

KARETNÍ HRA PRŠÍ

Dokumentace k zápočtovému programu

Karty jsou opředeny mnoha legendami. V některých pramenech je původ prvních karet připisován antickému filozofovi Chilonovi, který prý karetní hru považoval za prostředek zapomenutí chudiny na svou bídu. Na starých evropských kartách je často zobrazován bůh Merkur, jehož darem lidstvu prý karty jsou. Jiná legenda praví, že karetní hra vznikla z dlouhé chvíle při obléhání Tróje, motiv krásné Heleny se však na žádném karetním listu nedochoval.....

Radek Klein

2/28/2007



KARETNÍ HRA PRŠÍ

Dokumentace k zápočtovému programu

ZADÁNÍ

Naprogramujte karetní hru Prší v programovacím jazyku Pascal. Hru bude hrát jeden hráč s počítačem a bude probíhat v textovém režimu.

Pravidla hry

Každému hráči jsou vylosovány čtyři karty a jedna karta doprostřed stolu. Zbytek balíku tvoří talón, ze kterého hráči dobírají karty. Hráč, který je na tahu, smí odhodit buď kartu vyložené barvy nebo kartu stejné číselné hodnoty. Nemůže-li nebo nechce-li hráč odhodit kartu, musí vzít kartu z talónu. Filka lze odhodit na kteroukoliv barvu a hráč určí, jakou barvu představuje pro dalšího hráče. Odhodí-li hráč sedmu, následující hráč musí odhodit také sedmu. Pokud nemá, musí si z talónu vzít 2 karty násobené počtem sedmiček odhozených po sobě. Odhodí-li hráč eso, následující hráč stojí, tj. vynechá jeden tah. Vítězem je ten, kdo se jako první zbaví všech karet.



Ovládání

Hra se ovládá přes klávesnici. Na začátku hry si hráč zvolí, kolik dostane každý hráč do začátku karet do ruky. Na vstupu musí být zadáno kladné celé číslo v rozmezí 1 až 8. Program poté rozdává hráči a počítači zvolený počet karet a vypíše karty, které má hráč v ruce, kolik karet má v ruce protihráč a zároveň kartu, která je na vrchu zahraných karet. Poté se zeptá, jakou kartu chce hráč zahrát. Karty jsou ve formátu barva-hodnota např. kule-8 nebo zaludy-spodek. Zahrání karty se provádí napsáním čísla karty a stisknutím ENTERu.

V případě, že by se hráč pokusil zahrát kartu, kterou nemá nebo podle pravidel nemůžete zahrát, program ho upozorní, aby nepodváděl a znovu vyzve k zapsání karty.

PROGRAM

Následující část pojednává o samotném programu z hlediska programátorského a obsahuje vysvětlení jeho funkčnosti, popisy procedur atd. Hra byla vytvořena ve vývojovém prostředí Bloodshed Dev-Pascal 1.9.2, kde byl jako kompilátor nastaven Freepascal.

Reprezentace dat

Z celého programu pro mne bylo asi nejobtížnější částí vymyslet, jak budou jednotlivé karty, balíčky i karty v rukách obou hráčů reprezentovány. Nakonec jsem se rozhodl pro typ záznam, který jsem v pascalu nadefinoval následujícím způsobem:

```
type Tkarta = record
    barva:array[1..32] of string;
    hodnota:array[1..32] of string;
    obsahuje:array[1..32] of boolean;
end;
```

Všechny položky záznamu jsou typu pole o 32 prvcích, což odpovídá počtu karet. Pro představu se v podstatě jedná o tabulku o 32 řádkách a sloupcích *ID* (jasné identifikační číslo karty), *barva*, *hodnota* a *obsahuje*. Tato „tabulka“ se pak různými způsoby prochází. V poli *obsahuje* se nachází vždy informace, zda se karta nachází v určené proměnné.

Proměnné tohoto typu jsem nadefinoval celkem čtyři:

- Hrac – představuje karty, které má v ruce hráč
- Pc – karty, které má v ruce počítač
- Balik – talón prozatím nezahraných karet, ze kterých se líže
- Zahrane – karty, které již byly zahrány

Tyto proměnné pak byly naplněny podobným způsobem během procedury inicializace:

