()

def vysychanie():
    global sirka
    canvas.delete('rieka')
    if sirka>0:
        sirka = sirka - 5
        rieka(x,sirka)
        if zachranil==0:
            canvas.after(200,vysychanie)
    else:
        canvas.create_text(500,325,text='Smola :(', font='Arial 30',fill='red')

zachranil=0
x = randint(100,500)
sirka = randint(100,300)
rieka(x,sirka)
ryba_y = randint(0,630)
ryba_x = randint(0,1)
if ryba_x==0:
    ryba_x=randint(0,x-50)
else:
    ryba_x=randint(x+sirka,1000-50)

kresli_rybu(ryba_x,ryba_y)
vysychanie()
canvas.bind_all('<Left>',vlavo)
canvas.bind_all('<Right>',vpravo)
```

Prémia 1

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=900,height=650, bg='skyblue')
canvas.pack()

def vykresli():
    canvas.create_oval(-100,200,600,1000,fill='green',outline='')
    canvas.create_oval(200,150,900,1000,fill='green',outline='')
    canvas.create_oval(600,200,1200,1000,fill='green',outline='')
    for i in range(pocet_roliet):
        canvas.create_rectangle(i*sirka,0,i*sirka+sirka,650,width=20)
    for j in range(pocet_roliet):
        for i in range(65):
            canvas.create_rectangle(j*sirka+10,i*10,j*sirka+sirka-
10,i*10+10,fill='grey',tags='roleta'+str(j))

def dalsia(event):
    global posuvam
    posuvam=posuvam+1
    if posuvam==pocet_roliet:
        posuvam=0

def hore(event):
    canvas.move('roleta'+str(posuvam),0,-10)

def dole(event):
    canvas.move('roleta'+str(posuvam),0,10)

def prekresli():
    global posuvam, pocet_roliet, sirka
    posuvam=0
    pocet_roliet=int(entry1.get())
    sirka = 900 / pocet_roliet
    canvas.delete('all')
    vykresli()

posuvam=0
```

```
pocet_roliet=3
sirka = 900 / pocet_roliet
vykresli()
canvas.bind_all('<Right>',dalsia)
canvas.bind_all('<Up>',hore)
canvas.bind_all('<Down>',dole)
entry1 = tkinter.Entry()
entry1.pack()
button1 = tkinter.Button(text='zmeň počet roliet',command=prekresli)
button1.pack()
```

Test 6302

1. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def rieka(x,sirka):
    canvas.create_rectangle(x,0,x+sirka,650,fill='blue')

def kresli_rybu(x,y):
    canvas.create_oval(x,y,x+35,y+15,fill='gold',width=2, tags='ryba')
    canvas.create_oval(x+10,y+3,x+15,y+8,fill='black', tags='ryba')
    canvas.create_line(x+35,y+7,x+45,y,x+45,y+15,x+35,y+7,width=2, tags='ryba')

def zisti_ci_zachranil():
    if x<ryba_x<x+sirka:
        canvas.create_text(500,325,text='Želaj si tri želania, zachránil si zlatú rybku',
font='Arial 30',fill='gold')

def vlavo(event):
    global ryba_x
    ryba_x = ryba_x-5
    canvas.delete('ryba')
    kresli_rybu(ryba_x,ryba_y)
    zisti_ci_zachranil()

def vpravo(event):
    global ryba_x
    ryba_x = ryba_x+5
    canvas.delete('ryba')
    kresli_rybu(ryba_x,ryba_y)
    zisti_ci_zachranil()

x = randint(100,500)
sirka = randint(100,300)
rieka(x,sirka)
ryba_y = randint(0,630)
ryba_x = randint(0,1)
if ryba_x==0:
    ryba_x=randint(0,x-50)
else:
    ryba_x=randint(x+sirka,1000-50)

kresli_rybu(ryba_x,ryba_y)
canvas.bind_all('<Left>',vlavo)
canvas.bind_all('<Right>',vpravo)
```

2. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650,bg="blue")
canvas.pack()

def kresli_rybuA(x,y,cislo):
    canvas.create_oval(x,y,x+35,y+15,fill='gold',width=2, tags='ryba')
    canvas.create_oval(x+10,y+3,x+15,y+8,fill='black', tags='ryba')
    canvas.create_line(x+35,y+7,x+45,y,x+45,y+15,x+35,y+7,width=2, tags='ryba')
    canvas.create_text(x+20,y-20,text=cislo,fill='white')

def kresli_rybuB(x,y,cislo):
    canvas.create_oval(x,y,x-35,y+15,fill='gold',width=2, tags='ryba')
    canvas.create_oval(x-10,y+3,x-15,y+8,fill='black', tags='ryba')
    canvas.create_line(x-35,y+7,x-45,y,x-45,y+15,x-35,y+7,width=2, tags='ryba')
    canvas.create_text(x-20,y-20,text=cislo,fill='white')

def klik(suradnice):
    x = suradnice.x
    y = suradnice.y
    global pocet_ryb, minimum
    minimum = int(entry1.get())
    if pocet_ryb<10 and x<500:
        pocet_ryb = pocet_ryb+1
        kresli_rybuA(x,y,pocet_ryb+minimum)
        kresli_rybuB(1000-x,y,pocet_ryb+minimum)

pocet_ryb=0
minimum=0
canvas.create_line(500,0,500,650,width=5)

canvas.bind('<Button-1>',klik)
entry1 = tkinter.Entry()
entry1.pack()
```

3. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def klik(suradnice):
    x = 500
    y = suradnice.y
    canvas.create_text(x,y,text=entry1.get(), font='Arial 30',tags='titulky')

def pusti():
    global pocet_posunov
    canvas.move('titulky',0,-10)
    pocet_posunov=pocet_posunov+1
    if pocet_posunov>2*65:
        pocet_posunov=0
        canvas.move('titulky',0,2*650)
    canvas.after(100,pusti)

def start():
    canvas.move('titulky',0,650)
    pusti()

def zmaz():
    canvas.delete('all')

pocet_posunov=0
entry1 = tkinter.Entry()
entry1.pack()
```

```
button1 = tkinter.Button(text='Pusti titulky',command=start)
button1.pack()
button2 = tkinter.Button(text='Zmaž',command=zmaž)
button2.pack()

canvas.bind('<Button-1>',klik)
```

Test 6303

1. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

def lampa(x,y,cislo):
    canvas.create_rectangle(x-10,y,x+10,y-40,fill='grey')
    canvas.create_line(x,y-40,x,y-80,fill='grey',width=5)
    canvas.create_oval(x-10,y-80,x+10,y-100,fill='yellow')
    canvas.create_text(x,y-20,text=cislo)

def lampa_odraz(x,y,cislo):
    canvas.create_rectangle(x-10,y,x+10,y+40,fill='grey')
    canvas.create_line(x,y+40,x,y+80,fill='grey',width=5)
    canvas.create_oval(x-10,y+80,x+10,y+100,fill='yellow')
    canvas.create_text(x,y+20,text=cislo,angle=180)

def klik(suradnice):
    global pocet
    pocet=pocet+1
    x=suradnice.x
    y=suradnice.y
    if y<325 and pocet<10:
        lampa(x,y,pocet)
        lampa_odraz(x,325+325-y,pocet)

canvas.create_rectangle(0,0,1000,325,fill='green')
canvas.create_rectangle(0,325,1000,650,fill='blue')

pocet=0
canvas.bind('<Button-1>',klik)
```

2. úloha

```
import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

maxy=650

def panelak(x,vyska,farba):
    canvas.create_rectangle(x,maxy-vyska*20,x+100,maxy,fill=farba,tags='panelak')
    y=maxy
    for i in range(vyska):
        canvas.create_rectangle(x+10,y-2,x+90,y-18,fill='skyblue',tags='panelak')
        y=y-20

def klik(suradnice):
    x=suradnice.x
    vyska=int(entry1.get())
    farba=choice(('gray1','gray2','gray'))
    panelak(x,vyska,farba)

def animuj():
```

```

global vystavba
canvas.move('panelak',0,-5)
vystavba=vystavba+5
if vystavba>=650:
    vystavba=0
    canvas.move('panelak',0,maxy)
canvas.after(100,animuj)

def start():
    canvas.move('panelak',0,maxy)
    animuj()

def zmaz():
    canvas.delete('all')

vystavba=0
canvas.bind('<Button-1>',klik)
button1=tkinter.Button(text='spusti výstavbu',command=start)
button1.pack()
button2=tkinter.Button(text='zmaž',command=zmaz)
button2.pack()
entry1=tkinter.Entry()
entry1.pack()

```

3. úloha

```

import tkinter
from random import *
canvas = tkinter.Canvas(width=1000,height=650)
canvas.pack()

maxy=650

def prekresli_cln():
    canvas.delete('cln')
    canvas.create_oval(cln_x-15,cln_y-30,cln_x+15,cln_y+30,fill='yellow',outline='red',width=10,tags='cln')

def povoden():
    global povoden_y
    povoden_y=povoden_y+5
    canvas.delete('voda')
    canvas.create_rectangle(0,0,1000,povoden_y,fill='blue',tags='voda')
    prekresli_cln()
    prekresli_osobu()
    testuj_zachranu()
    if not (osoba_y<povoden_y and zachraneny==0):
        canvas.after(500,povoden)
    if osoba_y<povoden_y and zachraneny==0:
        canvas.create_text(500,325,text='Smola :(',font='Arial 30',fill='red')

def prekresli_osobu():
    canvas.delete('osoba')
    canvas.create_oval(osoba_x-10,osoba_y-10,osoba_x+10,osoba_y+10,tags='osoba',fill='black')

def testuj_zachranu():
    global zachraneny
    if cln_x-10<osoba_x<cln_x+10 and cln_y-25<osoba_y<cln_y+25:
        canvas.create_text(500,325,text='Ďakujem! Zachránil si ma!',font='Arial 30',fill='red')
        zachraneny=1

cln_x=randrange(970)
cln_y=600
zachraneny=0

```



```
prekresli_cln()
osoba_x=randrange(950)
osoba_y=randint(100,600)
prekresli_osobu()
povoden_y=0
povoden()

def vlavo(event):
    global osoba_x,cln_x
    osoba_x=osoba_x-5
    if zachraneny==1:
        cln_x=cln_x-5
        prekresli_cln()
    prekresli_osobu()
    testuj_zachranu()

def vpravo(event):
    global osoba_x,cln_x
    osoba_x=osoba_x+5
    if zachraneny==1:
        cln_x=cln_x+5
        prekresli_cln()
    prekresli_osobu()
    testuj_zachranu()

def hore(event):
    global osoba_y,cln_y
    osoba_y=osoba_y-5
    if zachraneny==1:
        cln_y=cln_y-5
        prekresli_cln()
    prekresli_osobu()
    testuj_zachranu()

def dole(event):
    global osoba_y,cln_y
    osoba_y=osoba_y+5
    if zachraneny==1:
        cln_y=cln_y+5
        prekresli_cln()
    prekresli_osobu()
    testuj_zachranu()

canvas.bind_all('<Left>',vlavo)
canvas.bind_all('<Right>',vpravo)
canvas.bind_all('<Up>',hore)
canvas.bind_all('<Down>',dole)
```

1.3 Hodnotenie testov

V tabuľkách hodnotenia testov uvádzame jednotlivé úkony, za ktoré prideliujeme body. Počet pridelených bodov k danému úkonu je vo vedľajšom stĺpci (teda v našich konkrétnych prípadoch vždy jeden bod). Ak hodnotíme niektorý úkon viac ako jedným bodom, je tento úkon rozpísaný v dvoch riadkoch. Je to len z praktických dôvodov. Sme zvyknutí hodnotiť test písaním bodov elektronicky priamo v tabuľke a ukázalo sa nám, že pri rovnakej váhe každého úkonu eliminujeme niektoré omyly.

Častou chybou pri hodnotení býva tendencia za ťažšie veci pridať viac bodov a za veľmi ľahké naopak bodov málo. V takom prípade sa však môže stať, že študent s vedomosťami na stupeň 3 nemal šancu túto známku získať, pretože stratil veľké množstvo bodov za ťažké veci, ktoré má vedieť len jednotkár a ku ktorým sa trojkár ani len nedostane (napríklad, ak jednotkárské učivo tvorí až 40 % z celkového hodnotenia, začína na úspešnosti 60 %, čo je polovica intervalu pre stupeň 3 - stupnicu sme uviedli v úvode). Takéto hodnotenie môže byť diskriminujúce pre slabších žiakov.

Naopak, v našom hodnotení sa snažíme dbať na to, aby úkony s rôznou náročnosťou mali rovnakú váhu, alebo niekedy dokonca ľahší úkon môže byť za viac bodov ako ťažší. Pri príprave a nastavení hodnotenia najmä polročného testu je žiaduce si vopred stanoviť, ktoré vedomosti požadujeme od jednotkára, dvojkára, trojkára a štvorkára. Za preukázanie týchto vedomostí v príprave hodnotenia testu priradíme body a spočítame, či to, čo požadujeme napr. na známku 4, podľa nastaveného bodovania aj stačí na získanie tejto známky. Inak povedané, ak bodujeme ťažšie veci, ktoré požadujeme len od jednotkára, nemali by tieto poznatky v percentuálnej úspešnosti tvoriť viac ako 15 % pri stupnici, kde je od 85 % známka 1.

Pri hodnotení môžeme prideliť aj čiastkové body. Napríklad pri inicializácii grafiky študent importoval modul `tkinter`, priradil canvas, ale nenapísal príkaz `canvas.pack()`. Za inicializáciu grafiky môžeme pridať 0,7 bodu namiesto 0. Na tomto príklade je vidieť, že pri riešení programu sa môže študent zaseknúť na neznámych vedomostiach, a teda nevie vyriešiť zvyšok úlohy.

Je vhodné oboznámiť študentov so systémom hodnotenia a učiť ich, že celý problém si majú rozdeliť na menšie podproblémy a tie riešiť jednotlivo a tým získať body. Ak študent nevie inicializovať grafiku, stále môže napísať zvyšné príkazy a súradnice určiť kreslením na papier. Alebo ak nevie zapísať test, či zachránil zlatú ryбку (test 6301 úloha 3), stále môže zapísať príkaz, ktorý vypíše informáciu o chytení zlatej ryбки, nastaví veľkosť písma a farbu. Takto získa body za úkony, ktoré vie vyriešiť. K tomuto je tiež potrebné študentov viesť.

Je na zváženie, či študentom dovoliť na teste používať help, ktorý je súčasťou Pythonu, alebo nie. Má to aj svoje úskalia. Na jednej strane aj pracujúci programátor bežne používa pri svojej práci help a internet, no na druhej strane potrebuje poznať základnú množinu bežne používaných príkazov a nevyhľadáva ich stále v helpe alebo na internete. Povolenie takýchto pomôcok odporúčame zvážiť podľa typu skupiny, ktorú učíme.

