

Dokumentace k zápočtovému programu

# curver 1.0

Programování I (NPRGo30)

1. ročník Bc. studia MFF UK - Informatika



Autor: Jakub Kinšt

## Obsah

|       |  |   |
|-------|--|---|
| 1     | Úvod.....                                    | 3 |
| 2     | Základní popis programu .....                | 3 |
| 3     | Možnosti a funkce .....                      | 3 |
| 3.1   | Curver.....                                  | 3 |
| 3.2   | Editor bludišť.....                          | 4 |
| 4     | Ovládání programu .....                      | 4 |
| 4.1   | Curver.....                                  | 4 |
| 4.2   | Editor bludišť.....                          | 4 |
| 5     | Použité datové struktury .....               | 5 |
| 5.1   | Hráč.....                                    | 5 |
| 5.2   | Bludiště.....                                | 5 |
| 6     | Jak program funguje .....                    | 6 |
| 6.1   | Curver.....                                  | 6 |
| 6.2   | Editor bludišť.....                          | 6 |
| 7     | Vysvětlení klíčových funkcí a procedur ..... | 7 |
| 7.1   | Curver.....                                  | 7 |
| 7.1.1 | Procedura Clear .....                        | 7 |
| 7.1.2 | Procedura Draw.....                          | 7 |
| 7.1.3 | Funkce Kolize.....                           | 7 |
| 7.2   | Editor bludišť.....                          | 7 |
| 7.2.1 | Funkce NaUsecce.....                         | 7 |
| 8     | Možná rozšíření hry .....                    | 7 |

## 1 Úvod

Tuto dokumentaci doplňují zdrojové kódy jednotlivých aplikací (*curver.pas*, *options.pas*, *editor/editor.pas*) a komentáře, které tyto zdrojové kódy obsahují. Podrobnou strukturu funkcí a procedur lze nejlépe pochopit právě v těchto zdrojových kódech.

## 2 Základní popis programu

Zápočtový program curver se skládá ze dvou částí - samotná hra curver, kterou doplňuje praktický editor bludišť určený výhradně pro hru curver. Jedná se o hru, která je vytvořená na motiv jedné velmi staré klasické hry se jménem *Achtung, Die Kurve* <sup>1</sup>. Základní princip je naprosto totožný, pouze s několika úpravami. Hra je koncipována pro 2-6 hráčů, přičemž každý z nich dvěma klávesami ovládá svého "hada" a snaží se tak ve hře vydržet co nejdéle. Had po sobě zanechává nevymazatelnou stopu, která se pak pro všechny stává překážkou. Had v sobě v pravidelných intervalech tvoří mezery, které lze v krajní situaci použít pro únik ze zdánlivě bezvýhodné situace. Hra pro hráče končí, pokud nastane kolize jeho "hada" s prvky bludiště nebo s "hady" ostatních hráčů (včetně toho svého).

Curver i editor bludišť je vytvořen pomocí vývojového nástroje Delphi (*Embarcadero® Delphi® 2010*). Kód je napsaný v jazyce Pascal.

## 3 Možnosti a funkce

### 3.1 Curver

Curver je vylepšená verze populární a klasické hry *Achtung, Die Kurve*!. Vylepšení spočívají v několika klíčových vlastnostech.

Každý ze šesti hráčů si může zvolit libovolnou barvu svého „hada“. Může si také zvolit, jakou kombinací dvou kláves bude svého „hada“ ovládat.

Hráč může velmi jednoduše vytvořit pomocí editoru vlastní bludiště, které lze snadno importovat do hry a hrát tak s libovolně nadefinovanou hrací plochou a zdmi. Tím se několikanásobně prodlouží hrací doba.

Dalším vylepšením je ta vlastnost, že pokud se na kraji hrací plochy nenalézá zeď, had může libovolně přejíždět z jednoho konce hrací plochy na druhý. To dodává hře naprosto nový rozměr hrátelnosti.