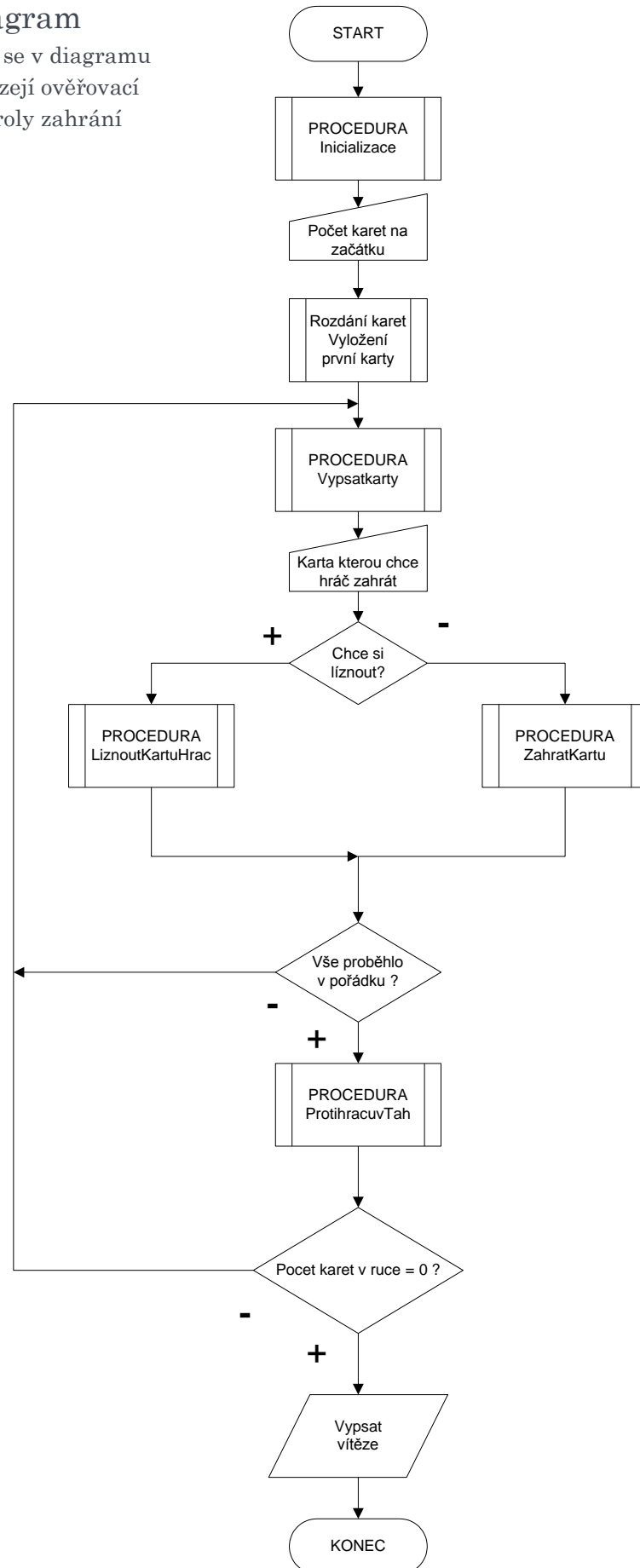
```
barva[1]:='cervena'; hodnota[1]:='7';           obsahuje[1]:=false;
barva[2]:='cervena'; hodnota[2]:='8';           obsahuje[2]:=false;
barva[3]:='cervena'; hodnota[3]:='9';           obsahuje[3]:=false;
barva[4]:='cervena'; hodnota[4]:='10';          obsahuje[4]:=false;
barva[5]:='cervena'; hodnota[5]:='spodek';      obsahuje[5]:=false;
.....
```

V programu si nikde nepamatuji, v jakém jsou karty pořadí. Místo toho se vždy ze zbývajících karet vybírá jedna náhodná, což není nejefektivnější způsob, ale vzhledem k výkonu dnešních PC a tomu, že se jedná jen o 32 karet, považuji ztráty za zanedbatelné. Ukládá se ovšem poslední zahraná karta, a to do dvou proměnných typu záznam. V proměnné *PosledniKarta* je karta, s kterou se právě manipuluje, *AktivniKarta* je už karta zahraná a nehybná.

```
PosledniKarta: record           {karta, s kterou se pracuje - prave se hraje}
    barva:string;
    hodnota:string;
end;
AktivniKarta: record           {karta, ktera je jiz zahrana na vrchu baliku}
    barva:string;
    hodnota:string;
end;
```

Vývojový diagram

Pro zjednodušení se v diagramu zejména nenacházejí ověřovací podmínky a kontroly zahrání speciálních karet



Popis hlavního programu

Na počátku zavoláme proceduru *inicializace*, která slouží zejména k načtení proměnných a zobrazení ovládnutí a pravidel. Zjistíme počet karet, které budou do začátku rozděny každému hráči, rozdáme je a smažeme obrazovku. Následuje vyložení první karty, její vypsání a zjištění, zda se nejedná o eso nebo sedmičku. Následující část se opakuje až do doby než má některý z hráčů na ruce 0 karet:

Voláním procedury *vypsankarty* se vypíší na obrazovku informace o kartách v ruce hráče, poslední zahráné kartě atd. Pokud bylo v tahu protihráče zahráno eso, nabývá proměnná *ZahranoEsoPC* hodnoty TRUE, vypíše se, že hráč stojí a na tahu je opět protihráč. Pokud eso nebylo zahráno, zeptá se program hráče, jakou chce zahrát kartu. Pokud hráč vloží nulu, žádná karta se nezahráje a místo toho si lízne počet karet určených proměnnou *PocetLiznuti*. V případě jiného čísla program zavolá proceduru *ZahratKartu* která zkontroluje, zda je možno kartu zahrát a pokud ano, tak ji přesune z hráčovy ruky na balík zahráných karet. Pokud vše proběhlo v pořádku, následuje protihráčův tah. Nakonec se ještě zjistí, zda některý z hráčů nemá na ruce žádné karty a pokud tato situace nastane, program vypíše vítěze a hra skončí, jinak se celý děj opakuje.