Test 5101 AB

úkon	počet bodov
inicializácia grafiky	1
nakreslenie podkladového obdĺžnika	1
nastavenie farby	1
nakreslenie vnútorného obdĺžnika	1
nastavenie farby	1
použitie premennej	1
random hodín	1
správny interval	1
zapamätanie hodín do premennej	1
výpis textu (hodiny)	1
výpis textu (minúty)	1
výpis z premennej	1
farba výpisu	1
nastavenie veľkosti písma	1
nakreslenie dvojbodky	1
nakreslenie mesiaca	1
vymazanie jeho časti	1
nastavenie farby	1
celkový výzor (zodpovedá predlohe)	1
nakreslenie podstavy lampy	1
nastavenie farby	1
nakreslenie vodorovnej tyče lampy	1
nakreslenie zvislej tyče lampy	1
nastavenie farby	1
správne poradie príkazov pri kreslení (prekryvanie)	1
nakreslenie tienidla lampy	1
nastavenie farby tienidla	1
vymazanie časti oválu	1
vymazanie šikmou čiarou	1
nakreslenie svetla	1
nakreslenie vypínača	1
nastavenie farby	1
žrebovanie výšky vypínača	1
správny interval žrebovania	1
výška vypínača je podľa žrebovania	1
celkový výzor (zodpovedá predlohe)	1
SPOLU	36

Test 5201 AB

úkon	počet bodov
inicializácia grafiky	1
vytvorenie funkcie	1
nadpis	1
nastavenie veľkosti a fontu	1
nastavenie farby	1
for cyklus	1
správny počet opakovaní	1
kreslenie jednej perforácie	1
nastavenie farby	1
hrúbka čiar / veľkosť okienka	1
kreslenie na relatívnych súradniciach	1
kreslenie v cykle	1
celkový výsledok	1
for cyklus	1
správny počet opakovaní	1
kreslenie druhej perforácie	1
kreslenie na relatívnych súradniciach	1
kreslenie v cykle	1
celkový výsledok	1
kreslenie filmových políčok	1
for cyklus	1
kreslenie obdĺžnika	1
nastavenie farby	1
kreslenie na relatívnych súradniciach	1
kreslenie v cykle	1
správne nakreslenie podľa predlohy	1
import modulu random	1
random x a y pre kreslenie nuly	1
nakreslenie nuly	1
nastavenie farby	1
posun nuly od začiatku políčka	1
kreslenie v cykle	1
celkový výsledok nakreslenia nuly - podľa zadania	1
random x a y pre kreslenie jednotky	1
nakreslenie jednotky	1
nastavenie farby	1
posun jednotky od začiatku políčka	1
kreslenie v cykle	1
celkový výsledok nakreslenia jednotky - podľa zadania	1
SPOLU	39

Test 5301

úkon	počet bodov
inicializácia grafiky	1
importovanie modulu random	1
nastaveniu Canvas-u	1
nakreslenie rieky pomocou čiary	1
nastavenie hrúbky čiary	1
nastavenie farby	1
šikmosť čiary podľa predlohy	1
nakreslenie hornej časti mosta (obdĺžnik / čiara)	1
nastavenie farby	1
most - správny rozmer, aby bol cez celú rieku	1
cyklus na oblúky	1
počet opakovaní	1
nakreslenie kruhu - časť oblúka	1
nakreslenie obdĺžnika - spodná časť oblúka	1
kreslenie na relatívnych súradniciach	1
nastavenie farby	1
nastavenie outline na prázdne	1
celkový výsledok	1
zábradlie cyklus	1
zábradlie čiara	1
kreslenie čiary cyklom	1
správne umiestnenie zábradlia	1
zmenšovanie y v cykle	1
nakreslenie tabuľa - obdĺžnik	1
správna veľkosť tabule	1
nastavenie farby okraja	1
nastavenie hrúbky okraja	1
nastavenie farby výplne	1
nadpis Next Entry	1
nastavenie farby	1
vycentrovanie v rámci obdĺžnika	1
žrebovanie hodín	1
správny interval	1
zapamätanie do premennej	1
zmena hodín v cykle	1
cyklus	1
výpis	1
nastavenie farby a veľkosti	1
kreslenie celej hodiny a polhodiny	1
kreslenie v cykle	1
žrebovanie skupiny	1
zapamätanie do premennej	1
zvyšovanie čísla skupiny	1
v cykle	1
výpis čísla skupiny	1

nastavenie farby a veľkosti	1
nakreslenie okraja značky	1
nastavenie farby - červená	1
nastavenie farby - biele vnútro	1
nakreslenie čiary krížom	1
nastavenie farby	1
oranžový obdĺžnik	1
biely obdĺžnik	1
červený obdĺžnik	1
nastavenie farieb obdĺžnikov cigarety	1
vytvorenie funkcie na kreslenie hlavy	1
použitie parametrov	1
kruh hlavy - nakreslenie	1
náhodná farba zo zoznamu	1
cyklus na vlasy	1
počet opakovaní	1
žrebovanie náhodnej farby pred cyklom	1
náhodná farba vlasov	1
hrúbka čiar	1
žrebovanie vrchu vlasov	1
správne umiestnenie na hlave hore	1
žrebovanie spodku vlasov	1
správne umiestnenie na hlave dole	1
celkový výsledok	1
cyklus - kreslenie davu	1
žrebovanie súradníc	1
nakreslenie davu	1
používanie funkcií v programe	1
SPOLU	73

Test 5302

úkon	počet bodov
inicializácia grafiky	1
importovanie modulu random	1
nastaveniu Canvas-u	1
cyklus na stípy	1
počet opakovaní	1
nakreslenie obdĺžnika / čiary (stípy)	1
kreslenie na relatívnych súradniciach	1
nastavenie farby	1
celkový výsledok	1
poradie kreslenia - tieň, stíp	1
tieň posun x - žrebovanie	1
tieň posun y - žrebovanie	1
zapamätanie do premennej	1
nakreslenie čiary	1
spodok stípu a tieňa na rovnakej pozícii	1
vrch stípa + posuny	1
nastavenie farby	1
kreslenie v cykle	1
nastavenie farby strechy chrámu	1
kreslenie strechy - 1. čiara	1
kreslenie strechy - 2. čiara	1
kreslenie strechy - 3. čiara	1
cyklus na schody	1
čiara na schody	1
kreslenie v cykle	1
zmenšovanie x - začiatku schodu	1
zväčšovanie x - konca schodu	1
počet opakovaní	1
tabuľa - nakreslenie obdĺžnika	1
správna veľkosť tabule	1
nastavenie farby okraja	1
nastavenie hrúbky okraja	1
nastavenie farby výplne	1
nadpis Next Entry	1
nastavenie farby	1
vycentrovanie v rámci obdĺžnika	1
žrebovanie hodín	1
správny interval	1
zapamätanie do premennej	1
zmena hodín v cykle	1
cyklus	1
výpis	1
nastavenie farby a veľkosti	1
kreslenie celej hodiny a polhodiny	1
kreslenie v cykle	1

žrebovanie skupiny	1
zapamätanie do premennej	1
zvyšovanie čísla skupiny	1
v cykle	1
výpis čísla skupiny	1
nastavenie farby a veľkosti	1
nakreslenie okraja značky	1
nastavenie farby - červená	1
nastavenie farby - biele vnútro	1
nakreslenie čiary krížom	1
nastavenie farby	1
nakreslenie pohára - čiary	1
oblúk pohára - oval	1
vymazanie časti pohára	1
dokreslenie vrchu	1
vytvorenie funkcie na kreslenie hlavy	1
použitie parametrov	1
kruh hlavy - nakreslenie	1
náhodná farba zo zoznamu	1
cyklus na vlasy	1
počet opakovaní	1
žrebovanie náhodnej farby pred cyklom	1
náhodná farba vlasov	1
hrúbka čiar	1
žrebovanie vrchu vlasov	1
správne umiestnenie na hlave hore	1
žrebovanie spodku vlasov	1
správne umiestnenie na hlave dole	1
celkový výsledok	1
cyklus - kreslenie davu	1
žrebovanie súradníc	1
nakreslenie davu	1
používanie funkcií v programe	1
SPOLU	78

Test 6101A

úkon	počet bodov
inicializácia grafiky	1
nastavenie veľkosti a farby Canvas-u	1
nakreslenie čiary	1
umiestnenie čiary v strede a nastavenie hrúbky	1
kliknutie ľavým tlačidlom myši - bind	1
procedúra na ľavé tlačidlo	1
ľavé tlačidlo - súradnice - zistenie X	1
ľavé tlačidlo - súradnice - zistenie Y	1
uhol - zistenie podľa súradnice	1
výpis textu	1
na súradnici kliknutia	1
nastavenie farby	1
natočenie textu	1
if - či som v jednej časti	1
správna reakcia na if	1
opačný uhol natočenia textu	1
nastavenie farby	1
kreslenie textu	1
na zadanej súradnici	1
celkovo - správna reakcia na if	1
kliknutie pravým tlačidlom myši - bind	1
procedúra na pravé tlačidlo	1
pravé tlačidlo - súradnice - zistenie X	1
pravé tlačidlo - súradnice - zistenie Y	1
for cyklus	1
štvorček - kreslenie	1
kreslenie v cykle	1
počet nakreslených štvorčekov	1
nastavenie veľkosti a rozostupov	1
02 úloha	
random modulu import + použitie randomu	1
inicializácia grafiky	1
nastavenie veľkosti Canvas-u	1
vytvorenie funkcie	1
žrebovanie x / nastavenie x	1
žrebovanie y / nastavenie y	1
rozsah / farba	1
podmienka	1
podmienka	1
if	1
kreslenie oválu	1
nastavenie farby	1
v správnej vetve if-u	1
celkový vzhľad	1
prvé volanie časovača	1
ďalšie volanie časovača	1
SPOLU	45

Test 6201A

úkon	počet bodov
inicializácia grafiky	1
nastavenie veľkosti Canvas-u	1
tlačidlo štart - vytvorenie, text a príkaz	1
tlačidlo zastav / pokračuj - vytvorenie, text a príkaz	1
entry1 - vytvorenie	1
nakreslenie šedej cesty - vodorovnej	1
nakreslenie šedej cesty - zvislej	1
cyklus na vodorovnú stredovú čiaru	1
kreslenie bielych obdĺžnikov - vodorovná stredová čiara	1
nastavenie farby obdĺžnikov	1
správna veľkosť	1
správna pozícia	1
kreslenie v cykle	1
cyklus na zvislú čiaru	1
kreslenie bielych obdĺžnikov - zvislá stredová čiara	1
nastavenie farby obdĺžnikov	1
správna veľkosť	1
správna pozícia	1
kreslenie v cykle	1
nakreslenie autobusu	1
pozícia autobusu v premennej	1
použitie funkcií	1
tags autobusu / alebo niečo, čo rieši pohyb	1
čítanie vstupu z Entry	1
prevod na číslo	1
kreslenie vagónu	1
kreslenie vagónu	1
nastavenie farby	1
kreslenie na relatívnych súradniciach	1
cyklus	1
podľa počtu vagónov	1
vykreslenie vagónov	1
tags vagónov	1
súradnica vlaku - premenná	1
spustenie animácie	1
canvas.after	1
zvyšovanie y autobusu	1
move autobusu	1
test spodného okraja obrazovky - pre autobus	1
presunutie hore	1
test kolízie s vlakom	1
nepohybovanie v kolízii a následné pokračovanie autobusu	1
zmenšovanie súradnice vlaku	1
move vlaku	1
test okraja	1

test podľa konca vlaku	1
presun vlaku na opačný okraj	1
zmena aj v premennej	1
pozastavenie	1
pokračovanie	1
SPOLU	50

Test 6202

úkon	počet bodov
inicializácia grafiky	1
nastavenie veľkosti Canvas-u	1
entry1 - vytvorenie	1
entry2 - vytvorenie	1
načítanie krajiny	1
načítanie čísla	1
int čísla	1
vplyv vstupu na dĺžku dresu	1
udalosť klik	1
vytvorenie funkcie	1
zistenie súradníc	1
ovál - kreslenie	1
obdĺžnik - kreslenie	1
krajina - výpis	1
číslo - výpis	1
nastavenie farieb	1
kreslenie na mieste kliknutia	1
premenná na počet hráčov 1. družstva	1
premenná na počet hráčov 2. družstva	1
použitie global	1
zvýšenie premennej	1
if či som neprekročil stanovený počet	1
reakcia pravé tlačidlo	1
nastavenie modrej farby pre pravé tlačidlo	1
02 úloha	
inicializácia grafiky	1
entry1 - vytvorenie	1
načítanie vstupu z entry	1
premenná na číslo zo vstupu	1
zobrazenie čísla	1
nastavenie farby a veľkosti	1
tlačidlo1 - vytvorenie, text a príkaz	1
tlačidlo2 - vytvorenie, text a príkaz	1
funkcia štart	1
použitie global	1
odpočítavanie času	1
časovač použitie	1
volanie canvas.after	1
test nuly	1
výpis oznamu	1
zastavenie časovača	1
pauza	1
pokračovanie	1
SPOLU	42

