



Gymnázium, Praha 6, Arabská 14
Piškvorky po síti, vedoucí práce Mgr. J. Lána



Piškvorky po síti

Ročníková práce

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne 3. května 2021

Anotace

Úkolem mé ročníkové práce bylo naprogramovat klasické piškvorky pro dva hráče a na pět znaků v řadě. Hráči budou muset hrát na jedné síti LAN (pokud si ručně nenastaví port-forwarding) a každý na svém vlastním počítači.

Zadání

Úkolem mé ročníkové práce bude naprogramovat klasické piškvorky pro dva hráče a na pět znaků v řadě v jazyce Java a za pomoci knihoven JavaFX a Java.Net. Piškvorky se musí hrát na dvou počítačích v síti LAN, aby bylo zadání mé práce rozdílné od zadání ostatních lidí.

Obsah

1. Úvod	1
1.1. Cíl práce.....	1
2. Piškvorky (Gomoku).....	1
2.1. Historie	1
2.2. Pravidla	2
2.3. Výpočetní řešení	3
3. Program	4
3.1. Použité technologie	4
1.3.1. <i>Java</i>	4
1.3.2. <i>Java.net</i>	4
1.3.3. <i>JavaFX</i>	4
3.2. Vzhled a GUI	5
3.3. Vyhodnocování výsledků	6
4. Zhodnocení	8
5. Zdroje	9
6. Seznam obrázků.....	10

1. Úvod

1.1. Cíl práce

Cílem této práce bylo naprogramovat jednu z nejhranějších her školních lavic, a to piškvorky, ale aby hráči nemuseli sedět spolu v lavici, tak se tato verze piškvorku hraje po síti.

Tuto práci jsem si vybral, protože na základní škole jsme piškvorky hodně hráli a dělali jsme třeba i piškvorkové turnaje, tudíž je to jedna z mých oblíbených her.

Tato práce se bude zabývat historií piškvorek, tvorbou mého programu, a nakonec i jeho problematikou.

2. Piškvorky (Gomoku)

2.1. Historie

Piškvorky vznikly pravděpodobně nezávisle na sobě v různých částech světa, ale první zmínky o období piškvorek pochází z Číny a jsou datovány do období 2 tisíce let před Kristem. Dále byly údaje o hře nalezeny i v antickém Řecku a předkolumbovské Americe.

Od této doby se piškvorky dobře uchytily v Japonsku se jménem Gomoku, což přímo znamená pět v řadě (go = pět, moku = průsečík), a toto jméno si piškvorky celosvětově zachovaly dodnes.



Obrázek 1 Gomoku (1)

V roce 1989 se v Japonsku konalo první mistrovství světa v Gomoku a dále tomu tak bylo opět za dva roky v roce 1991 ještě v Sovětském Svazu, poté se ale mistrovství na několik let přestala pořádat. Znovu byla obnovena v roce 2009 mistrovstvím v Pardubicích a od té doby se toto mistrovství pořádá každé dva roky.

V České republice se mistrovství konalo ještě jednou, a to v roce 2017 v Praze, první Čech Martin Muzika ale mistrovství vyhrál až v roce 2019 v Estonsku. Dále se konají také týmová mistrovství, ale ta jsou na rozdíl od těch pro jednotlivce každý sudý rok, a ne každý lichý.



Obrázek 2 Mistrovství světa v Praze 2017 (2)

2.2. Pravidla

Nyní se ale pojďme podívat na pravidla piškvorek, vysvětlíme si základní pravidla, která se používají na světových turnajích. Hra začíná tím, že začínající hráč umístí na herní plochu 15x15 3 znaky, dva křížky a jedno kolečko, a druhý hráč má nyní na výběr ze 3 různých tahů. Buďto bude hrát za křížky nebo bude hrát za kolečka a položí další kolečko, anebo položí další kolečko a křížek a nechá vybírat druhého hráče. Poté už se hráči střídají v tazích a snaží se dosáhnout pěti svých znaků v řadě. Pokud hrajete standard piškvorky, tak musíte dosáhnout přesně pěti a pokud hrajete freestyle piškvorky, tak můžete dosáhnout pěti a více.

Piškvorky ale kromě těchto pravidel mají také různé další varianty, mezi nimiž například Rendžu, Šestvorky, Pentago a další. Dokonce existují i 3D piškvorky, ve kterých se nehraje na plánu 15x15 ale 8x8x8, ale jako obvykle je za úkol vytvořit nepřerušovanou řadu pěti svých znaků.