Popis jednotlivých procedur

Inicializace

Na počátku vymažu obrazovku a inicializuji generátor náhodných čísel, který bude zapotřebí pro pozdější lízání karet. Do proměnné *Hrac* poté dosadím jednotlivé karty. Karty jsou v pořadí, ve kterém budou i později prohledávány a v případě tahu počítače zahrány. Proto jsou na prvních čtyřech místech esa, aby se jich počítač snažil zbavit jako prvních a až na úplném konci filkové pro změnu barvy v případech, kdy není žádná jiná vhodná karta nalezena. Bližší popis je možné nalézt v části Reprezentace dat. Stejně hodnoty jsou pak vloženy do proměnných *pc*, *balik* a *zahrane* jen s tím rozdílem, že na počátku jsou všechny karty v balíku (talónu) a tedy pro všechny *i* má *Balik.Obsahuje[i]* hodnotu true. Poslední částí je vypsání pravidel v případě kladné odpovědi hráče.

ZamichatBalik

Procedura slouží k zamíchání zahráných karet do balíku v případě, že z něj byly odebrány všechny karty. Nejprve se zjistí, zda je v něm ještě nějaká karta. Pokud ano, tak se nic nestane. V opačném případě zamíchá zahráné karty do nového balíku, z kterého se opět mohou lízat karty. Zamíchání se provede tím, že se prochází cyklem mezi všemi kartami a u všech karet obsažených v zahráných kartách se nastaví proměnná *zahrane.obsahuje[k]* na hodnotu FALSE a *balik.obsahuje[k]* na hodnotu TRUE. *K* je index právě procházené karty (hodnota řídicí proměnné).

LíznoutKartu(kdo)

Procedura slouží k tomu, aby si podle parametru procedury hráč (pro `kdo='hrac'`) nebo počítač (`kdo='pc'`) líznuli kartu z balíku. Líznutí proběhne tím způsobem, že se vybere náhodná karta z balíku a ta se z balíku přesune do hráčovy/počítačovo ruky. Tato akce je realizována opakovaným generováním náhodného čísla a zjišťováním, zda se karta s tímto indexem nachází v balíku. Děj se opakuje do doby než se takováto karta najde. Jakmile se najde, tak je přesunuta do ruky hráče/počítače (např. pro hráče vypadá kód: `hrac.obsahuje[j]:=TRUE`; `balik.obsahuje[j]:=FALSE`) a přes logickou proměnnou *VPoradku* je cyklus ukončen. Protože by v případě, že by už nebyla v balíku žádná karta, probíhalo hledání karty do nekonečna, je ještě před prohledáváním zavolána procedura *ZamichatBalik*. Pokud `kdo='hrac'`, je ještě vypsána líznutá karta.

Důležitou proměnnou je *PocetLiznuti* udávající, kolik karet si má hráč líznout. Standardně je v ní hodnota 1, ale může se zvýšit po zahrání sedmiček. Pakliže je v ní hodnota větší než jedna, zmenší se její hodnota o jedničku a procedura zavolá sebe samu. To se provádí až do doby, než je její hodnota rovna jedné, tím pádem si hráč lízne odpovídající počet karet.

VypsatKarty

Procedura složí k podání informací o současném stavu hry, tedy vypsání karet na ruce hráče, počtu protihráčových karet a též informace o poslední zahrané kartě. Všechny karty jsou očíslované, líznutí další karty je označeno nulou.

ZahratKartu

Procedura, která slouží k zahrání karty hráče.

Na počátku nastavíme logické proměnné, které hlídají zahrání esa a zda vše proběhlo v pořádku na hodnotu FALSE, poté určíme, která karta odpovídá zadanému číslu. Opět procházíme všech 32 karet a zjišťujeme nejprve, zda se karta, kterou hrajeme (*PosledniKarta*), rovná kartě v hráčově balíku. Tím vyloučíme nesmyslné zadání a zároveň zjistíme, s jakým indexem karty budeme dále pracovat. Ověříme, zda hráč danou kartu vlastní a poté, zda se dá zahrát na současnou kartu.

To je možné jen ve třech případech:

- 1) Barvy zahrané (*PosledniKarta.Barva*) a aktivní karty (*AktivniKarta.Barva*) jsou totožné
- 2) Hodnoty obou karet jsou shodné
- 3) Zahraná karta je filek

Pokud je alespoň jedna z těchto podmínek splněna, cyklus pokračuje a ověří se, zda se nepokoušíme zahrát jinou kartu než sedmičku, pokud byla zahrána v minulém kole sedmička (*PocetLiznuti*>1). Pokud je vše v pořádku, nastaví se proměnná *VPoradku* do hodnoty TRUE, karta z ruky je zahrána do balíku a stane se aktivní kartou.

Dále je zapotřebí ošetřit speciální karty. Pokud byla poslední zahraná karta sedmička, zvýší se hodnota proměnné *PocetLiznuti*, v případě zahrání esa se nastaví proměnná *ZahranoEso* na TRUE a počítač bude v příštím kole stát. Pokud byl zahrán filek, vypíšu se možnosti, na jakou je možno změnit barvu, a proměnná *aktivnikarta.barva* se podle výběru hráče změní.

Pokud neproběhlo něco v pořádku a proměnná *VPoradku* zůstala na hodnotě FALSE, program napomene hráče, aby nepodváděl, neboť chtěl zahrát kartu, kterou nevládní nebo podle pravidel nemůže zahrát.