Test 6301

úkon	počet bodov
01 Titulky	
inicializácia grafiky	1
nastavenie Canvas-u	1
entry - vytvorenie	1
tlačidlo - Pusti Titulky - vytvorenie, text, príkaz	1
tlačidlo - zmaž - vytvorenie, text, príkaz	1
reakcia na ľavý klik	1
funkcia na ľavý klik	1
vstupné parametre	1
spracovanie súradnice y	1
ignorovanie súradnice x	1
písanie textu	1
na mieste kliknutia	1
väčšie písmo	1
tagovanie alebo iné riešenie	1
entry.get	1
do textu	1
pustenie tituliek - posun dole	1
spustenie animácie	1
časovač na konci	1
posúvanie objektov hore	1
if mimo plochy	1
nastavenie na nové miesto	1
zmazanie všetkého	1
02 Rolety	
inicializácia grafiky	1
nastavenie Canvas-u	1
modré pozadie	1
kopec1	1
kopec2	1
kopec3	1
nastavenie farby	1
kopce celkový výsledok	1
okno for cyklus	1
rám okna - nakreslenie	1
nastavenie farby	1
roleť ľavá	1
roleť stredná	1
roleť pravá	1
tags	1
tags	1
funkcie	1
cyklus na roletu	1
obdĺžniky	1
relatívne kreslenie v cykle	1

nastavenie farby výplne	1
nastavenie farby okrajov	1
veľkosť častí roliet	1
počet opakovaní	1
správne vykreslenie všetkých troch roliet	1
reakcia na kláves 1	1
reakcia na kláves 2	1
reakcia na kláves 3	1
zapamätanie aktuálne ovládanej rolety	1
priradenie na začiatku - štandardne prvá roleta	1
funkcie po zmene čísla	1
nastavenie aktuálneho čísla rolety	1
použitie global	1
reakcia na šípku hore	1
reakcia na šípku dole	1
funkcia hore	1
funkcia dole	1
použitie global	1
testy aktuálne ovládanej rolety	1
testy aktuálne ovládanej rolety	1
ovládanie aktuálnej rolety	1
ovládanie aktuálnej rolety	1
posun hore	1
posun dole	1
03 Zachráň rybu	
inicializácia grafiky	1
import modulu random	1
nastavenie Canvas-u	1
modrý obdĺžnik zvislý - nakreslenie	1
náhodné miesto	1
náhodná šírka	1
zapamätanie do premennej	1
správne vykreslenie	1
žrebovanie vľavo alebo vpravo	1
test na to, v ktorej časti sa kreslí (vľavo od rieky alebo vpravo)	1
podľa if-u (v ktorej časti) žrebovanie súradnice	1
žrebovanie súradníc	1
vyžrebovanie správneho miesta (len mimo rieky)	1
zapamätanie miesta do premennej	1
funkcia rybka	1
kreslenie oválu	1
nastavenie hrúbky a farby elipsy	1
oko - kreslenie	1
nastavenie farby oka	1
kreslenie čiary - chvost	1
kreslenie čiary - chvost	1

nastavenie hrúbky a farby	1
nakreslenie podľa súradníc	1
kreslenie na relatívnych súradniciach	1
reakcia na vľavo	1
funkcia vľavo	1
reakcia na vpravo	1
funkcia vpravo	1
posunutie ryby vľavo	1
posunutie ryby vpravo	1
prekreslenie ryby alebo zmena v súradniciach vľavo	1
prekreslenie ryby alebo zmena v súradniciach vpravo	1
použitie global	1
umiestnenie testu - záchrana ryby	1
test - zápis if-u	1
správnosť podmienky	1
reakcia ak áno	1
výpis oznamu	1
nastavenie farby a pozície	1
zmenšovanie rieky	1
časovač	1
zmenšenie šírky rieky v premennej	1
prekreslenie rieky	1
test úplného konca - vyschnutie rieky	1
výpis oznamu smola	1
nastavenie farby a pozície	1
SPOLU	113

Test 6302

úkon	počet bodov
01 Zachráň rybu	
inicializácia grafiky	1
import modulu random	1
nastavenie Canvas-u	1
modrý obdĺžnik zvislý - nakreslenie	1
náhodné miesto	1
náhodná šírka	1
zapamätanie do premennej	1
správne vykreslenie	1
žrebovanie vľavo alebo vpravo	1
test na to, v ktorej časti sa kreslí (vľavo od rieky alebo vpravo)	1
podľa if-u (v ktorej časti) žrebovanie súradnice	1
žrebovanie súradníc	1
vyžrebovanie správneho miesta (len mimo rieky)	1
zapamätanie miesta do premennej	1
funkcia rybka	1
kreslenie oválu	1
nastavenie hrúbky a farby elipsy	1
oko - kreslenie	1
nastavenie farby oka	1
kreslenie čiary - chvost	1
kreslenie čiary - chvost	1
nastavenie hrúbky a farby	1
nakreslenie podľa súradníc	1
kreslenie na relatívnych súradniciach	1
reakcia na vľavo	1
funkcia vľavo	1
reakcia na vpravo	1
funkcia vpravo	1
posunutie ryby vľavo	1
posunutie ryby vpravo	1
prekreslenie ryby alebo zmena v súradniciach vľavo	1
prekreslenie ryby alebo zmena v súradniciach vpravo	1
použitie global	1
umiestnenie testu - záchrana ryby	1
test - zápis if-u	1
správnosť podmienky	1
reakcia ak áno	1
výpis oznamu	1
nastavenie farby a pozície	1
celková funkčnosť programu a kreslenie	1
02 Akvárium	
inicializácia grafiky	1
nastavenie Canvas-u	1

hrubá čiara v strede	1
entry - vytvorenie	1
reakcia na ľavé tlačidlo	1
funkcia na ľavé tlačidlo	1
vstupné parametre	1
spracovanie x	1
spracovanie y	1
if vľavo	1
vetva pre áno	1
funkcia ryba	1
kreslenie ryby ovál	1
nastavenie hrúbky a farby oválu	1
oko - nakreslenie	1
nastavenie farby	1
čiara chvost	1
čiara chvost	1
nastavenie hrúbky a farby	1
nakreslenie podľa súradníc - na mieste kliknutia	1
kreslenie na relatívnych súradniciach	1
číslo nad rybou - výpis	1
nastavenie bielej farby	1
počet rýb - nastavenie na začiatku	1
použitie global	1
pričítanie pri kliku	1
zistenie hodnoty z entry	1
int	1
pričítanie vstupnej hodnoty k číslu ryby	1
if na max počet rýb	1
správnosť podmienky	1
kreslenie ak áno	1
výpočet zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
písanie čísla pre zrkadlovú rybu	1
03 Titulky	
inicializácia grafiky	1
nastavenie Canvas-u	1
entry - vytvorenie	1
tlačidlo - Pusti Titulky - vytvorenie, text, príkaz	1
tlačidlo - zmaž - vytvorenie, text, príkaz	1
reakcia na ľavý klik	1
funkcia na ľavý klik	1
vstupné parametre	1
spracovanie súradnice y	1
ignorovanie súradnice x	1

písanie textu	1
na mieste kliknutia	1
väčšie písmo	1
tagovanie alebo iné riešenie	1
entry.get	1
do textu	1
pustenie titulkov - posun dole	1
spustenie animácie	1
časovač na konci	1
posúvanie objektov hore	1
if mimo plochy	1
nastavenie na nové miesto	1
zmazanie všetkého	1
SPOLU	100

Test 6303

úkon	počet bodov
01 Pobrežie	
inicializácia grafiky	1
nastavenie Canvas-u	1
nakreslenie zelenej polovice	1
nakreslenie modrej polovice	1
reakcia na ľavé tlačidlo	1
funkcia na ľavé tlačidlo	1
vstupné parametre	1
spracovanie x	1
spracovanie y	1
if som hore	1
vetva ak áno	1
funkcia lampa	1
kreslenie oválu - lampa	1
tvar a farba oválu	1
držák lampy horný - kreslenie	1
nastavenie farby + bez orámovania	1
spodný stojan - nakreslenie	1
nastavenie farby + orámovanie	1
nakreslenie podľa súradníc	1
kreslenie na relatívnych súradniciach	1
číslo na stojane - výpis	1
umiestnenie čísla	1
počet lúčov nastavenie na začiatku	1
použitie global	1
pričítanie počtu lúčov pri kliku	1
if na max počet lúčov	1
správnosť podmienky	1
kreslenie ak áno	1
výpočet zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
kreslenie zrkadlovej pozície	1
písanie čísla	1
otočenie čísla	1
02 Výstavba	
inicializácia grafiky	1
nastavenie Canvas-u	1
entry - vytvorenie	1
spustiť výstavbu - tlačidlo - vytvorenie, text, príkaz	1
zmaž - tlačidlo - vytvorenie, text, príkaz	1
reakcia na ľavý klik	1
funkcia na ľavý klik	1
vstupné parametre	1

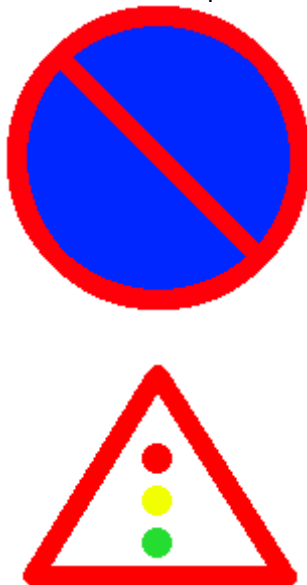
spracovanie x -ovej súradnice	1
ignorovanie y -ovej súradnice	1
výber náhodnej farby	1
nakreslenie obdĺžnika	1
výplň farbou	1
entry.get()	1
int	1
výpočet veľkosti podľa vstupu	1
cyklus	1
cyklus - počet opakovaní podľa vstupu	1
okno - kreslenie	1
nastavenie farby	1
okno kreslenie v cykle	1
kreslenie na relatívnych súradniciach	1
celkový výsledok nakreslenia domu	1
tags alebo niečo podobné - na posúvanie	1
kreslenie na mieste kliknutia	1
pustenie výstavby - posun dole	1
spustenie animácie	1
časovač na konci funkcie	1
posúvanie objektov hore	1
if celé ukázané	1
nastavenie na nové miesto	1
zmazanie všetkého	1
03 Záchranár	
inicializácia grafiky	1
import modulu random	1
nastavenie Canvas-u	1
modrý obdĺžnik záplava - nakreslenie	1
záplava nastavenie výšky na 0	1
človek kreslenie	1
náhodné miesto - žrebovanie x	1
náhodné miesto - žrebovanie y	1
zapamätanie do premennej x a y	1
nakreslenie podľa pozície	1
žrebovanie x pre čln	1
zapamätanie do premennej x	1
fixné y pre čln	1
oval - kreslenie	1
podľa súradníc	1
žltá časť	1
červený okraj	1
správne vykreslenie	1
reakcia na vľavo	1
funkcia vľavo	1
reakcia na vpravo	1

funkcia vpravo	1
reakcia na hore	1
funkcia hore	1
reakcia na dole	1
funkcia dole	1
posunutie vľavo	1
posunutie vpravo	1
posunutie hore	1
posunutie dole	1
prekreslenie alebo zmena v súradniciach vľavo a vpravo	1
prekreslenie alebo zmena v súradniciach hore a dole	1
použitie global	1
umiestnenie testu záchrany	1
test záchrany - zápis	1
správnosť podmienky	1
vetva ak áno	1
výpis oznamu	1
nastavenie farby a pozície	1
zväčšovanie záplavy	1
časovač	1
zväčšovanie v premennej	1
prekreslenie	1
test úplného konca - človek sa utopil	1
výpis smola	1
nastavenie farby a pozície	1
pamätanie si, že som na člne	1
if či som na člne	1
vetva ak áno	1
pohyb aj s člnom	1
pohyb aj s člnom všetky smery	1
SPOLU	117

2. Zadania ďalších testov

Test č. 2 A po kapitole 3	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5102A

01 Vytvorte program, ktorý nakreslí nasledovné dopravné značky:



Prémia Doplňte do programu nakreslenie tejto dopravnej značky:



KONIEC TESTU

Test č. 2 B po kapitole 3	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5102B

01 Vytvorte program, ktorý nakreslí nasledovné dopravné značky:



Prémia Doplňte do programu nakreslenie tejto dodatkovej tabule:



KONIEC TESTU

Test č. 2 C po kapitole 3	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy
Čas na vypracovanie:	45 minút
Variant:	C
Kód testu:	5102C

01 Vytvorte program, ktorý nakreslí nasledovné dopravné značky:



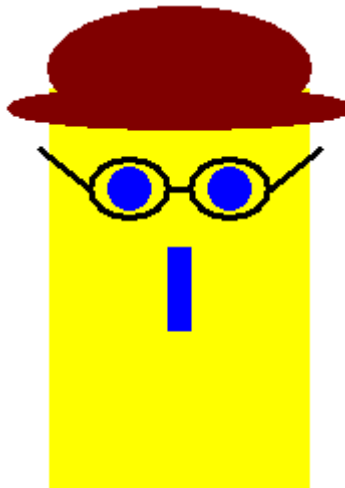
Prémia Doplňte do programu nakreslenie tejto dodatkovej tabule:



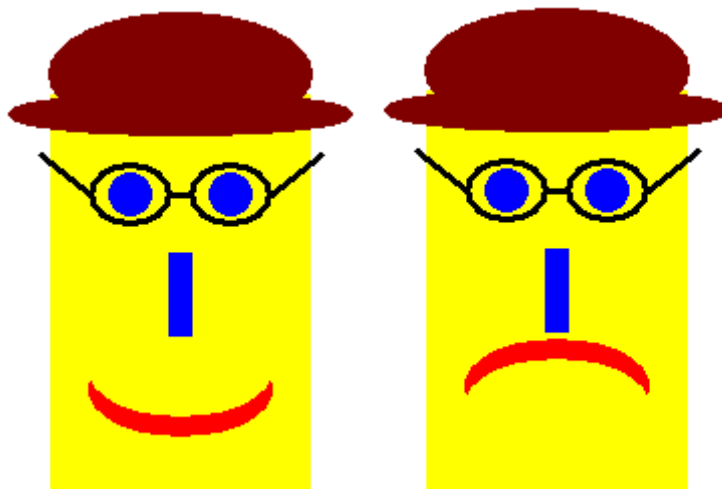
KONIEC TESTU

Test č. 3 A po kapitole 3	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy
Čas na vypracovanie:	45 minút
Varianta:	A
Kód testu:	5103A

-
- 01** Vytvorte program, ktorý nakreslí tvár podľa predlohy:



-
- 02** Upravte program tak, aby nakreslil veselú a smutnú tvár vedľa seba podľa obrázku:



KONIEC TESTU

Test č. 3 B po kapitole 3	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy
Čas na vypracovanie:	45 minút
Varianta:	B
Kód testu:	5103B

-
- 01** Vytvorte program, ktorý nakreslí tvár podľa predlohy:



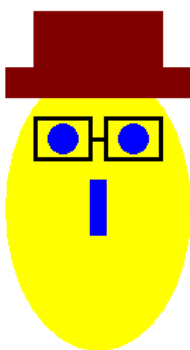
-
- 02** Upravte program tak, aby nakreslil smutnú a veselú tvár vedľa seba podľa obrázku:



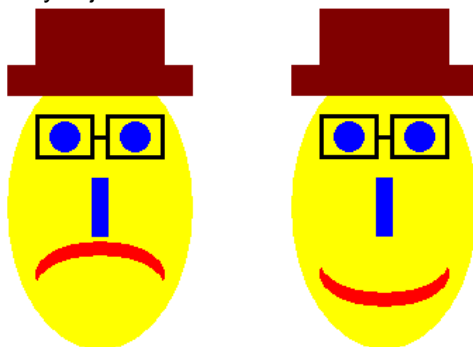
KONIEC TESTU

Test č. 2 A po kapitole 6	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5202A

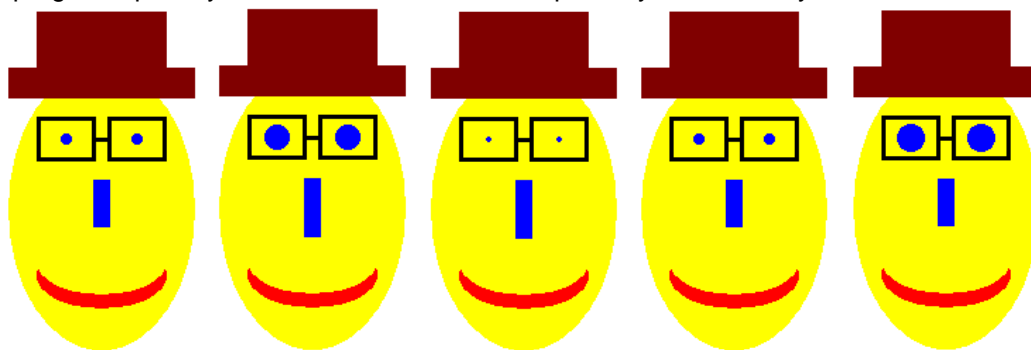
- 01** Vytvorte podprogram **Tvár**. Podprogram nakreslí na zadanom mieste tvár podľa predlohy:



- 02** Vytvorte ďalšie dve funkcie **Veselý** a **Smutný**, ktoré na zadanom mieste nakreslia veselú a smutnú tvár, pričom už využijú funkciu **Tvár**.