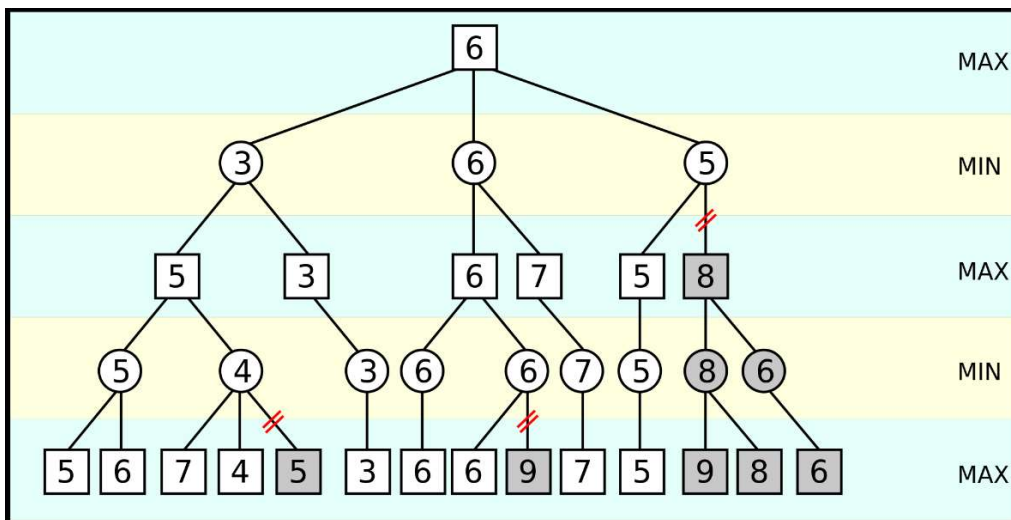
2.3. Výpočetní řešení

V této kapitole se podíváme na výpočetní řešení piškvorek. Jak na omezeném, tak i neomezeném poli má začínající hráč vždy vítěznou strategii, to plyne z toho, že kdykoliv by hráč, který hraje až druhý, měl vítěznou strategii, tak by ji mohl použít začínající hráč. Pokud nezačínající hráč změní svou strategii, poté už není ta původní vítězná, protože nemůže existovat vítězná strategie pro začínajícího i nezačínajícího hráče. Když je strategie nezačínajícího hráče změněna, může být opět použita druhým hráčem, tudíž nezačínající hráč nikdy nemá vítěznou strategii.

Tuto výhodu začínajícího hráče se snaží vyrovnat pravidlo Swap2, které jsem v podstatě popsal v kapitole Pravidla. Jedná se o variantu Koláčového pravidla, které se využívá například v šachách. Toto pravidlo funguje na základě výběru strany až po prvním tahu, kdy první hráč táhne jednou bílou figurou a poté si druhý hráč může vybrat, jestli se mu tento tah líbí a chce hrát za bílé nebo jestli chce hrát za černé.

Fakt, že začínající hráč má na klasickém 15x15 poli vítěznou strategii, dokázal jako první Holandský expert v informatice L. Victor Allis, vyhrávací strategii má ale samozřejmě začínající hráč i na větších herních plochách, protože druhý hráč nemůže toto pole, narozdíl od začínajícího hráče, využít pro vynucení remízy.

Pokud hrajete proti počítačovému protihráči, tak ten je nejspíše založen na algoritmu minimax, který funguje na základě vyzkoušení všech možností několik pohybů dopředu a následném obodování jejich výsledků. Tento algoritmus sice funguje velmi dobře, ale s každým přidaným pohybem pro lepší přesnost se čas a výpočetní síla pro jeho provedení několikanásobně zvyšuje, proto se na něj aplikují různé optimalizace. Jednou z těchto optimalizací je například alfa-beta ořezávání, což spočívá ve sledování půltahů, a pokud už se nějaký nemůže vyrovnat ostatním, tak už se nepočítají jeho důsledky a je rovnou vyřazen.



Obrázek 3 Ukázka alfa-beta ořezávání na algoritmu minimax (3)

3. Program

3.1. Použité technologie

1.3.1. Java

Java je univerzální multiplatformní programovací jazyk, který je objektově orientovaný a navržený tak, aby měl co nejméně implementačních závislostí. Jeho cílem je umožnit vývojářům „napsat jednou, spustit kdekoli,“ což znamená, že kompilovaný kód Java může běžet na všech platformách, které podporují Javu bez nutnosti kompilace. Java aplikace jsou většinou kompilovány do byte kódu, který může běžet na libovolném virtuálním stroji (Java Virtual Machine) bez ohledu na architekturu počítače. Původně byl vyvinutý firmou Sun Microsystems v roce 1995. V současnosti je Java jedním z nejpobulárnějších programovacích jazyků s nahlášenými 9 miliony vývojářů. (4)

1.3.2. Java.net

Java.net je Java balíček, umožňující TCP a UDP připojení mezi Java aplikacemi a pomáhá tak snadno vytvářet aplikace, jako je ta moje.