ProtihracuvTah

Nejprve se nastaví pomocné logické proměnné na hodnotu FALSE, hodnot TRUE se dosáhne až při splnění určitých podmínek.

Pakliže bylo v minulém tahu zahráno eso (*ZahranoEso*=TRUE), počítač jen oznámí, že stojí a ukončí svůj tah. Pokud byla v minulém tahu zahrána sedmička, budou se hledat karty jen mezi sedmičkami, v obráceném případě mezi všemi kartami kromě filků. Poté probíhá hledání a zahrání karty obdobně jako v proceduře *ZahratKartu*. Pakliže žádná z karet nevyhovuje, zjistí se, zda má počítač filka a pokud ano, tak ho zahraje. Barvu změni podle počtu zbývajících karet na ruce. V případě, že nemůže zahrát žádnou kartu, lízne si odpovídající počet karet z talónu.

Změny ve verzi 1.1

- Zjednodušeno zahrávání karet, místo složitého zapisování barvy a hodnoty jsou karty, které hráč drží v ruce, očíslovány a stačí jen zadat číslo a stisknout ENTER.
- Předělány opakující se části kódu, procedury *LiznoutKartuHrac* a *LiznoutKartuPC* byly spojeny v jednu
- Úprava barev v některých částech programu a pravidel/popisu ovládání
- Opravena chyba ve výběru změny barvy poté, co počítač zahraje filka.
- Přehlednější vypisování karet bez velkých mezer pokud si hráč lízne najednou více karet

ZDROJOVÝ KÓD

Jedná se o kód verze 1.0, aktuální zdrojový kód se nachází v adresáři Source

```

program Prsi;

  { Karetní hra Prsi hrana proti pocitaci }
  { Radek Klein, I. ročník, stud. skupina I31 }
  { Zimní semestr 2006/7 }
  { PRG030 Programování I }

uses crt;

type Tkarta = record           {zaznam vyuzivany pro reprezentaci karet}
  barva:array[1..32] of string;
  hodnota:array[1..32] of string;
  obsahuje:array[1..32] of boolean;
end;

var PocetKaretNaZacatku,i,j,k,PocetKaretHrac,PocetKaretPC,zmenabarvy:integer;
    PocetCervenyh,PocetZelenyh,PocetKuli,PocetZaludu,PocetLiznuti:integer;
    StartI,KonecI,ZbyvaKarta:integer; {pomocne promenne pro reseni sedmicek}
    Hrac,PC,Balik,Zahrane:Tkarta;    {jednotliva umisteni karet}
    VPoradku,Konec,ZahranoEso,ZahranoEsoPC,ZahranFilekPC:boolean;
    Co:string;
    PosledniKarta: record       {karta, s kterou se pracuje - prave se hraje}
      barva:string;
      hodnota:string;
    end;
    AktivniKarta: record       {karta, ktera je jiz zahrana na vrchu baliku}
      barva:string;
      hodnota:string;
    end;

  {***** INICIALIZACE *****}

procedure Inicializace;
begin

  ClrScr; Randomize; TextColor(LightCyan);

  {naplneni promennych}

  with hrac do
  begin
  barva[1]:='cervena';  hodnota[1]:='eso';      obsahuje[1]:=FALSE;
  barva[2]:='zelena';   hodnota[2]:='eso';      obsahuje[2]:=FALSE;
  barva[3]:='kule';     hodnota[3]:='eso';      obsahuje[3]:=FALSE;
  barva[4]:='zaludy';   hodnota[4]:='eso';      obsahuje[4]:=FALSE;

  barva[5]:='cervena';  hodnota[5]:='8';        obsahuje[5]:=FALSE;
  barva[6]:='cervena';  hodnota[6]:='9';        obsahuje[6]:=FALSE;
  barva[7]:='cervena';  hodnota[7]:='10';       obsahuje[7]:=FALSE;
  barva[8]:='cervena';  hodnota[8]:='spodek';   obsahuje[8]:=FALSE;
  barva[9]:='cervena';  hodnota[9]:='kral';     obsahuje[9]:=FALSE;
  barva[10]:='zelena';  hodnota[10]:='8';       obsahuje[10]:=FALSE;
  barva[11]:='zelena';  hodnota[11]:='9';       obsahuje[11]:=FALSE;
  barva[12]:='zelena';  hodnota[12]:='10';      obsahuje[12]:=FALSE;
  barva[13]:='zelena';  hodnota[13]:='spodek';  obsahuje[13]:=FALSE;
  barva[14]:='zelena';  hodnota[14]:='kral';    obsahuje[14]:=FALSE;
  barva[15]:='kule';    hodnota[15]:='8';       obsahuje[15]:=FALSE;
  barva[16]:='kule';    hodnota[16]:='9';       obsahuje[16]:=FALSE;
  barva[17]:='kule';    hodnota[17]:='10';     obsahuje[17]:=FALSE;
  barva[18]:='kule';    hodnota[18]:='spodek';  obsahuje[18]:=FALSE;
  