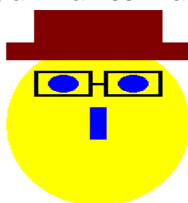
- 03** Vytvorte funkciu **NáhodnáTvár**, ktorá nakreslí na zadanom mieste tvár s náhodnou veľkosťou očí a s náhodnou dĺžkou nosa. Musí však platiť, že obe oči sú rovnako veľké. Doplňte do programu príkazy, ktoré nakreslia vedľa seba päť takýchto náhodných tvárí.



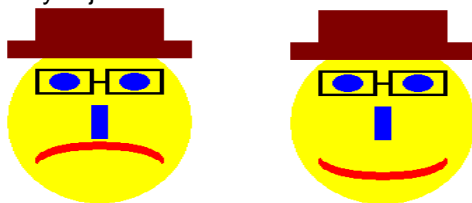
KONIEC TESTU

Test č. 2 B po kapitole 6	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5202B

- 01** Vytvorte podprogram **Tvár**. Podprogram nakreslí na zadanom mieste tvár podľa predlohy:



- 02** Vytvorte ďalšie dve funkcie **Veselý** a **Smutný**, ktoré na zadanom mieste nakreslia veselú a smutnú tvár, pričom už využijú funkciu **Tvár**.



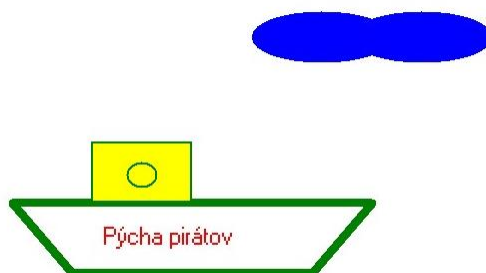
- 03** Vytvorte funkciu **NáhodnáTvár**, ktorá nakreslí na zadanom mieste tvár s náhodnou veľkosťou očí a s náhodnou dĺžkou nosa. Musí však platiť, že obe oči sú rovnako veľké. Doplňte do programu príkazy, ktoré nakreslia vedľa seba **päť** takýchto náhodných tvárí.



KONIEC TESTU

Test č. 3 A po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5203A

- 01** Vytvorte program, ktorý nakreslí loď s nápisom Pýcha pirátov. Vpravo nad loďou je mrak.



- 02** Upravte program tak, aby postupne nakreslil náhodný počet dažďových kvapiek. Dažďové kvapky budú umiestnené len pod mrakom a vo výške, ktorá nepresahuje hornú palubu lode.



- Prémia** Vytvorte kópiu programu, v ktorej sa bude kresliť mrak aj s dažďom na náhodnom mieste (bude v rovnakej nadmorskej výške, ale vo vodorovnom smere bude náhodne posunutý).

KONIEC TESTU

Test č. 3 B po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5203B

- 01** Vytvorte program, ktorý nakreslí loď s nápisom Pýcha pirátov. Vľavo nad loďou je mrak.



- 02** Upravte program tak, aby postupne nakreslil náhodný počet dažďových kvapiek. Dažďové kvapky budú umiestnené len pod mrakom a vo výške, ktorá nepresahuje hornú palubu lode.

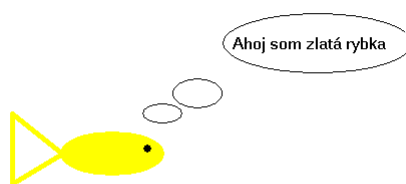


- Prémia** Vytvorte kópiu programu, v ktorej sa bude kresliť mrak aj s dažďom na náhodnom mieste (bude v rovnakej nadmorskej výške, ale vo vodorovnom smere bude náhodne posunutý).

KONIEC TESTU

Test č. 4 A po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	5204A

- 01** Vytvorte program, ktorý nakreslí zlatú ryбку a tri bubliny. V najväčšej bubline bude text „Ahoj, som zlatá rybka“.



- 02** Upravte program tak, aby postupne nakreslil potravu pre rybu. Potrava má tvar krížika a je náhodnej farby. Potrava sa kreslí len v dolnej časti obrazovky a jej počet je náhodný. Je tam však minimálne 20 ks krížikov a maximálne 200.



- Prémia** Vytvorte kópiu programu, v ktorej sa bude kresliť ryba s náhodne veľkým chvostom a okom. Chvost ale bude mať vždy tvar rovnoramenného trojuholníka.

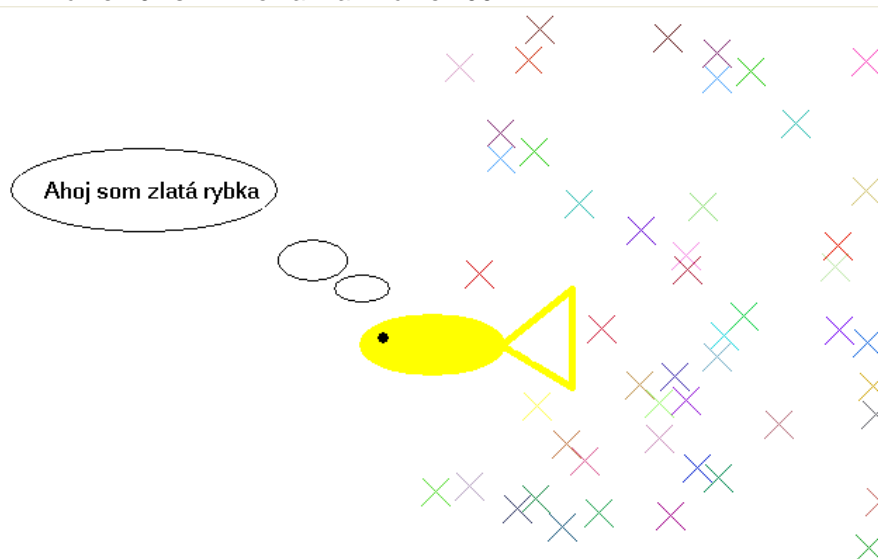
KONIEC TESTU

Test č. 4 B po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	5204B

- 01** Vytvorte program, ktorý nakreslí zlatú ryбку a tri bubliny. V najväčšej bubline bude text „Ahoj, som zlatá ryбка“.



- 02** Upravte program tak, aby postupne nakreslil potravu pre rybu. Potrava má tvar krížika a je náhodnej farby. Potrava sa kreslí len v pravej časti obrazovky a jej počet je náhodný. Je tam však minimálne 20 ks krížikov a maximálne 200.



- Prémia** Vytvorte kópiu programu, v ktorej sa bude kresliť ryba s náhodne veľkým chvostom a okom. Chvost ale bude mať vždy tvar rovnoramenného trojuholníka.

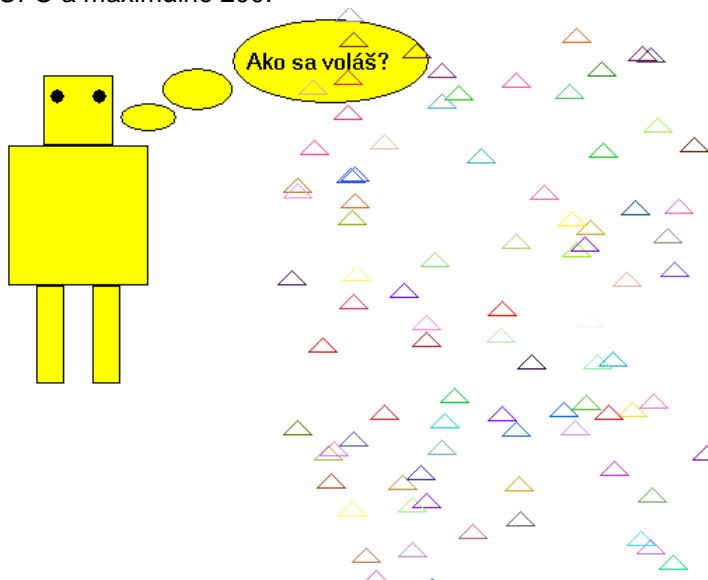
KONIEC TESTU

Test č. 4 C po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	C
Kód testu:	5204C

- 01** Vytvorte program, ktorý nakreslí robot a tri bubliny. V najväčšej bubline bude text „Ako sa voláš?“



- 02** Upravte program tak, aby postupne nakreslil UFO. UFO má tvar trojuholníka a je náhodnej farby. UFO sa kreslí len v pravej časti obrazovky a jeho počet je náhodný. Je tam však minimálne 20 ks UFO a maximálne 200.

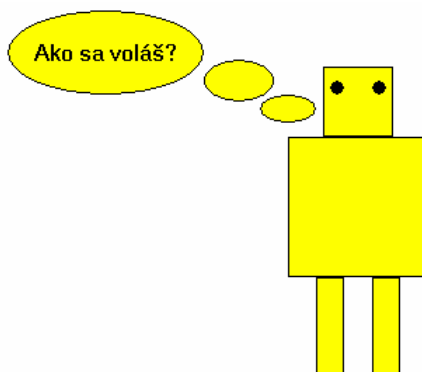


- Prémia** Vytvorte kópiu programu, v ktorom sa bude kresliť robot s náhodne veľkou hlavou a očami.

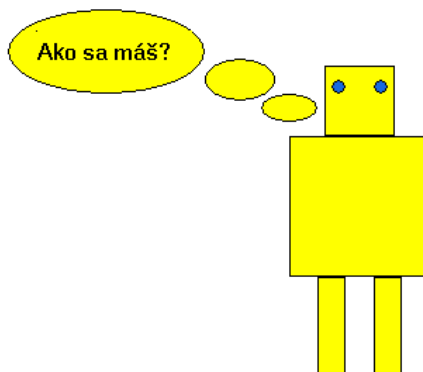
KONIEC TESTU

Test č. 4 D po kapitole 5	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus
Čas na vypracovanie:	45 minút
Variant:	D
Kód testu:	5204D

- 01** Vytvorte program, ktorý nakreslí robot a tri bubliny. V najväčšej bubline bude text „Ako sa voláš?“.



- 02** Upravte program tak, aby postupne nakreslil UFO. UFO má tvar trojuholníka a je náhodnej farby. UFO sa kreslí len v spodnej časti obrazovky a jeho počet je náhodný. Je tam však minimálne 20 ks UFO a maximálne 200.



- Prémia** Vytvorte kópiu programu, v ktorom sa bude kresliť robot s náhodne veľkou hlavou a očami.

KONIEC TESTU

Polročný test č. 3 A po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Varianta:	A
Kód testu:	5303A

01 Vytvorte program, ktorý nakreslí pohľadnicu z dovolenky. Program nakreslí nasledovné obrázky:

- Ostrov - v spodnej časti sa nakreslí hnedý ostrov.
- Postavu, ktorá má okrúhlu žltú hlavu. Na hlave sú modré oči. Telo postavy (šaty) tvorí 10 hrubších červených čiar, ktoré sú v pravidelných rozstupoch (využite for cyklus).
- Vlasy – na hlave postavy sa nakreslí náhodný počet vlasov. Jeden náhodný vlas je zelený a má hrúbku 4 body. Vlasy končia v náhodnom bode v okolí hlavy (nie hocikde v obrázku).
- Čajky sa nakreslia na náhodnej pozícii (nie nižšie ako je hlava postavy). Čajka je čierna a je zložená z dvoch čiar (čiernej farby).
- Na pohľadnici je nadpis Pozdrav z dovolenky. Nadpis sa napíše veľkosťou 20, červeným písmom Tahoma na bielom pozadí na spodnej časti ostrova.
- V ľavej časti sa kreslí cestovný poriadok - čas odchodu lodí z ostrova. Cestovný poriadok obsahuje hodiny od 10 do 17. Minúta odchodu v každej jednotlivéj hodine je náhodná od 10 do 59. Farba písma je čierna.



KONIEC TESTU

Polročný test č. 3 B po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	5303B

01 Vytvorte program, ktorý nakreslí pohľadnicu z dovolenky. Program nakreslí nasledovné obrázky:

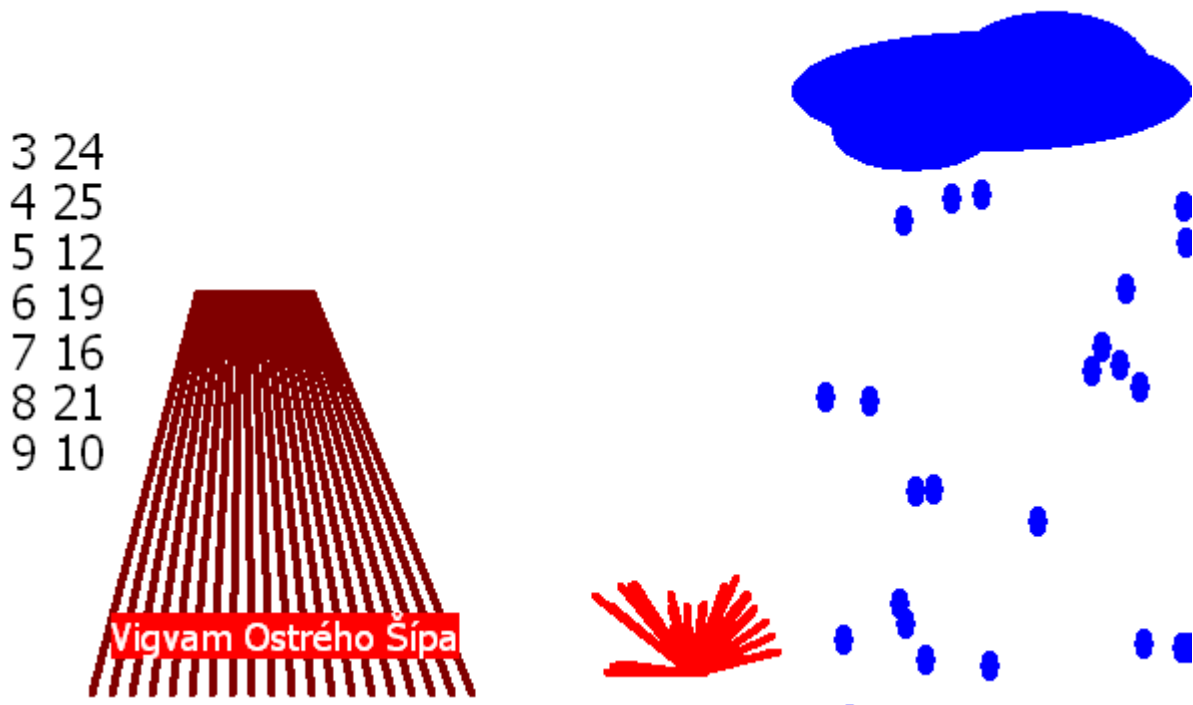
- Ostrov - v spodnej časti sa nakreslí hnedý ostrov.
- Postavu, ktorá má okrúhlu žltú hlavu. Na hlave sú modré oči. Telo postavy (šaty) tvorí 10 hrubších červených čiar, ktoré sú v pravidelných rozstupoch (využite for cyklus).
- Vlasy – na hlave postavy sa nakreslí náhodný počet vlasov. Jeden náhodný vlas je zelený a má hrúbku 4 body. Vlasy končia v náhodnom bode v okolí hlavy (nie hocikde v obrázku).
- Čajky sa nakreslia na náhodnej pozícii (nie nižšie ako je hlava postavy). Čajka je čierna a je zložená z dvoch čiar (čiernej farby).
- Na pohľadnici je nadpis Pozdrav z dovolenky. Nadpis sa napíše veľkosťou 20, červeným písmom Tahoma na bielom pozadí na spodnej časti ostrova.
- V ľavej časti sa kreslí cestovný poriadok - čas odchodu lodí z ostrova. Cestovný poriadok obsahuje hodiny od 12 do 19. Minúta odchodu v každej jednotlivéj hodine je náhodná od 10 do 59. Farba písma je čierna.