Herní zážitek je umocněn možností sbírat během hry několik typů bonusů. Některé jsou pro hráče přínosné, jiné mu naopak mohou uškodit. Bonusy se náhodně objevují během hry na hrací ploše a po určité době opět zmizí. Ve hře jsou implementovány 4 druhy bonusů. Prvním je dvojnásobné zrychlení hada a tím i zvětšení poloměru zatáčení (zároveň lze projíždět zdmi). Dále může hráč narazit na nepřítis pomocný bonus – znehybnění. Během krátkého časového intervalu nelze zatáčet. Třetím bonusem je částečná neviditelnost. Hráč vidí jen hlavu svého „hada“ a tělo jen částečně. Proto je vyhýbání-se tomuto „hadovi“ velice obtížné pro ostatní hráče.

---

<sup>1</sup> [http://cs.wikipedia.org/wiki/Achtung,\\_die\\_Kurve!](http://cs.wikipedia.org/wiki/Achtung,_die_Kurve!)

Posledním typem bonusu je možnost pro zvýhodněného hráče projíždět zdi, ostatní „hady“ i sám sebe. Všechny bonusy trvají krátký časový úsek.

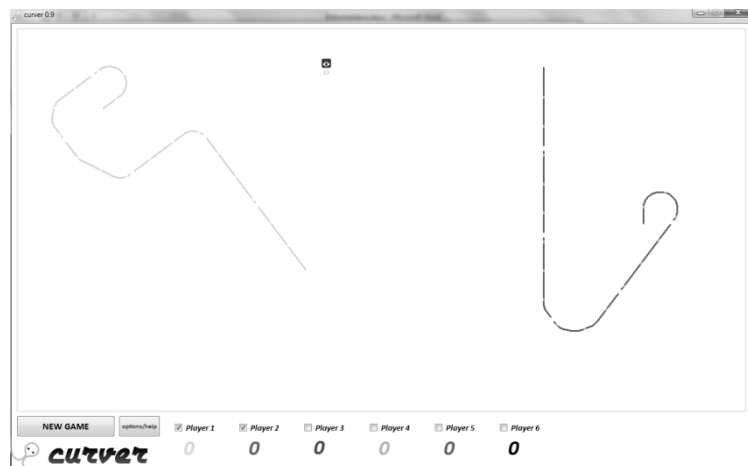
### 3.2 Editor bludišť

Editor nabízí velice jednoduché rozhraní pro tvorbu bludišť pro hru curver. Pomocí úseček a obdélníků lze vytvořit v podstatě jakoukoliv hrací plochu. Stačí ohraničit hrací plochu čarami a hráči již nemohou přejít z jednoho kraje plochy na druhý. Lze vytvářet různé objekty také uprostřed hrací plochy. Úsečky lze také velmi jednoduše smazat a nahradit jinými. Lze upravovat i již vytvořené bludiště. Aby nedocházelo k problémům, je nutné, aby autor bludiště také určil šest výchozích startovních pozic na hrací ploše.

## 4 Ovládání programu

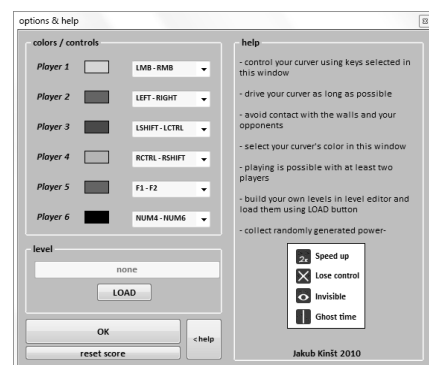
### 4.1 Curver

Hlavní okno aplikace curver se sestává pouze z několika málo ovládacích prvků. Nejdůležitější je tlačítko pro započítí nové hry (NEW GAME), jehož funkci může nahradit klávesa SPACE. Dále okno obsahuje šest zaškrtačkových políček, jimiž se určuje, kolik hráčů bude aktuálně hru hrát. Poslední ovládací prvek je tlačítko pro přístoupení k nastavení a nápovědě (options/help).



Při samotné hře je způsob ovládání plně nastavitelný. Každý hráč si v nabídce options/help zvolí dvojici kláves, které budou následně používat k řízení svého „hada“.