```

```

barva[19]='kule';    hodnota[19]='kral';    obsahuje[19]=FALSE;
barva[20]='zaludy'; hodnota[20]='8';      obsahuje[20]=FALSE;
barva[21]='zaludy'; hodnota[21]='9';      obsahuje[21]=FALSE;
barva[22]='zaludy'; hodnota[22]='10';     obsahuje[22]=FALSE;
barva[23]='zaludy'; hodnota[23]='spodek'; obsahuje[23]=FALSE;
barva[24]='zaludy'; hodnota[24]='kral';    obsahuje[24]=FALSE;

barva[25]='zaludy'; hodnota[25]='7';      obsahuje[25]=FALSE;
barva[26]='kule';   hodnota[26]='7';      obsahuje[26]=FALSE;
barva[27]='zelena'; hodnota[27]='7';      obsahuje[27]=FALSE;
barva[28]='cervena'; hodnota[28]='7';     obsahuje[28]=FALSE;

barva[29]='kule';   hodnota[29]='filek';  obsahuje[29]=FALSE;
barva[30]='zaludy'; hodnota[30]='filek';  obsahuje[30]=FALSE;
barva[31]='zelena'; hodnota[31]='filek';  obsahuje[31]=FALSE;
barva[32]='cervena'; hodnota[32]='filek';  obsahuje[32]=FALSE;
end;

PC:=hrac; Balik:=hrac; Zahrane:=hrac;

for i:=1 to 32 do Balik.Obsahuje[i]:=TRUE;

PocetLiznuti:=1;

{Uvodni text + vysvetleni pravidel a ovladani}

writeln('=\/= Prsi 1.0 by Radek Klein *****'); writeln;
TextColor(LightGray);
write('Chcete zobrazit pravidla a ovladani ? ano/ne: ');
readln(co);

if co='ano' then
begin
TextColor(Yellow);
writeln; writeln('Pravidla hry');
write('Kazdemu hraci jsou vylosovany ctyri karty a jedna karta doprostred ');
write('stolu. Zbytekbaliku tvorí talon, ze ktereho hraci dobírají karty. ');
write('Hrac, který je na tahu smí odhodit buď kartu vyložené barvy nebo ');
write('kartu stejné číselné hodnoty. ');
write('Nemůže-li nebo nechce-li hrac odhodit kartu, musí vzít kartu z ');
write('talonu. Filka lze odhodit na kteroukoliv barvu a hrac určí, jakou ');
write('barvu představuje pro dalšího hrace. ');
write('Odhodí-li hrac sedmu, následující hrac musí odhodit také sedmu. ');
write('Pokud nema, musisi z talonu vzít 2 karty nasobene poctem sedmicek ');
write('odhozenych po sobe. Odhodi-li hrac eso, nasledujici hrac stoji, tj. ');
write('vynecha jeden tah. Vitezem je ten, kdo se jako prvni zbavi vseh karet. ');
writeln; writeln; TextColor(lightgreen);
write('Stisknete ENTER pro pokračování...');
readln; clrscr; TextColor(yellow);
writeln('Ovladani');
write('Hra se ovlada pres klavesnici. Na pocatku hry se zvolí, kolik dostane ');
write('kazdy hrac do zacatku karet do ruky. Na vstupu musi by zadano kladne ');
write('cele cislo v rozmezi 1 az 8. Program pote rozda hraci a pocitaci ');
write('zvoleny pocet karet a vypise karty, ktere ma hrac v ruce, kolik karet ');
write('ma v ruce protihrac a zaroven kartu, ktera je na vrchu zahranych karet. ');
write('Pote se zepta, jakou kartu chce hrac zahrát. Karty jsou ve formátu ');
write('barva-hodnota napr. kule-8 nebo zaludy-spodek. Zahrani karty se provadi ');
writeln('ve dvou fazich:');
writeln;
write('1. Zvoleni barvy karty. Opiste presne nazev, který je uveden ve vypisu ');
write('u karty, kterou chcete zahrát. Vsechny nazvy jsou bez hacku a carek. ');

```

```

write(' Celkem jsou čtyři možnosti: červená, zelená, kule a žaludy. ');
write('Pokud nemůžete nebo nechcete zahrát zadnou kartu, napište "líznout" ');
write(' (bez úvozovek) a program za vás z talonu vezme kartu ');
writeln(' (v případě sedmicek více karet). Potvrďte klávesou ENTER. ');
writeln;
write('2. Zvolení hodnoty karty. Opet je zapotřebí opsat přesně výraz, jsou ');
write('povoleny následující možnosti: 7, 8, 9, 10, spodek, filek, kral, ');
write(' eso. Pokud jste už u předchozí volby vybrali možnost "líznout", ');
write('program se vás na hodnotu už nezeptá a na radu přijde ');
writeln('protihráč. Volba se opět musí potvrdit ENTERem. ');
writeln;
write('V případě, že byste se prepsali nebo se pokusili zahrát kartu, kterou ');
write(' nemáte nebo podle pravidel nemůžete zahrát, program vás upozorní, ');
writeln('aby jste nepodváděli a znovu vás vyzve k zapsání karty. ');
writeln; TextColor(lightgreen);
write('Stisknete ENTER pro pokračování...'); readln; clrscr;
TextColor(lightgray);
end;

end;

{***** ZAMICHANI BALIKU *****}

procedure ZamichatBalik;
begin
ZbyvaKarta:=0;

{nejprve zjistím zda jsou ještě nějaké karty v balíku}

for k:=1 to 32 do
begin
if Balik.Obsahuje[k] then inc(ZbyvaKarta);
end;

{pokud ne tak všechny zahrané karty přesunu do balíku}

if ZbyvaKarta=1 then
begin
writeln; textcolor(LightRed);
writeln('>>>> Dosly karty, zahrane karty byly zamichany do noveho baliku');
textcolor(LightGray);

for k:=1 to 32 do
begin
if Zahrane.Obsahuje[k] then
begin
Zahrane.Obsahuje[k]:=FALSE;
Balik.Obsahuje[k]:=TRUE;
end;
end;
end;

end;

end;

{***** LIZNUTI KARTY HRAC *****}

procedure LiznoutKartuHrac;
begin
VPoradku:=FALSE; ZahranoEso:=FALSE;