KONIEC TESTU

Polročný test č. 4 A po kapitole 7	
Typ testu:	Polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	A
Kód testu:	5304A

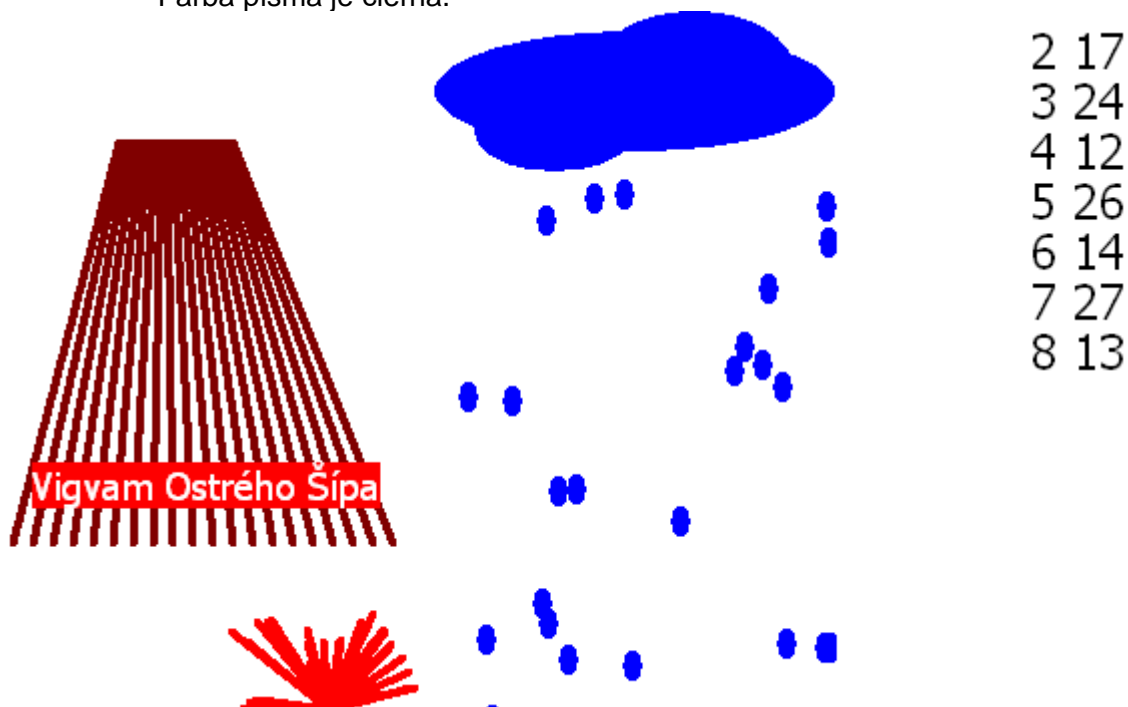
- 01** Vytvorte program, ktorý nakreslí pohľad do indiánskej osady. Program nakreslí:
- Modrý oblak vpravo hore.
 - Vigvam, ktorý pozostáva z 20 hrubších hnedých čiar, ktoré sú v pravidelných rozostupoch (využite for cyklus).
 - Oheň – nakreslí sa náhodný počet plameňov. Jeden náhodný plameň je červený a má hrúbku 4 body. Plamene končia v náhodnom bode v okolí ohniska (nie hocikde v obrázku).
 - Kvapky sa nakreslia na náhodnej pozícii v páse pod oblakom. Kvapka je modrá.
 - Na vigvame je nadpis „Vigvam Ostrého Šípa“. Nadpis sa napíše veľkosťou 14, bielym písmom Tahoma na červenom pozadí v spodnej časti vigvamu.
 - V ľavej časti sa napíše štatistika počtu upršaných dní. Najprv je v štatistike číslo mesiaca a za ním počet dní, v ktorých pršalo. Štatistika je za 3. až 9. mesiac. Počet upršaných dní v jednotlivom mesiaci je náhodný od 10 do 29. Farba písma je čierna.



KONIEC TESTU

Polročný test č. 4 B po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	5304B

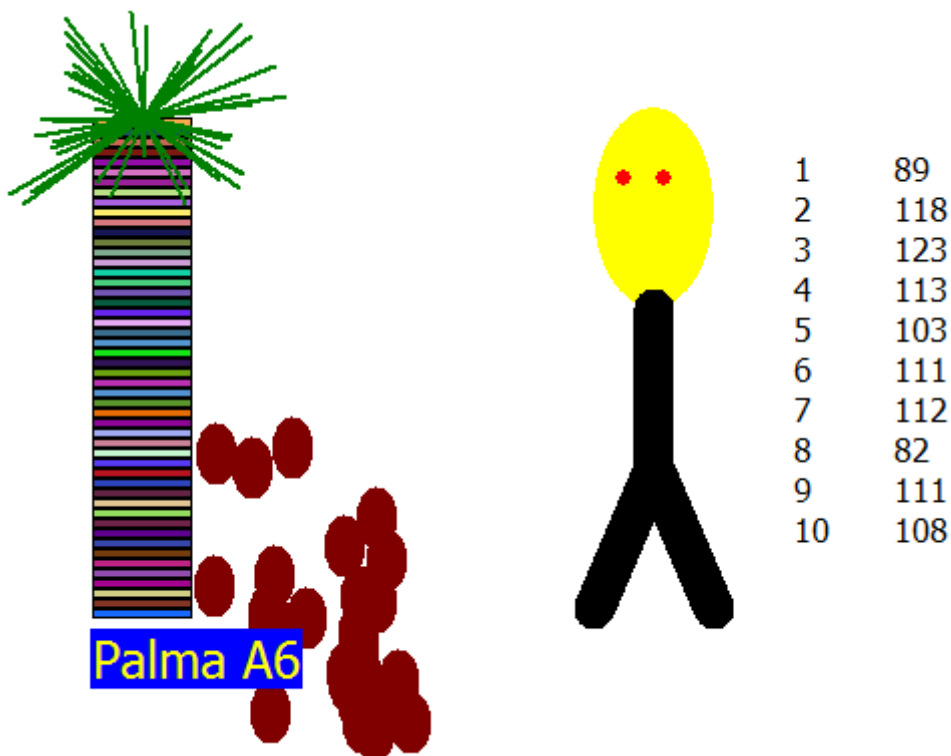
- 01** Vytvorte program, ktorý nakreslí pohľad do indiánskej osady. Program nakreslí:
- Modrý oblak hore v strede obrázku.
 - Vigvam, ktorý pozostáva z 20 hrubších hnedých čiar, ktoré sú v pravidelných rozostupoch (využite for cyklus).
 - Oheň – nakreslí sa náhodný počet plameňov. Jeden náhodný plameň je červený a má hrúbku 4 body. Plamene končia v náhodnom bode v okolí ohniska (nie hocikde v obrázku).
 - Kvapky sa nakreslia na náhodnej pozícii v páse pod oblakom. Kvapka je modrá.
 - Na vigvame je nadpis „Vigvam Ostrého Šípa“. Nadpis sa napíše veľkosťou 14, bielym písmom Tahoma na červenom pozadí v spodnej časti vigvamu.
 - V pravej časti sa napíše štatistika počtu upršaných dní. Najprv je v štatistike číslo mesiaca a za ním počet dní, v ktorých pršalo. Štatistika je za 2. až 8. mesiac. Počet upršaných dní v jednotlivom mesiaci je náhodný od 10 do 29. Farba písma je čierna.



KONIEC TESTU

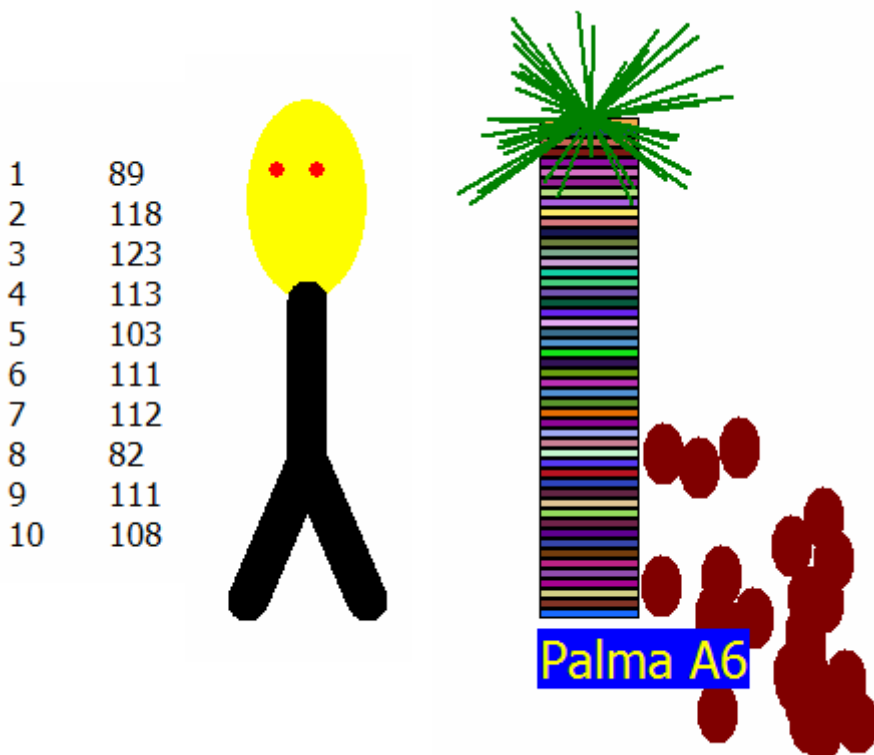
Polročný test č. 5 A po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	A
Kód testu:	5305A

- 01** Vytvorte program, ktorý nakreslí pohľad na orechovú plantáž. Program nakreslí:
- Postavu, ktorá je v strede. Postava človeka má žltú hlavu, červené oči, čierne nohy a telo.
 - Palma je vľavo. Kmeň palmy je zložený z tenkých obdĺžnikov náhodnej farby.
 - Náhodný počet listov je umiestnený na palme. Listy sú náhodné a kreslia sa len v blízkom okolí palmy. List je od hornej strany grafického plátna vzdialený minimálne 20 bodov, je zelený a má hrúbku 2 body.
 - Náhodný počet orechov sa kreslí vedľa palmy vpravo, ale len v jej spodnej časti. Orech je hnedý a má šírku 20 bodov a výšku 30.
 - Pod palmou je napísané jej označenie „Palma A6“. Označenie sa napíše veľkosťou 30, žltým písmom Tahoma na modrom pozadí.
 - Vpravo vedľa postavy sa napíše počet zozbieraných orechov z jednotlivých paliem. Prvé číslo je číslo palmy a vedľa nej počet orechov, ktorý je náhodný od 80 do 129. Farba písma je čierna a pozadie biele.



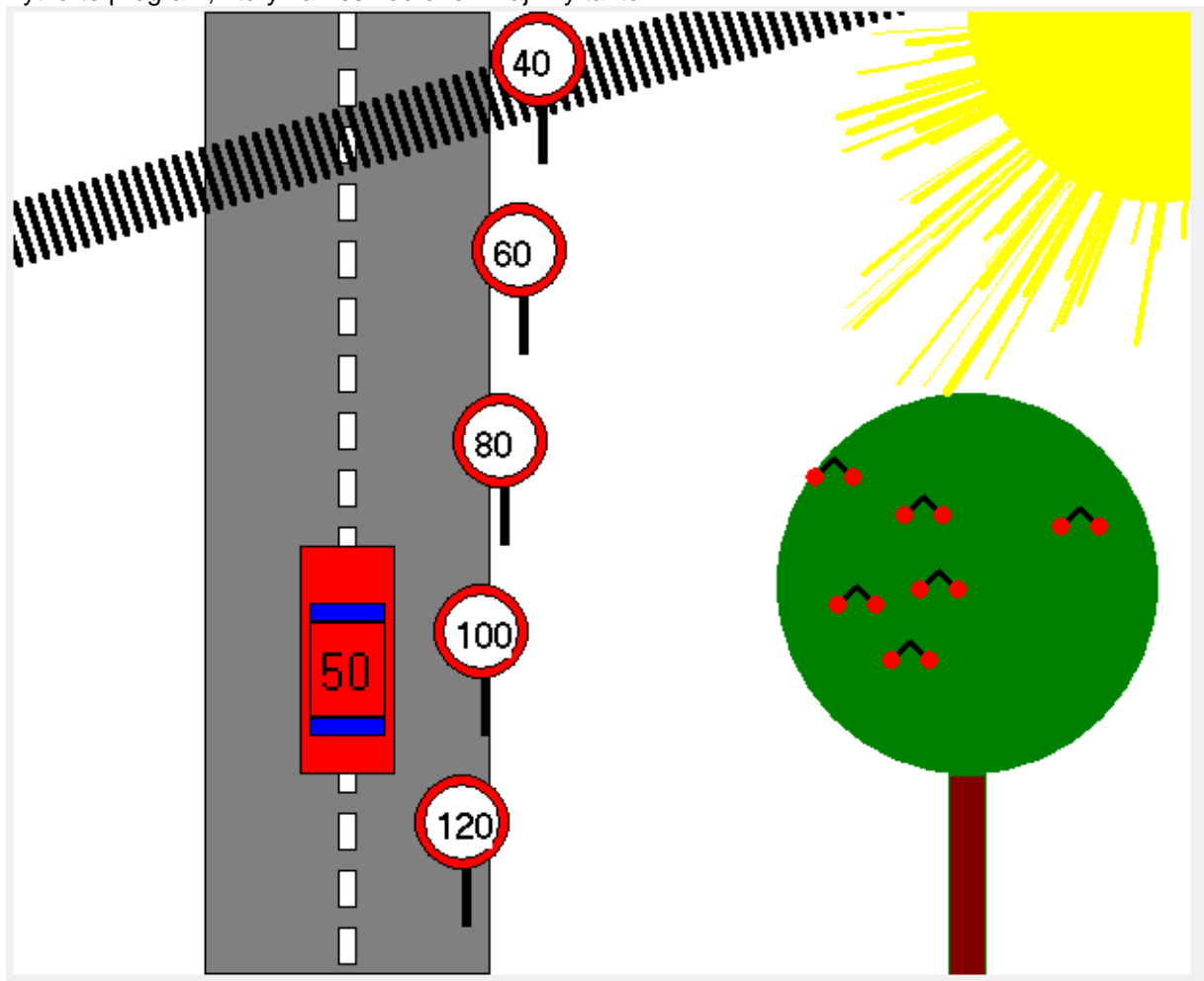
Polročný test č. 5 B po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	5305B