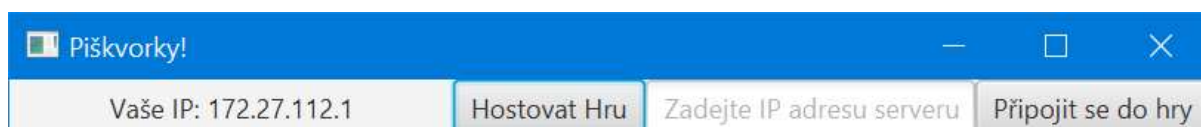
1.3.3. JavaFX

JavaFX je softwarová platforma postavená na bázi platformy Java z dílny společnosti Sun Microsystems. Slouží pro vývoj tzv. RIA aplikací (Rich Internet Applications), v překladu bohatých internetových aplikací. Jak již z názvu vyplývá, jedná se především o prvky zajišťující interaktivitu webových stránek, tedy stránek, které s uživatelem, oproti těm statickým, nějakým způsobem komunikují. V případě JavaFX webových aplikací, které se formou pluginu spouští z prohlížeče, navíc platforma umožňuje si aplikaci stáhnout a využívat ji jako klasický desktopový program, a to i v offline módu. JavaFX je cílena na vývojáře programů toužících po rychlém a snadném vývoji pro desktopy, internetové prohlížeče, ale i rychle rozvíjející se mobilní oblast. Vznikla jako reakce na masové rozšíření platforem jako je Adobe Flash, a především Microsoft Silverlight. (5)

3.2. Vzhled a GUI

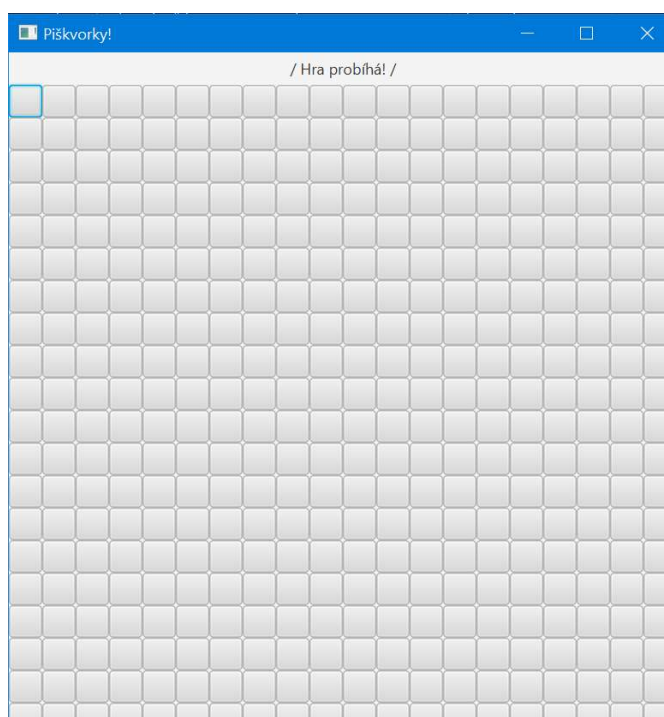
V samotné aplikaci můžete za celou hru vidět dvě okna, jedno z nich je dialogové okno pro výběr mezi hostováním hry a připojením k již hostované hře, a to druhé už je hra samotná.

Po spuštění programu se otevře základní okno se otevře dialogové okno pro připojení, které má v sobě 4 komponenty, prvním z nich je Label, který ukazuje vaši IP adresu na nejvyšším síťovém adaptéru, to však nemusí být vždy ta správná. Když program otevřu na svém hlavním počítači, tak se mi místo mé správné IP adresy ukáže loopback adresa WSL (Windows Subsystem for Linux), což samozřejmě není dobře, ale bohužel jsem nenašel, jak něco takového vyřešit. Dále můžete v dialogovém okně vidět tlačítko Hostovat hru, Text Field na zadání IP adresy hostovacího hráče a tlačítko Připojit se do hry.



Obrázek 4 Ukázka dialogového okna pro připojení (6)

Pokud si uživatel zvolí, že chce hostovat hru, tak se chod aplikace zastaví a čeká se na připojení od druhého hráče, a jakmile se připojí, tak se oběma načte hrací pole. Pokud si uživatel zvolí, že se chce připojit do hry, tak se musí nejdříve ujistit, že už druhá hráč vybral Hostovat hru, jinak nebude připojen. Pak už mu jen stačí zadat IP adresu hostovacího hráče a kliknout Připojit se do hry.