```

```

ZamichatBalik;

{nahodne si liznu kartu z baliku}

repeat
  j:=random(32);
  if Balik.Obsahuje[j]
    then begin
      Hrac.Obsahuje[j]:=TRUE;
      Balik.Obsahuje[j]:=FALSE;
      VPoradku:=TRUE;
      inc(PocetKaretHrac)
    end
until VPoradku;

writeln; textcolor(LightCyan);
writeln('Liznul jste si kartu ', Hrac.Barva[j], '-', Hrac.Hodnota[j]);
textcolor(LightGray);

{v pripade sedmicek se bude lizat tak dlouho nez se pocet liznuti dostane na
standarni jednicku ktera znamena ze nebyla zahrana sedmicka}

if PocetLiznuti>1 then
  begin
    dec(PocetLiznuti);
    LiznoutKartuHrac;
  end

end;

{***** LIZNUTI KARTY PC *****}

procedure LiznoutKartuPC;
begin
  VPoradku:=FALSE; ZahranoEsoPC:=FALSE;
  ZamichatBalik;

  {nahodne si PC lizne kartu z baliku}

  repeat
    j:=random(32);
    if Balik.Obsahuje[j]
      then begin
        PC.Obsahuje[j]:=TRUE;
        Balik.Obsahuje[j]:=FALSE;
        VPoradku:=TRUE
      end;
  until VPoradku;

  {v pripade sedmicek se bude lizat tak dlouho nez se pocet liznuti dostane na
standarni jednicku ktera znamena ze nebyla zahrana sedmicka}

  if PocetLiznuti>1 then
    begin
      dec(PocetLiznuti);
      LiznoutKartuPC;
    end

end;

```

```

{***** VYPSANI KARET *****}

procedure vypsatkarty;
begin

PocetKaretPC:=0;

{vypisu kolik mam v ruce karet}

if PocetKaretHrac = 1 then writeln('V ruce mate 1 kartu:');
if PocetKaretHrac = 2 then writeln('V ruce mate 2 karty:');
if PocetKaretHrac = 3 then writeln('V ruce mate 3 karty:');
if PocetKaretHrac = 4 then writeln('V ruce mate 4 karty:');
if PocetKaretHrac > 4 then writeln('V ruce mate ', PocetKaretHrac , ' karet:');
writeln;

{pote vypisu konkretni karty v ruce}

for i:=1 to 32 do
begin
if Hrac.Obsahuje[i] then writeln(Hrac.Barva[i], '-', Hrac.Hodnota[i]);
if PC.Obsahuje[i] then inc(PocetKaretPC)
end;

writeln;

{dale pocet karet v pocitacovo ruce}

if PocetKaretPC = 1 then writeln('Protihrac ma v ruce 1 kartu. ');
if PocetKaretPC = 2 then writeln('Protihrac ma v ruce 2 karty. ');
if PocetKaretPC = 3 then writeln('Protihrac ma v ruce 3 karty. ');
if PocetKaretPC = 4 then writeln('Protihrac ma v ruce 4 karty. ');
if PocetKaretPC > 4 then writeln('Protihrac ma v ruce ',PocetKaretPC, ' karet. ');

{a nakonec posledni zahrnou kartu}

write('Posledni zahrana karta je: ');
textcolor(White);
writeln(AktivniKarta.Barva, '-', AktivniKarta.Hodnota);
textcolor(LightGray);

end;

{***** ZAHRANI KARTY HRACE *****}

procedure ZahratiKartu;
begin
VPoradku:=FALSE; ZahranoEsc:=FALSE;

for i:=1 to 32 do
begin
if ((PosledniKarta.Barva=Hrac.Barva[i]) and
(PosledniKarta.Hodnota=Hrac.Hodnota[i])) then if Hrac.Obsahuje[i] then
if ((AktivniKarta.Barva=PosledniKarta.Barva) or
(AktivniKarta.Hodnota=PosledniKarta.Hodnota) or
(PosledniKarta.Hodnota='filek')) then
if (((PocetLiznuti>1) and (PosledniKarta.Hodnota='7')) or (PocetLiznuti=1))
then
begin