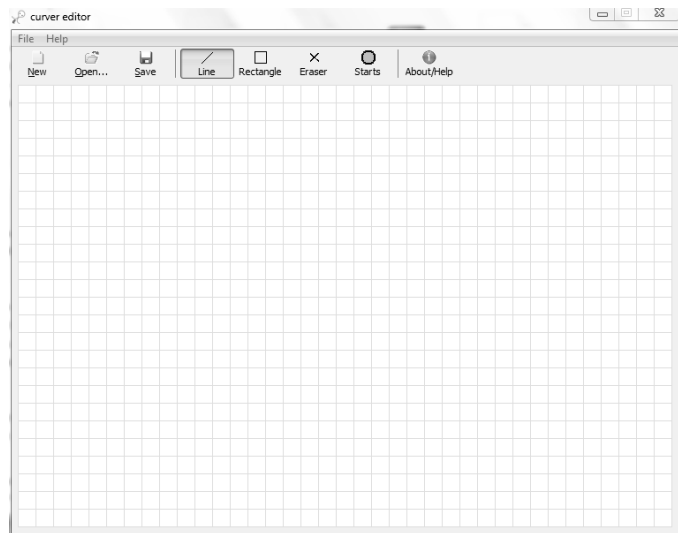
V nabídce options/help lze dále zvolit libovolnou barvu pro svého „hada“, lze vynulovat herní skóre a konečně se zde nachází i možnost importovat bludiště pomocí tlačítka LOAD.



### 4.2 Editor bludišť

Editor bludišť obsahuje několik klíčových ovládacích prvků. V první řadě je to hlavní menu, kterému odpovídá nástrojová lišta. Na té můžeme najít sadu tlačítek pro práci se soubory (New, Open, Save). Pomocí nich můžeme zakládat nová bludiště, můžeme již vytvořené otevřít nebo naopak nové uložit. Dále následují nástroje pro samotnou tvorbu bludišť. Pomocí nich lze vytvořit čáru(Line), obdélník(Rectangle), můžeme čáry mazat(Eraser) a můžeme také přesouvat

výchozí startovní pozice hráčů(Starts). S použitím daného nástroje pak už jen klikáme a kreslíme na obrázek reprezentující bludiště. Autor bludiště si vůbec nemusí lámat hlavu s tím, jak se bludiště zobrazí na různých rozlišeních obrazovky. Bludiště bude vždy poměrově přepočítáno, aby vypadalo stejně jako v editoru.



## 5 Použité datové struktury

### 5.1 Hráč

Každý z šesti hráčů je uložen v jednorozměrném poli typů P. Jedná se o záznam mnoha klíčových informací o hráči jako je jeho barva, namapování kláves, pole bodů tvořících jeho hada atd.

```
// Kazdy hrac je typu P
P = Record
cesta: Array of souradnice; // pole souradnic, mezi kterymi se vytvari cary
uhel: smallint; // aktualni uhel hrace urcuje jeho smer
barva: TColor; // barva hrace kterou si zvolil nebo defaultni
SkutecnaBarva: TColor; // skutecna barva kterou je hrac vykreslovan
skore: smallint; // aktualni skore hrace
hraje: boolean; // hraje hrac v aktualni hre ?
zivy: boolean; // je hrac zivy ?
delka: integer; // delka hada
vlevo: word; // klavesa vlevo
vpravo: word; // klavesa vpravo
rychlost: shortint; // aktualni rychlost
bon: shortint; // aktualni bonus (0..zadny)
bonTime: shortint; // zbyvajici cas bonusu
```

Pro uchovávání samotné cesty určující „hada“ konkrétního hráče jsem zvolil pole, které se dynamicky zvětšuje podle potřeby. Při spuštění nové hry je každému hráči přiděleno pole jednoho tisíce záznamů obsahujících souřadnici x a souřadnici y jednoho bodu. Alternativou bylo použití lineárního spojového seznamu, ale to se ukázalo jako řešení nevhodné. Při neustálém zvětšování cesty hráče bylo již pro procesor obtížné stále procházet tuto dynamickou strukturu. Proto se pole ukázalo jako lepší řešení.