- 01** Vytvorte program, ktorý nakreslí pohľad na orechovú plantáž. Program nakreslí:
- Postavu, ktorá je v strede. Postava človeka má žltú hlavu, červené oči, čierne nohy a telo.
 - Palma je pravo. Kmeň palmy je zložený z tenkých obdĺžnikov náhodnej farby.
 - Náhodný počet listov je umiestnený na palme. Listy sú náhodné a kreslia sa len v blízkom okolí palmy. List je od hornej strany grafického plátna vzdialený minimálne 20 bodov, je zelený a má hrúbku 2 body.
 - Náhodný počet orechov sa kreslí vedľa palmy vpravo, ale len v jej spodnej časti. Orech je hnedý a má šírku 20 bodov a výšku 30.
 - Pod palmou je napísané jej označenie „Palma A6“. Označenie sa napíše veľkosťou 30, žltým písmom Tahoma na modrom pozadí.
 - Vľavo vedľa postavy sa napíše počet zozbieraných orechov z jednotlivých paliem. Prvé číslo je číslo palmy a vedľa nej počet orechov, ktorý je náhodný od 80 do 129. Farba písma je čierna a pozadie biele.



Polročný test č. 6 A po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	A
Kód testu:	5306A

01 Vytvorte program, ktorý nakreslí obrázok krajinky takto:



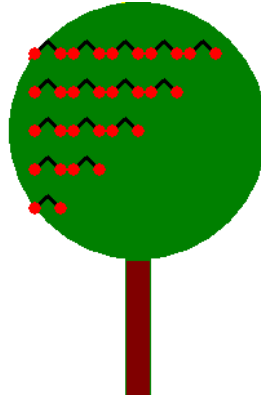
- V ľavej časti sa nakreslí šedá asfaltová cesta s bielou prerušovanou čiarou v jej strede (použite cyklus).
- Vpravo dole sa nakreslí strom, ktorý má zelenú korunu a hnedý kmeň.
- V programe bude funkcia čerešňa, ktorá na zadanom mieste nakreslí jednu dvojitú čerešňu, ktorá má čiernu stopku a červený plod. Využitím tejto funkcie nakreslite na korune stromu náhodný počet čerešní.
- V pravom hornom rohu sa nakreslí slnko, z ktorého náhodne vybieha 50 lúčov (použite cyklus).
- V strede cesty v jej dolnej časti sa nakreslí auto. Auto je červenej farby a predné a zadné okno je modré. Na streche auta je jeho číselné označenie, ktoré je desaťnásobkom náhodného čísla od 1 do 7.
- Pri ceste nakreslite pomocou cyklu päť značiek s označením maximálnej povolenej rýchlosti. Značky sú umiestnené v pravej časti cesty a sú umiestnené v čiastočne

šikmom smere. Značka je vo vnútri biela, okolo je hrubý červený rám, ktorý je orámovaný tenkou čiernou čiarou. Na značkách sú postupne tieto rýchlosti (od spodnej časti) 120, 100, 80, 60, 40.

- V hodnej časti je železničná trať - podvaly koľajníc, ktoré sú nakreslené šikmo naprieč cestou. Jednotlivé podvaly sú hrubšie čierne čiary (použite cyklus).

Prémia

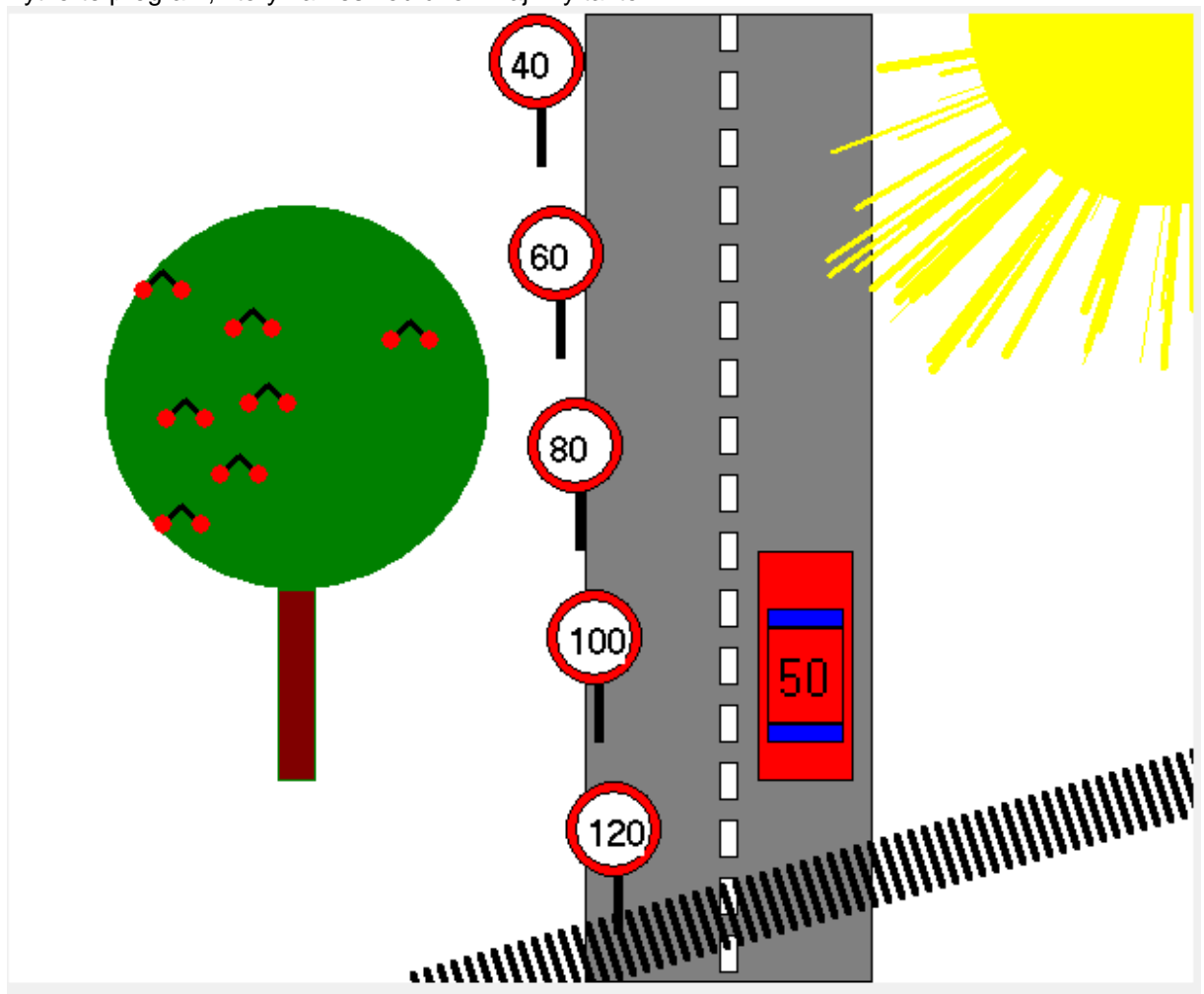
Vytvorte funkciu, ktorá nakreslí na strome takto usporiadané čerešne (použite cyklus):



KONIEC TESTU

Polročný test č. 6 B po kapitole 7	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4.1 a 4.2 Premenné a náhodné hodnoty 4.3 Kreslenie obrázkov na náhodnom mieste (kap 4.3) 5. Opakovanie časti programu – for cyklus 6. Vytváranie podprogramy 7. Úlohy na opakovanie I {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	5306B

01 Vytvorte program, ktorý nakreslí obrázok krajinky takto:

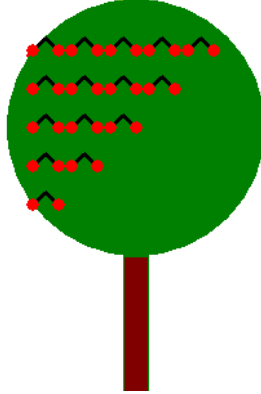


- V pravej časti sa nakreslí šedá asfaltová cesta s bielou prerušovanou čiarou v jej strede (použite cyklus).
- Vľavo dole sa nakreslí strom, ktorý má zelenú korunu a hnedý kmeň.
- V programe bude funkcia čerešňa, ktorá na zadanom mieste nakreslí jednu dvojitú čerešňu, ktorá má čiernu stopku a červený plod. Využitím tejto funkcie nakreslite na korune stromu náhodný počet čerešní.
- V pravom hornom rohu sa nakreslí slnko, z ktorého náhodne vybieha 50 lúčov (použite cyklus).
- V strede cesty v jej dolnej časti sa nakreslí auto. Auto je červenej farby a predné a zadné okno je modré. Na streche auta je jeho číselné označenie, ktoré je desaťnásobkom náhodného čísla od 3 do 9.

- Pri ceste nakreslite pomocou cyklu päť značiek s označením maximálnej povolenej rýchlosti. Značky sú umiestnené v ľavej časti cesty a sú umiestnené v čiastočne šikmom smere. Značka je vo vnútri biela, okolo je hrubý červený rám, ktorý je orámovaný tenkou čiernou čiarou. Na značkách sú postupne tieto rýchlosti (od spodnej časti) 120, 100, 80, 60, 40.
- V spodnej časti je železničná trať - podvaly koľajníc, ktoré sú nakreslené šikmo naprieč cestou. Jednotlivé podvaly sú hrubšie čierne čiary (použite cyklus).

Prémia

Vytvorte funkciu, ktorá nakreslí na strome takto usporiadané čerešne (použite cyklus):



KONIEC TESTU

Test č. 2 po kapitole 10	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač
Čas na vypracovanie:	45 minút
Variant:	-
Kód testu:	6102

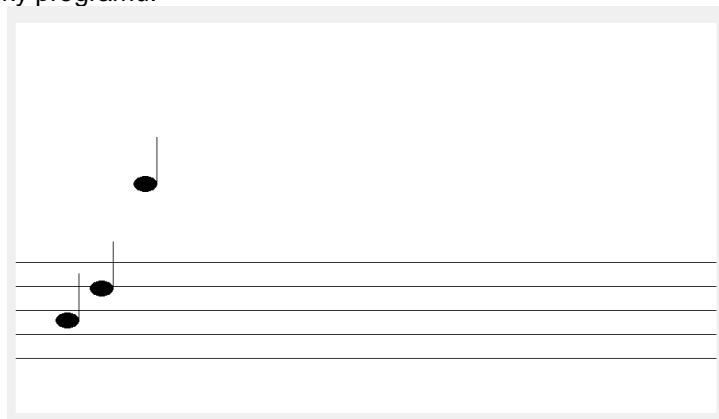
01 Vytvorte program náhodná melódia, ktorý bude pracovať takto:

- v spodnej časti je notová osnova,
- po stlačení klávesu medzera sa spustí padanie nôt,
- od vrchu obrazovky padá nota, ktorá sa zastaví na notovej osnove - na náhodnom mieste, no to miesto spĺňa pravidlá notového zápisu (nota je buď na čiare, alebo medzi čiarami),
- keď dopadne jedna nota, začne od vrchu padať ďalšia (kúsok vedľa poslednej), noty sú umiestnené v pravidelných rozstupoch (vedľa seba),
- program sa zastaví, až keď na osnove už nie je miesto pre ďalšiu notu,
- ak je na notovej osnove viac nôt rovnakého tónu, tak sú presne v rovnakej línii – napríklad, ak sú dve noty medzi prvou a druhou čiarou, tak sú v strede medzi čiarami osnovy.

Program pomáha aj neznalcom nôt takto:

- podľa toho, kde práve sme kurzorom myši na notovej osnove, nám program píše v hornej časti obrazovky tón, ktorý by sa zahral (čiže c1, d1, e1, f1 , g1, a1, h1, c2, d2, e2, f2)

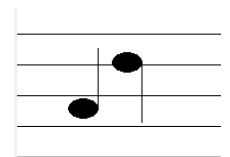
Ukážka obrazovky programu:



BONUS

Upravte program tak, že:

- sa správne píše palička noty – čiže ak je na tretej čiare alebo vyššie, píše sa palička smerom dole.



KONIEC TESTU

Test č. 3 A po kapitole 11	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole
Čas na vypracovanie:	45 minút
Variant:	A
Kód testu:	6203A

01 Vytvorte program, ktorý nám umožní „odtláčať“ do grafickej plochy „3-D“ farebný text takto:



- na miesto kliknutia v grafickej ploche sa napíše text, ktorý je zadaný v súčiastke Entry,
- text je zobrazený formou „3D“, nakreslí sa 5-krát tesne vedľa seba – každé ďalšie nakreslenie je posunuté o 1 bod dole a 1 bod vpravo,
- každé posunutie v „3D“ efekte je nakreslené náhodnou farbou zo zoznamu, ktorý si sami určíte.