```

```

Zahrane.Obsahuje[i]:=TRUE;
Hrac.Obsahuje[i]:=FALSE;
dec(PocetKaretHrac);
VPoradku:=TRUE;
AktivniKarta.Barva:=PosledniKarta.barva;
AktivniKarta.Hodnota:=PosledniKarta.hodnota;

if PosledniKarta.Hodnota='7' then
begin
  if PocetLiznuti=1 then PocetLiznuti:=2
                    else PocetLiznuti:=PocetLiznuti+2;
end;

if PosledniKarta.Hodnota='eso' then ZahranoEso:=TRUE;
if PosledniKarta.Hodnota='filek' then
begin
  textcolor(lightgreen);
  writeln; writeln('FILEK: Na jakou barvu chcete zmenit?'); writeln;
  writeln('1: Cervena');
  writeln('2: Zelena');
  writeln('3: Kule');
  writeln('4: Zaludy');
  writeln; write('Vase volba (cislo): ');
  readln(ZmenaBarvy);
  textcolor(LightGray);

  case ZmenaBarvy of
  1: AktivniKarta.Barva:='cervena';
  2: AktivniKarta.Barva:='zelena';
  3: AktivniKarta.Barva:='kule';
  4: AktivniKarta.Barva:='zaludy';
  else AktivniKarta.Barva:=PosledniKarta.Barva;
  end;

  end;
end;
end;

if not VPoradku then
begin
  writeln; TextColor(LightRed);
  writeln('Nepodvadej !!!!! ;-)');

  if ((PocetLiznuti > 1) and (PocetLiznuti < 6)) then
  begin
    write('Na sedmicku muzete zahrat jen sedmicku nebo si liznout ');
    writeln(PocetLiznuti, ' karty');
  end;

  if PocetLiznuti > 5 then
  begin
    write('Na sedmicku muzete zahrat jen sedmicku nebo si liznout ');
    writeln(PocetLiznuti, ' karet');
  end;

  TextColor(LightGray)

end;
end;

{***** TAH POCITACE *****)

```

```

procedure ProtihracuvTah;
begin
  textcolor(LightMagenta);
  VPoradku:=FALSE; ZahranoEsoPC:=FALSE; ZahranFilekPC:=FALSE;

  {v pripade ze je zahrana sedmicka (PocetLiznuti>1) tak prochazime jen sedmicky}

  if PocetLiznuti=1 then begin StartI:=1; KonecI:=28; end
    else begin StartI:=25; KonecI:=28; end;

  {nasledujici kod se provede jen v pripade ze hrac nezahral v min. kole eso}

  if not ZahranoEso then
  begin
  for i:=StartI to KonecI do
  begin
    if ((AktivniKarta.Barva=PC.Barva[i])or(AktivniKarta.Hodnota=PC.Hodnota[i]))
      and PC.Obsahuje[i] then if not VPoradku then
      begin
        Zahrane.Obsahuje[i]:=TRUE;
        PC.Obsahuje[i]:=FALSE;
        dec(PocetKaretPC);
        VPoradku:=TRUE;
        PosledniKarta.Barva:=PC.Barva[i];
        PosledniKarta.Hodnota:=PC.Hodnota[i];
        AktivniKarta.Barva:=PosledniKarta.Barva;
        AktivniKarta.Hodnota:=PosledniKarta.Hodnota;
        writeln;
        writeln('Protihrac zahral ',PosledniKarta.Barva,'-',PosledniKarta.Hodnota);
        if poslednikarta.hodnota='7' then
        begin
          if PocetLiznuti=1 then PocetLiznuti:=2
            else PocetLiznuti:=PocetLiznuti+2;

          end;
          if PosledniKarta.Hodnota='eso' then ZahranoEsoPC:=TRUE;
          if PosledniKarta.Hodnota='filek' then ZahranFilekPC:=TRUE;;
        end;
      end;
  end;

  {paklize zadna z karet nevyhovuje, zjistim zda ma pocitac filka a pokud ano tak ho zahraje}

  if not VPoradku then
  begin
    if (PC.Obsahuje[29] and (PocetLiznuti=1)) then
    begin
      Zahrane.Obsahuje[29]:=TRUE;
      PC.Obsahuje[29]:=FALSE;
      dec(PocetKaretPC);
      VPoradku:=TRUE; ZahranFilekPC:=TRUE;
      PosledniKarta.Barva:=PC.Barva[29];
      PosledniKarta.Hodnota:=PC.Hodnota[29];
      AktivniKarta.Barva:=PosledniKarta.Barva;
      AktivniKarta.Hodnota:=PosledniKarta.Hodnota;
      writeln;
      write('Protihrac zahral kuloveho filka');
    end

    else if (PC.Obsahuje[30] and (PocetLiznuti=1)) then
    begin