Barva se někdy může lišit od skutečné barvy. Barva je statická barva hráče během hry, zatímco Skutečná barva je barva, kterou je aktuálně „had“ vykreslován. Toho je využíváno pro efekty pohybu atd.

Vlevo, vpravo jsou klávesy aktuálně zvolené pro ovládání „hada“. Jedná se o typ *word*, protože každá klávesa na klávesnici má jedinečný identifikátor.

### 5.2 Bludiště

Bludiště je uloženo pomocí souřadnic bodů v poli prvků typu *smallint*. Jednotlivé úsečky jsou v poli odděleny předdefinovaným oddělovačem *-1*. Toto pole lze z editoru bludišť exportovat do souboru s příponou *.crl*. Tento soubor se skládá ze dvou řádků. Na prvním z nich

se nachází zmiňované pole úseček tvořících bludiště. Na druhém je uloženo 12 souřadnic, na kterých budou startovat hráči. Tento soubor lze pak importovat znovu v editoru, nebo už přímo ve hře curver, kde je opět ve formě pole. Velikost bludiště je ve hře upravena podle aktuálního rozlišení obrazovky.

## 6 Jak program funguje

### 6.1 Curver

Jádrum hry je komponenta Timer, která se stará a plynulý běh hry. Timer po určených časových úsecích vždy vykoná proceduru OnTimer. Timer běží pouze během hraní. Pokaždé vykonává několik klíčových operací. Nejprve je nutné odchyťávat stisknutí kontrolních kláves. O to se stará cyklus, který vždy zkontroluje, jestli není stisknuta klávesa daného hráče. Pokud je, změní se hodnota úhlu hráče, kterým jeho had směřuje. Dále je potřeba zapsat do pole cesty každému živému hráči novou souřadnici, kam se pohnul. To je realizováno pomocí jednoduchého goniometrického vzorce. Podle aktuálního úhlu je vypočítána směrnice a v určité vzdálenosti je pak vytvořen nový bod hada. Úhel je uchováván ve stupních. Proto se při výpočtu dělí 180(velikostí PI ve stupních).

```
krok := Player[i].rychlost;
noveX := round(Player[i].cesta[Player[i].delka].x + krok * cos
  (PI * Player[i].uhel / 180));
noveY := round(Player[i].cesta[Player[i].delka].y + krok * -sin
  (PI * Player[i].uhel / 180));
```

V další fázi se jednou za náhodnou periodu na náhodnou pozici umísťuje náhodný typ bonusu a začíná se odpočítávat doba zobrazení na jednom místě. Poté bonus zmizí a může se (ale i nemusí) zobrazit bonus další. Potřebujeme také zkontrolovat, jestli nějaký hráč právě nenarazil na ikonu bonusu, v tom případě zjistit jaký to byl bonus a pomocí proměnné *Player.bon* mu nastavit právě ten bonus.

Ještě se Timer musí postarat o kontrolu kolize s jinými předměty na hrací ploše. To je realizováno pomocí funkce, která zjišťuje, zda je kontrolovaný bod na úsečce mezi dvěma jinými. V situaci, kdy hráč opustí hřiště, je jeho pozice nastavená přesně na druhý konec hřiště. (Pokud není v bludišti definován okraj hrací plochy)

Nakonec Timer smaže celou hrací plochu, která je realizována Obrázkem s vlastností Canvas. Díky tomu lze pomocí procedur do obrázku zakreslovat úsečky. Pokud je zvoleno bludiště je nutné ho znovu vykreslit a nakonec je pomocí úseček dané barvy pospojováno celé pole cesty každého hráče.

Alternativou mohla být strategie, při které by se neustále nesmazávalo hrací pole a pouze by se přikreslovalo. V tu chvíli by ani nebylo potřeba uchovávat cestu hráče, protože by se kolize mohly řešit pouze pomocí barev pixelů na hrací ploše. Tato možnost se mi ale zdála nepraktická, zejména kvůli ztrátě dynamičnosti. Přišli bychom tím o mnoho efektů, které takto můžeme využít. Jedná se zejména o efekty využití v různých bonusech jako je zmizení, možnost projíždět zdi atd.