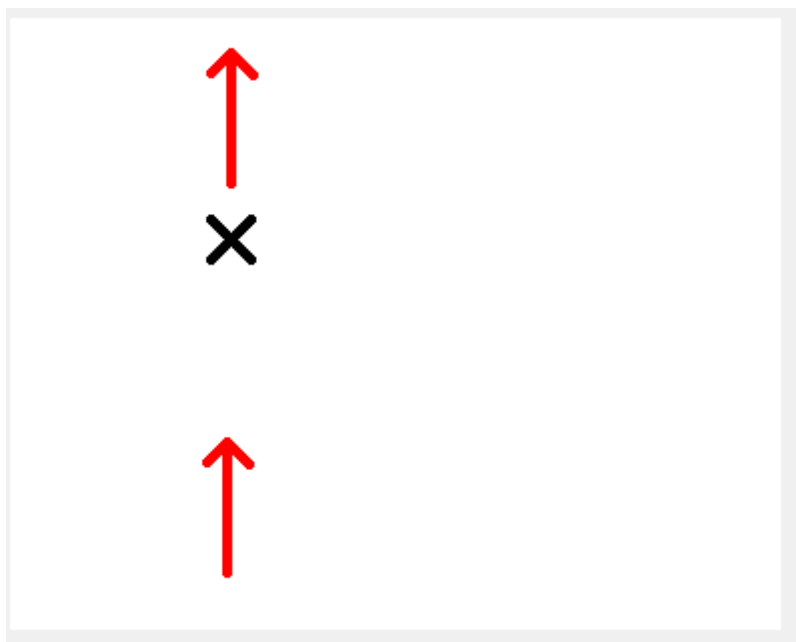
Zväčšená ukážka:

Ahoj

... test pokračuje na ďalšej strane ...

02

Vytvorte program, ktorý bude pohybovať modrý krížik veľkosti 20x20 bodov s hrúbkou čiary 5 bodov z dolnej časti obrazovky smerom k hornej časti. Keď je krížik úplne hore, znovu sa objaví v dolnej časti a ďalej sa pohybuje smerom hore. Ak hráč klikne na krížik, program mu pripočíta 1 bod. Počet bodov sa priebežne zobrazuje na obrazovke.



KONIEC TESTU

Test č. 3 B po kapitole 11	
Typ testu:	priebežný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole
Čas na vypracovanie:	45 minút
Variant:	B
Kód testu:	6203B

01 Vytvorte program, ktorý nám umožní „odtláčať“ do Image-u farebný text takto:

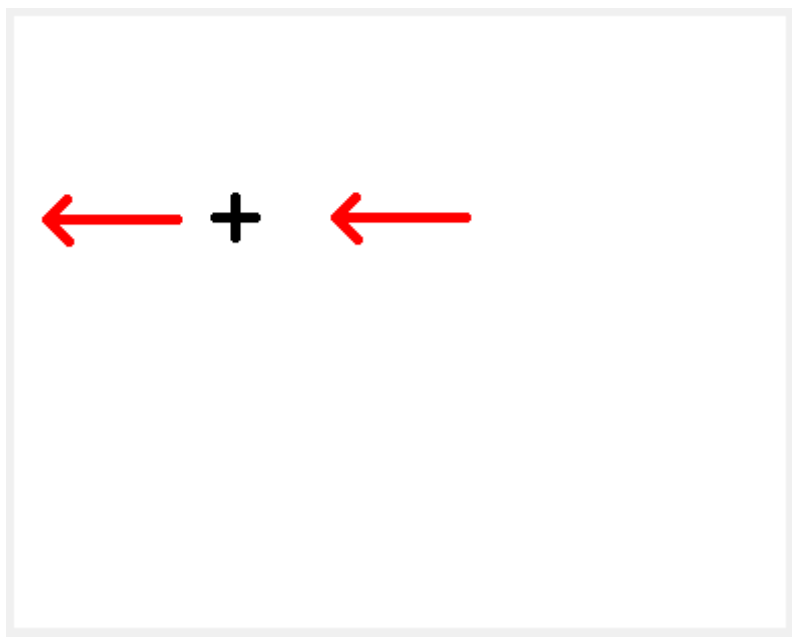


- na miesto kliknutia Imagu sa napíše text – päťkrát pod sebou, ktorý je zadany v komponente Edit1,
- každé posunutie v texte je nakreslené náhodnou farbou.

... test pokračuje na ďalšej strane ...

02

Vytvorte program, ktorý bude pohybovať červený krížik veľkosti 20x20 bodov s hrúbkou čiary 5 bodov z pravej časti obrazovky smerom k ľavej časti. Keď je krížik úplne vľavo, znovu sa objaví v pravej časti a ďalej sa pohybuje smerom vľavo. Ak hráč klikne na krížik, program mu pripočíta 1 bod. Počet bodov sa priebežne zobrazuje na obrazovke.

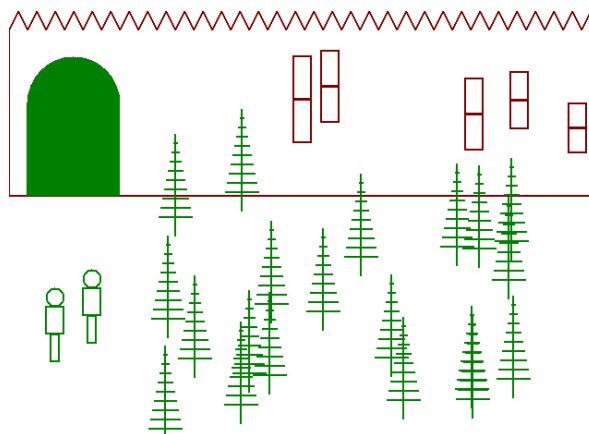


KONIEC TESTU

Polročný test č. 4 A po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	A
Kód testu:	6304A

Text k úlohám č. 1 - 6.

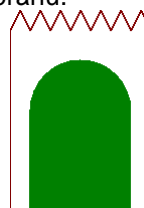
Vytvorte v Pythone program, ktorý bude simulovať preteky rytierov o dobytie hradu podľa nasledovného zadania. Snažte sa vytvoriť program čo najefektívnejšie, všade, kde je to možné a vhodné, použite for cyklus. V prípade, že neviete riešiť celú úlohu, vyriešte aspoň časť, ktorú viete. V obrázku použite farby podľa predlohy.



- 01** V hornej časti obrazovky nakreslite hradby (celá šírka obrazovky).



- 02** V ľavej časti hradieb nakreslite vstupnú bránu.



- 03** Kliknutím na hradby – okrem časti kde je brána, sa nakreslí na mieste kliknutia okno so zadanou výškou. Výšku zadávame pomocou súčiastky Entry. Rám okna je nakreslený hnedou farbou.



výška okna 30



výška okna 50

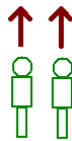


výška okna 70

-
- 04** Upravte program tak, že po stlačení tlačidla „Strom“ sa pred hradbami na náhodnom mieste nakreslí jeden zelený **strom**. Stromy sa **nekreslia** pred vstupom do hradu.



-
- 05** Upravte program tak, že bude simulovať preteky dvoch rytierov bežiacich k hradu. Rytieri bežia smerom k bráne z ľavého dolného rohu obrazovky. Na začiatku sú pri okraji obrazovky vedľa seba a potom sa začnú pohybovať smerom hore. Pri každom posune sa každý z nich posunie o náhodný krok s veľkosťou 0 až 10 bodov. Keď sa jeden z nich dostane k bráne, animácia zastane a na obrazovku sa vypíše číslo rytiera, ktorý vyhral.



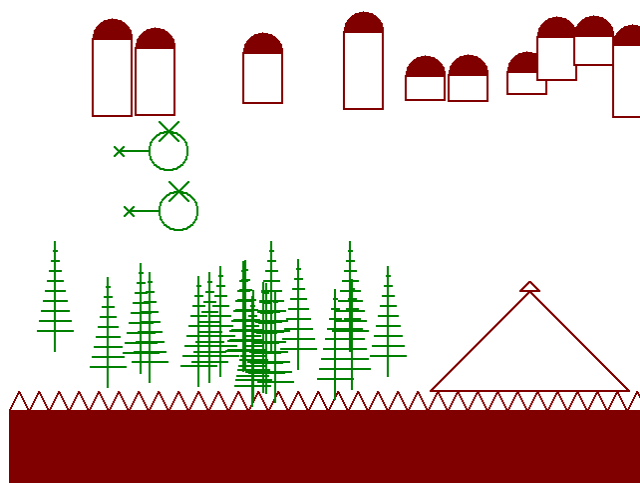
-
- 06** Upravte program tak, že nám na začiatku umožní zadať tip na víťaza (vami zvolenou formou – napríklad tlačidlami alebo pomocou Entry a pod.) a až potom odštartovať preteky. Po skončení pretekov program vyhodnotí, či sme tipovali správne.

KONIEC TESTU

Polročný test č. 4 B po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	6304B

Text k úlohám č. 1 - 6.

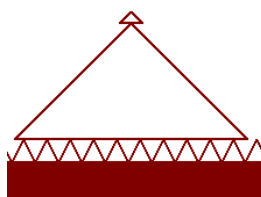
Vytvorte v Pythone program, ktorý bude simulovať preteky vrtuľníkov podľa nasledovného zadania. Snažte sa vytvoriť program čo najefektívnejšie, všade, kde je to možné a vhodné, použite for cyklus. V prípade, že neviete riešiť celú úlohu, vyriešte aspoň časť, ktorú viete. V obrázku použite farby podľa predlohy.



- 01** V hornej časti obrazovky nakreslite hradby (celá šírka obrazovky).



- 02** V pravej časti hradieb nakreslite strážnu vežu.



- 03** Kliknutím v hornej časti sa nakreslí na mieste kliknutia veža so zadanou výškou. Výšku zadávame pomocou súčiastky Entry. Všetky veže majú rovnakú šírku.



výška = 30



výška = 50

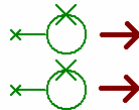


výška = 70

-
- 04** Upravte program tak, že po stlačení tlačidla „Strom“ sa nad hradbami na náhodnom mieste nakreslí jeden zelený **strom**. **Stromy sa nekreslia** cez strážnu vežu.



-
- 05** Upravte program tak, že bude simulovať preteky dvoch vrtuľníkov letiacich z ľavého okraja smerom k pravému. Vrtuľníky sú na začiatku pri okraji obrazovky vedľa seba a potom sa začnú pohybovať smerom vpravo. Pri každom posune sa každý z nich posunie o náhodný krok s veľkosťou 0 až 10 bodov. Keď sa jeden z nich dostane k pravému okraju, animácia zastane a na obrazovku sa vypíše číslo vrtuľníka, ktorý vyhral.



-
- 06** Upravte program tak, že nám na začiatku umožní zadať tip na víťaza (vami zvolenou formou – napríklad tlačidlami alebo pomocou Entry a pod.) a až potom odštartovať preteky. Po skončení pretekov program vyhodnotí, či sme tipovali správne.

KONIEC TESTU

Polročný test č. 5 A po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	A
Kód testu:	6305A

01 Vytvorte program, ktorý má nasledovné vlastnosti:

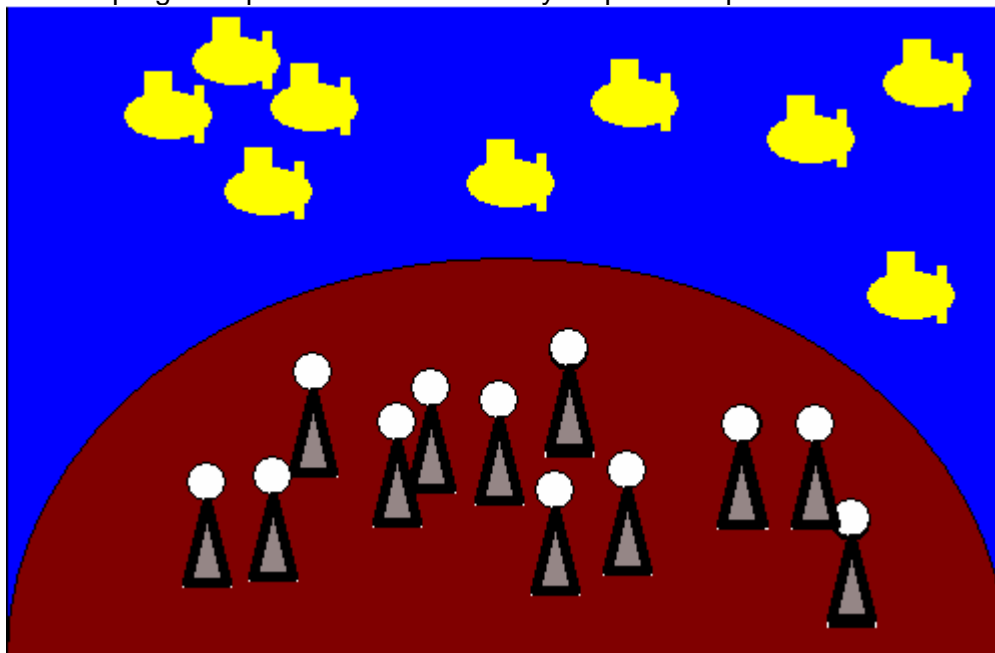
- Hneď po spustení programu sa v dolnej časti nakreslí hnedý ostrov a vo zvyšnom okolí modré more.
- Kliknutím na ostrov kreslíme „domorodkyne“ (kliknutie na ostrov je približné, v čo najvhodnejšom obdĺžniku). Domorodkyňa má bielu hlavu a telo je čierne s hrúbkou 5 bodov (pozri obrázok).



- Kliknutím na more kreslíme žltú ponorku (pozri obrázok).

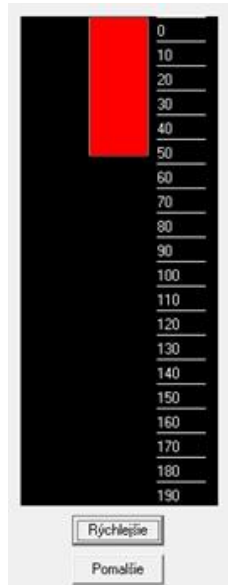


Ukážka programu po naklikaní domorodkýň a ponoriek používateľom:



02 Vytvorte program do palubnej dosky auta takto:

- Hneď po spustení programu sa ukáže čierna palubná doska, na ktorej sa zobrazuje rýchlosť auta.
- V pravej časti je nakreslená stupnica.
- V ľavej časti sa ukazuje červený stĺpec, ktorý znázorňuje aktuálnu rýchlosť.
- Tlačidlom rýchlejšie zvýšime rýchlosť o 5.
- Tlačidlom pomalšie znížime rýchlosť o 5.
- Program nám nedovolí nastaviť rýchlosť nad 130 km/hod.
- Program nám nedovolí nastaviť rýchlosť pod 0 km/hod.
- Po každej zmene rýchlosti sa zmení aj červený stĺpec.



03 Vytvorte program „preteky UFO“ takto:

- V spodnej časti obrazovky sa nakreslia vedľa seba modré a červené UFO.
- Oba lietajúce tanierové sa pohybujú smerom hore v pravidelných intervaloch. Pričom každé z nich sa posunie počas jedného presunu o náhodný krok, ktorého veľkosť je 1 až 10 bodov.
- Animácia zastane, keď jedno UFO sa dostane k hornému okraju obrazovky. Program aj vypíše, ktoré UFO vyhralo.
- Používateľ môže preteky ovplyvňovať. Keď klikne na niektoré UFO, tak ho posunie o 20 bodov dole.