```

```

Zahrane.Obsahuje[30]:=TRUE;
PC.Obsahuje[30]:=FALSE;
dec(PocetKaretPC);
VPoradku:=TRUE; ZahranFilekPC:=TRUE;
PosledniKarta.Barva:=PC.Barva[30];
PosledniKarta.Hodnota:=PC.Hodnota[30];
AktivniKarta.Barva:=PosledniKarta.Barva;
AktivniKarta.Hodnota:=PosledniKarta.Hodnota;
writeln;
write('Protihrac zahral zaludoveho filka');
end

else if (PC.Obsahuje[31] and (PocetLiznuti=1)) then
begin
  Zahrane.Obsahuje[31]:=TRUE;
  PC.Obsahuje[31]:=FALSE;
  dec(PocetKaretPC);
  VPoradku:=TRUE; ZahranFilekPC:=TRUE;
  PosledniKarta.Barva:=PC.Barva[31];
  PosledniKarta.Hodnota:=PC.Hodnota[31];
  AktivniKarta.Barva:=PosledniKarta.Barva;
  AktivniKarta.Hodnota:=PosledniKarta.Hodnota;
  writeln;
  write('Protihrac zahral zeleneho filka');
end

else if (PC.Obsahuje[32] and (PocetLiznuti=1)) then
begin
  Zahrane.Obsahuje[32]:=TRUE;
  PC.Obsahuje[32]:=FALSE;
  Dec(PocetKaretPC);
  VPoradku:=TRUE; ZahranFilekPC:=TRUE;
  PosledniKarta.Barva:=PC.Barva[32];
  PosledniKarta.Hodnota:=PC.Hodnota[32];
  AktivniKarta.Barva:=PosledniKarta.Barva;
  AktivniKarta.Hodnota:=PosledniKarta.Hodnota;
  writeln;
  write('Protihrac zahral cerveneho filka');
end

{paklize nema zadnou kartu kterou by mohl zahrat musi si liznout kartu}

else
begin
  writeln;
  case PocetLiznuti of
    1: writeln('Pocitac si liznul kartu');
    2: writeln('Pocitac si liznul 2 karty');
    4: writeln('Pocitac si liznul 4 karty');
    6: writeln('Pocitac si liznul 6 karet');
    8: writeln('Pocitac si liznul 8 karet');
    else writeln('Pocitac si liznul kartu');
  end;
  LiznoutKartuPC;
end;
end;
end

else
begin
  writeln; textcolor(Yellow);

```

```

        writeln('Stojim.....');
        textcolor(LightGray);
    end;

    {paklize byl zahran filek spocita kolik ma jednotlivych barev karet a vybere
    tu kde jich je nejvic}

    if ZahranFilekPC then
    begin
        PocetCervenyh:=0; PocetZelenyh:=0; PocetKuli:=0; PocetZaludu:=0;
        for i:=1 to 28 do
        begin
            if PC.Obsahuje[i] then
            begin
                if PC.Barva[i]='cervena' then inc(PocetCervenyh);
                if PC.Barva[i]='zelena' then inc(PocetZelenyh);
                if PC.Barva[i]='kule' then inc(PocetKuli);
                if PC.Barva[i]='zaludy' then inc(PocetZaludu);
            end;
        end;
        if PocetZaludu > 0 then AktivniKarta.Barva:='zaludy';
        if PocetKuli > PocetZaludu then AktivniKarta.Barva:='kule';
        if PocetZelenyh > PocetKuli then AktivniKarta.Barva:='zelena';
        if PocetCervenyh > PocetCervenyh then AktivniKarta.Barva:='cervena';
        write(' a zmenil barvu na ', AktivniKarta.Barva);
    end;
    textcolor(lightgray)

end;

{***** HLAVNI PROGRAM *****)

begin

inicializace;

{Rozdani karet: musi byt zadano cislo 1-8}

repeat
write('Zadejte pocet karet na zacatku v ruce: ');
readln(PocetKaretNaZacatku); writeln;
until ((PocetKaretNaZacatku > 0) and (PocetKaretNaZacatku < 9));

for i:=1 to PocetKaretNaZacatku do
begin
    LiznoutKartuHrac; LiznoutKartuPC;
end;

{vylozeni prvni karty}
clrscr;
vporadku:=FALSE;

repeat
j:=random(32);
if balik.obsahuje[j]
then begin
    zahrane.obsahuje[j]:=TRUE;
    balik.obsahuje[j]:=FALSE;
    vporadku:=TRUE;
    aktivnikarta.barva:=zahrane.barva[j];
end;
end;
end;

```

```

        aktivnikarta.hodnota:=zahrane.hodnota[j];
        end;
until vporadku;

writeln('Prvni vylozena karta je ',aktivnikarta.barva,'-',aktivnikarta.hodnota);

{osetreni specialnich karet}

if aktivnikarta.hodnota='eso' then ZahranoEsoPC:=TRUE;
if aktivnikarta.hodnota='7' then PocetLiznuti:=2;

{hlavni opakujici se cast programu}

repeat

writeln; writeln('-----'); writeln;
vypsatkarty;
writeln;

if not ZahranoEsoPC then
begin
writeln('Jakou kartu chcete zahrat?');
writeln;
write('Co: '); readln(co);
if co = 'liznout'
then begin liznoutkartuhrac; poslednikarta.hodnota:='nic'; end;
if co <> 'liznout'
then begin
poslednikarta.barva:=co;
write('Hodnota: '); readln(poslednikarta.hodnota);
zahratkartu
end;
end;

if ZahranoEsoPC then
begin
textcolor(yellow); writeln('Stojis...');
textcolor(lightgray); vporadku:=TRUE;
end;

if vporadku then ProtihracuvTah;
{zjistim zda maji hrac i pocitac jeste nejake karty a pokud ano dej se opakuje}

if ((PocetKaretHrac=0) or (PocetKaretPC=0)) then konec:=TRUE;

until konec;

{Vypisu na obrazovku kdo vyhral}

writeln;

if PocetKaretHrac=0
then begin
textcolor(lightcyan);
writeln('PRSI !'); writeln;
write('Vyhral jste ! :-)'); end
else begin
textcolor(lightred);
writeln('PRSI !'); writeln;
write('Prohral jste ! :-P');
end;

readln;

end. { Uff konecne konec =) }

```