### 6.2 Editor bludišť

Editor funguje principiálně podobně jako curver. Neustále běží Timer, který kontroluje jaký je zrovna zvolený nástroj a podle toho se pak chová při klikání nebo pohybu myši. Vytváření

bludišť je usnadněno aplikací pomocné mřížky. Úsečky lze pak snadno napojovat, protože krajní body se automaticky zaokrouhlují na definovanou hrubost mřížky. V každé iteraci Timeru se tedy vše smaže a znovu vykreslí, protože se vytvořené úsečky okamžitě ukládají do pole.

## 7 Vysvětlení klíčových funkcí a procedur

### 7.1 Curver

#### 7.1.1 Procedura Clear

Procedura Clear se stará o obnovení hrací plochy před opětovným vykreslením aktuální situace. Nejprve se pomocí bílého obdélníku smaže celá plocha, následně je na hřiště vykresleno bludiště. To je realizováno průchodem pole, kde jsou umístěny krajní body úseček. Jejich souřadnice musí být vynásobeny poměrem velikosti hřiště a velikosti plochy v editoru bludišť. Poté se na hřiště vykreslí úsečka po úsečce.

#### 7.1.2 Procedura Draw

Velmi důležitá procedura Draw se stará o vykreslení momentální situace na hrací plátno. Operace kromě toho, že vykreslí bonus, pokud je potřeba, jednoduchým cyklem každému hráči vykreslí hada. Je potřeba určit speciální případ pro hráče, který má bonus neviditelnosti a podle toho se chovat. Dále jsou v proceduře zakomponovány podmínky, díky nimž se v „hadovi“ vytvoří mezery a také ty, které vytvářejí efekt pohybu (pohyblivé proužky se světlejší barvou).

#### 7.1.3 Funkce Kolize

Tato funkce se stará o to, aby se zjistilo, jestli hráč v daném momentu nekoliduje s jiným hráčem nebo sám se sebou. Provádí se v každém kole Timeru pro každého hráče zvlášť. Vstupním parametrem je právě hráč. Dvěma vnořenými cykly zkontroluje každý bod každého „hada“ s výjimkou první bodu vlastního hada (to by kolidoval neustále) a zjistí, zda vzdálenost mezi ním a hlavou (hada) ze vstupního parametru není příliš nízká. V tom případě vrátí true.

### 7.2 Editor bludišť

#### 7.2.1 Funkce NaUsecce

Funkce se šesti parametry  $X1, Y1, X2, Y2, X, Y$  určí, zda bod se souřadnicemi  $[X, Y]$  leží či neleží na úsečce určené body  $[X1, Y1], [X2, Y2]$ . Funkce nejdříve zvlášť probere případy, které by v obecném výpočtu dělaly problémy (dělení nulou), a poté počítá podle obecného vzorce. Ten funguje tak, že porovnává parametr přímky určené původními body s parametrem přímky mezi jedním bodem z úsečky a bodem o který se zajímáme. Pokud jsou tyto parametry s určitou odchylkou rovné a zároveň se souřadnice  $x$  bodu  $[X, Y]$  nalézají mezi souřadnicemi  $x$  bodů z úsečky, bod leží na této úsečce a funkce vrátí true.

## 8 Možná rozšíření hry

Hru curver lze rozšířit mnoha způsoby. Je možné přidat několik dalších druhů bonusů. V tomto případě jsou možnosti téměř nevycerpitelné.

Další možné rozšíření by mohlo být zprovoznění síťové verze hry. Zde bude velkou výhodou to, že cesty hráčů jsou dynamicky uchovávány. To znamená, že je lze snadno přenášet. Dokonce mě napadlo zprovoznění ovládní hry na mobilním telefonu a pomocí například technologie Bluetooth ovládat „hady“. Pak by se ještě zvýšil možný počet hráčů. Dovedu si představit například několik hráčů hrajících na mobilním telefonu před velkým promítacím plátnem, na které by se hrací plocha promítala.

Curver je rozšiřitelný již nyní díky nespočetnému množství bludišť, které lze v editoru velice snadno a rychle vyrobit. Lze také rozšířit možnosti editoru bludišť. Mohly by být dostupné různé typy objektů atd.