KONIEC TESTU

Polročný test č. 5 B po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	B
Kód testu:	6305B

01 Vytvorte program, ktorý má nasledovné vlastnosti:

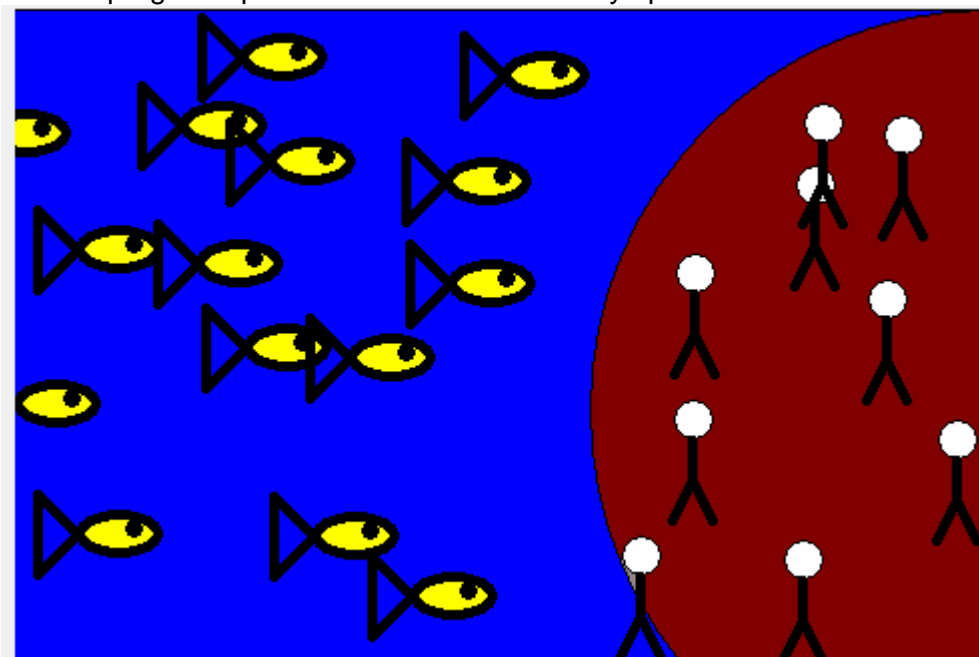
- Hneď po spustení programu sa v pravej časti nakreslí hnedý ostrov a vo zvyšnom okolí modré more.
- Kliknutím na ostrov kreslíme „domorodcov“ (kliknutie na ostrov je približné, v čo najvhodnejšom obdĺžniku). Domorodec má bielu hlavu a telo je čierne s hrúbkou 5 bodov (pozri obrázok).



- Kliknutím na more kreslíme rybu. Ryba má telo žltej farby. Obrys a oko sú čierne. Hrúbka obrysu je 5 bodov (pozri obrázok).

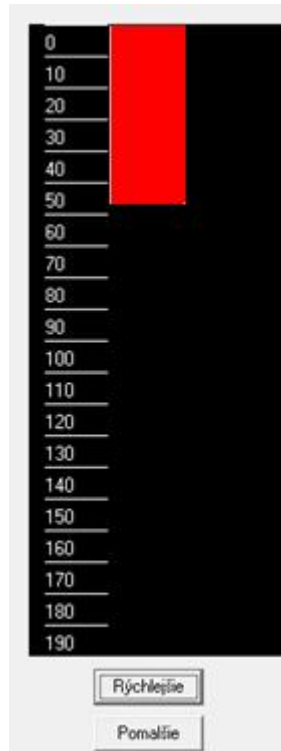


Ukážka programu po naklikaní domorodcov a rýb používateľom:



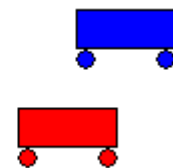
02 Vytvorte program do palubnej dosky auta takto:

- Hneď po spustení programu sa ukáže čierna palubná doska, na ktorej sa zobrazuje rýchlosť auta.
- V ľavej časti je nakreslená stupnica.
- V pravej časti sa ukazuje červený stĺpec, ktorý znázorňuje aktuálnu rýchlosť.
- Tlačidlom rýchlejšie zvýšime rýchlosť o 5.
- Tlačidlom pomalšie znížime rýchlosť o 5.
- Program nám nedovolí nastaviť rýchlosť nad 130 km/hod.
- Program nám nedovolí nastaviť rýchlosť pod 0 km/hod.
- Po každej zmene rýchlosti sa zmení aj červený stĺpec.



03 Vytvorte program „preteky vagónov“ takto:

- V pravej časti obrazovky sa nakreslia vedľa seba modrý a červený vagón.
- Oba vagóny sa pohybujú smerom vľavo v pravidelných intervaloch. Pričom každý z nich sa posunie počas jedného presunu o náhodný krok, ktorého veľkosť je 1 až 10 bodov.
- Animácia zastane, keď sa jeden z vagónov dostane k ľavému okraju obrazovky. Program aj vypíše, ktorý vagón vyhral.
- Používateľ môže preteky ovplyvňovať. Keď klikne na niektorý vagón, tak ho posunie o 20 bodov späť (vpravo).



KONIEC TESTU

Polročný test č. 5 C po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	C
Kód testu:	6305C

01 Vytvorte program, ktorý má nasledovné vlastnosti:

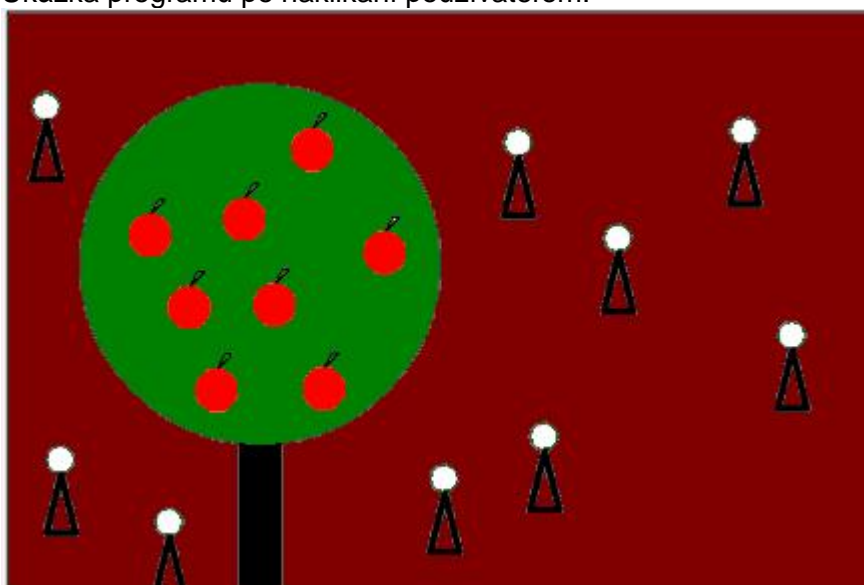
- Hneď po spustení programu sa v ľavej časti nakreslí zelený strom s čiernym kmeňom. Zvyšné pozadie bude hnedé.
- Kliknutím na korunu stromu kreslíme v mieste kliknutia jablká (kliknutie na korunu stromu je približné, v čo najvhodnejšom obdĺžniku). Jablká sú červené a majú čiernu stopku s hrúbkou 3 body (pozri obrázok).



- Kliknutím mimo stromu kreslíme v mieste kliknutia ľudí. Telo je nakreslené čiernou farbou s hrúbkou 5 bodov. Hlava je vyplnená bielou farbou a okraj má hrúbku 1 (pozri obrázok).



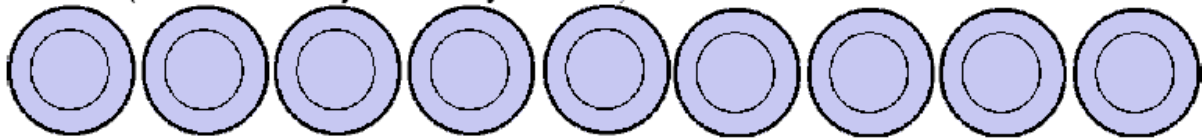
Ukážka programu po naklikaní používateľom:



- 02** V jedálni sme si položili na stôl lístok na obed. Spolužiaci si na stôl rozložili tanier a lístok zakryli. Kliknutím na tanier treba nájsť lístok. Vytvorte program, ktorý bude simulovať takúto situáciu.

Najprv na ploche nakreslite lístok (malý zelený obdĺžnik) – na náhodnom mieste, ale v rade, kde sa budú rozkladať tanier. Hneď potom sa cez lístok nakreslia bledomodré tanier (lístok môže aj trochu vyčnievať).

Lístok: 



Úlohou hráča je kliknúť na tanier (resp. okolie), za ktorým je skrytý lístok. Hráčovi program radí, či treba kliknúť vpravo alebo vľavo. Keď hráč klikne v blízkosti skrytého lístka, program mu oznámi, že vyhral.

- 03** Vytvorte program „preteky UFO“ takto:

- V spodnej časti obrazovky sa nakreslia vedľa seba modré a červené UFO.
- Oba lietajúce tanier sa pohybujú smerom hore v pravidelných intervaloch. Pričom každé z nich sa posunie počas jedného presunu o náhodný krok, ktorého veľkosť je 1 až 10 bodov.
- Animácia zastane, keď jedno UFO sa dostane k hornému okraju obrazovky. Program aj vypíše, ktoré UFO vyhralo.
- Používateľ môže preteky ovplyvňovať. Keď klikne na niektoré UFO, tak ho posunie o 20 bodov dole.



KONIEC TESTU

Polročný test č. 5 D po kapitole 14	
Typ testu:	polročný test
Typ úloh:	praktické otvorené úlohy riešené na počítači
Potrebné znalosti:	3. Grafické príkazy 4. Premenné a náhodné hodnoty 5. Opakovanie časti programu – for cyklus 6. Vytvárame podprogramy 8. Klikanie myšou a ovládanie klávesnicou 9. Podmienené príkazy 10. Časovač 11. Tlačidlá a vstupné pole 12. Posúvanie objektov canvasu 13. Vytvárame jednoduché hry {odporúčané} 14. Úlohy na opakovanie II {odporúčané}
Čas na vypracovanie:	120 minút
Variant:	D
Kód testu:	6305D

01 Vytvorte program, ktorý má nasledovné vlastnosti:

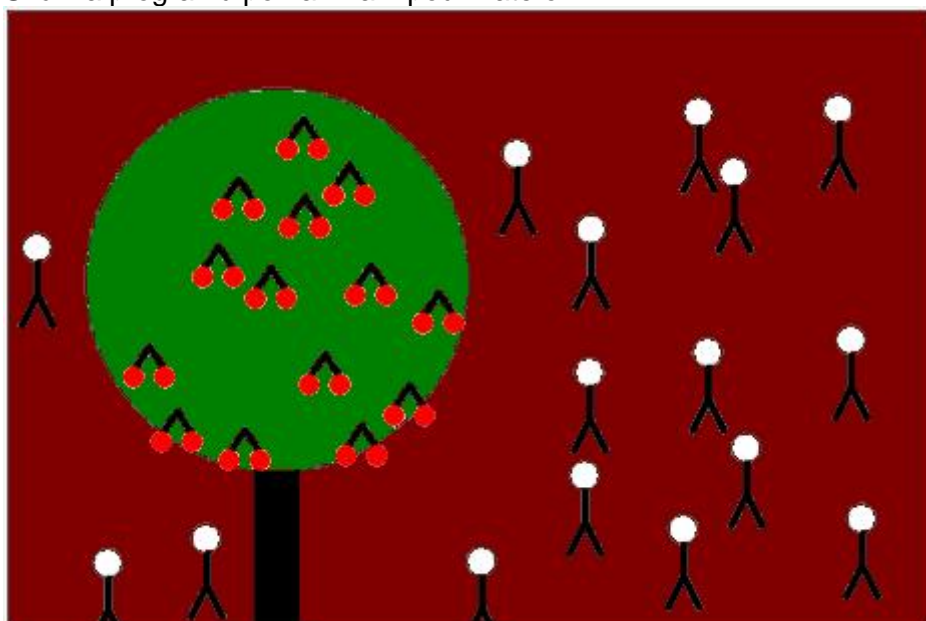
- Hneď po spustení programu sa v ľavej časti nakreslí zelený strom s čiernym kmeňom. Zvyšné pozadie bude hnedé.
- Kliknutím na korunu stromu kreslíme v mieste kliknutia čerešne (kliknutie na korunu stromu je približné, v čo najvhodnejšom obdĺžniku). Čerešne sú červené a majú čiernu stopku s hrúbkou 5 bodov (pozri obrázok).



- Kliknutím mimo stromu kreslíme v mieste kliknutia ľudí. Telo je nakreslené čiernou farbou s hrúbkou 5 bodov. Hlava je vyplnená bielou farbou a okraj má hrúbku 1 (pozri obrázok).



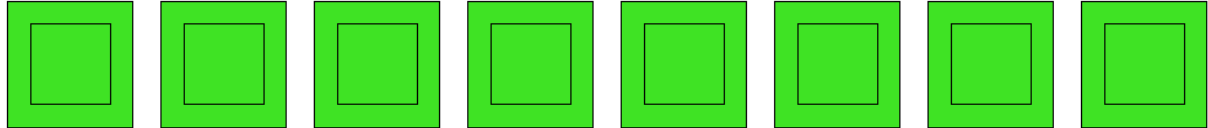
Ukážka programu po naklikaní používateľom:



- 02** V jedálni sme si položili na stôl lístok na obed. Spolužiaci si na stôl rozložili tanier a lístok zakryli. Kliknutím na tanier treba nájsť lístok. Vytvorte program, ktorý bude simulovať takúto situáciu.

Najprv na ploche nakreslite lístok (malý modrý obdĺžnik) – na náhodnom mieste, ale v rade, kde sa budú rozkladať tanier. Hneď potom sa cez lístok nakreslia zelené štvorcové tanier (lístok môže aj trochu vyčnievať).

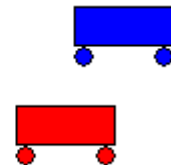
Lístok: 



Úlohou hráča je kliknúť na tanier (resp. okolie), za ktorým je skrytý lístok. Hráčovi program radí, či treba kliknúť vpravo alebo vľavo. Keď hráč klikne v blízkosti skrytého lístka, program mu oznámi, že vyhral.

-
- 03** Vytvorte program „preteky vagónov“ takto:

- V pravej časti obrazovky sa nakreslia vedľa seba modrý a červený vagón.
- Oba vagóny sa pohybujú smerom vľavo v pravidelných intervaloch. Pričom každý z nich sa posunie počas jedného presunu o náhodný krok, ktorého veľkosť je 1 až 10 bodov.
- Animácia zastane, keď sa jeden z vagónov dostane k ľavému okraju obrazovky. Program aj vypíše, ktorý vagón vyhral.
- Používateľ môže preteky ovplyvňovať. Keď klikne na niektorý vagón, tak ho posunie o 20 bodov späť (vpravo).



KONIEC TESTU