Obrázek 5 GUI samotné hry (7)

Jakmile se herní pole načtou, hráči mohou začít hrát. První na řadě je vždy připojující se hráč, tak ten může nyní položit svůj první znak. Je nutné podotknout, že jsem pro mou verzi piškvorek usnadnil pravidla na klasické piškvorky, které by si každý zahrál na kusu papíru, tudíž neobsahují žádné pravidlo Swap2 a hrací plocha je zvětšena na 20x20. Poté už se hráči jenom střídají, dokud jeden z nich nevyhraje nebo se nezaplní celé pole – dojde k remíze.

3.3. Vyhodnocování výsledků

Naprosto nejkomplikovanější částí programu je bezesporu funkce pro zjišťování, jestli už jeden z hráčů nevyhrál. Tuto funkci má v sobě vestavěné každé tlačítko a kdykoliv jeden z hráčů odehraje, tak se spustí. Funguje tak, že projde všech 400 tlačítek a u každého vyzkouší, jestli nemá vedle sebe další tlačítko se stejným znakem ve směru, ve kterém tato část funkce kontroluje. Pokud má, tak to tak udělá ještě třikrát, a pokud to vyjde, tak hráč s daným znakem vyhrál. Tato funkce už takto fungovala, ale měla jeden problém, a to že jste na okrajích se svou horizontální řadou mohli pokračovat o jednu řadu níže, protože hrací plocha nevyužívá souřadnicový systém s x a y, ale jenom s x, jak je vidět na obrázku 7. Implementoval jsem proto do každého if ještě všechny postranních pozic, a pokud se jedné z nich pozice rovná, tak už se funkce dál nekouká.

```
try {  
    if (tlacitko(buttonList.get(k)).equals("X") && k != 19 && k != 39 && k != 59 && k !=  
        if (tlacitko(buttonList.get(k + 21)).equals("X") && k + 21 != 19 && k + 21 != 39  
            if (tlacitko(buttonList.get(k + 42)).equals("X") && k + 42 != 19 && k + 42  
                if (tlacitko(buttonList.get(k + 63)).equals("X") && k + 63 != 19 && k +  
                    if (tlacitko(buttonList.get(k + 84)).equals("X")) {  
                        anim = 1;  
                        animCount();  
                    }  
                }  
            }  
        }  
    }  
} catch (IndexOutOfBoundsException e) {  
}
```

Obrázek 6 Část funkce pro kontrolu vítězství (8)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99
100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119
120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139
140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179
180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199
200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219
220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259
260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279
280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299
300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319
320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339
340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359
360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379
380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399

Obrázek 7 Výpis souřadnic na poli (9)

4. Zhodnocení

Pro mne byla tato práce asi taková, jakou jsem ji čekal. Nebyla o nic těžší, ale ani o nic lehčí, i když jsem měl trochu problém s posíláním dat po síti, protože jsem to dělal poprvé v životě. Trochu mě mrzí, že má program čas od času nějaké malé chyby, například znak, který jste položili vy se objeví až když odehraje i protihráč, ale to je jen maličkost. Jinak jsem se svou prací spokojen a jsem rád, že se mi ji podařilo vytvořit.

5. Zdroje

1. *Gomoku World*. [Online] 2009. [Citace: 30. 4 2021.] <http://gomokuworld.com/gomoku/1>.
2. *Gomoku World Tournaments*. [Online] 2017. [Citace: 30. 4 2021.] <http://gomokuworld.com/tournaments/191>.
3. *Wikipedia*. [Online] 2021. [Citace: 30. 4 2021.] https://cs.wikipedia.org/wiki/Alfa-beta_o%C5%99ez%C3%A1v%C3%A1n%C3%AD.
4. *IT Slovník*. [Online] 2008. [Citace: 30. 4 2021.] <https://it-slovník.cz/pojem/java>.
5. *Wikipedia*. [Online] 2020. [Citace: 30. 4 2021.] <https://cs.wikipedia.org/wiki/JavaFX>.

6. Seznam obrázků

Obrázek 1 Gomoku (1)	1
Obrázek 2 Mistrovství světa v Praze 2017 (2)	2
Obrázek 3 Ukázka alfa-beta ořezávání na algoritmu minimax (3)	3
Obrázek 4 Ukázka dialogového okna pro připojení (6)	5
Obrázek 5 GUI samotné hry (7)	5
Obrázek 6 Část funkce pro kontrolu vítězství (8)	6
Obrázek 7 Výpis souřadnic na poli (9)	7