**Pawn game – Dokumentace**

Jako zápočtový program jsem si vybral hru s názvem Pawn game. Vytvořil jsem program, ve kterém si ji budete moci zahrát proti jinému hráči, či proti počítači.

**Pravidla**

Pawn game je desková hra, která se hraje na šachovnici 8x8.Hra se hraje se šachovými pěšci, kteří začínají na stejných polích jako ve hře šachy. Pěšci se v Pawn game pohybují stejně jako v šachách.

Stejně jako v šachách hru začíná bílý hráč. Vítězí ten hráč, který jako první dostane jednoho ze svých pěšců na protilehlý okraj šachovnice, nebo ten který sebere všechny soupeřovi figurky.

I v této hře může nastat remíza a ta nastává pokud hráč, který je na tahu, nemůže táhnout ani jedním svým pěšcem.

**Ovládání programu**

Program vám po celou dobu napovídá jaký vstup přesně zadat. Vstupy nesmí začínat a končit mezerou. Vstupy mohou být zadány velkými nebo malými písmeny (písmena se mohou i střídat např.: vstup: AnO). Zadání vašeho tahu má přesný formát a ten je: pole, na kterém stojí pěšec + mezera + pole na které chcete tímto pěšcem táhnout (př.: E2 E4). Pole jsou označena stejně jako v šachách. První znak je písmeno A-H a druhý znak je číslo 1-8. V případě špatného vstupu vám program vypíše chybu a vyzve vás k opětovnému zadání vstupu.

Součástí programu je i možnost uložit si poslední hru ve formě textu podobného šachovému partiáři. Je ovšem možné mít uloženou pouze jednu hru. Pokud uložíte novou hru, přepíšete tak hru starou.

**Výstup programu**

Výstup programu je stejný jako uživatelský vstup. Výstupem programu je text ve formátu: pole z kterého počítač táhne + mezera + pole na které počítač táhne (př.: D7 D5).

Jakmile skončí hra, program vypíše výsledek hry (př.: Počítač vyhrál, REMÍZA, apod.).

**Program**

Program je řízený vstupy uživatele. Obsahuje dvě rozhraní uživatel proti uživateli a uživatel proti počítači.

V programu je použita třída nazvaná „pěšec“, která slouží zejména k zjednodušení problematiky tahů figurek.

V rozhraní uživatel proti počítači se rozhoduje počítač o svém tahu pomocí minimaxového algoritmu. Před každým svým tahem prozkoumá počítač strom hry do určité hloubky, a poté se rozhodne jaký tah zvolit.

Tah počítač volí na základě výsledku statické ohodnocovací funkce, která se v dané hloubce podívá na pozici na šachovnici, pozici ohodnotí a vrátí číslo, které pozici přiřadí. Na základě tohoto čísla se minimaxový algoritmus rozhoduje, který tah zvolí.

**Zhodnocení mé práce**

Nejprve musím říci, že jsem se svým programen spokojen. Na začátku jsem měl dva cíle, jeden menší a druhý větší. Mým menším cílem bylo napsat program, který bezpečně porazí program Pawn Game, který lze stáhnout na Google play. Tento cíl jsem splnil na výbornou. Netuším kolikrát jsem tyto dva programy proti sobě postavil, ale pokaždé zvítězil právě ten můj.

Druhý cíl byl o něco větší a to naprogramovat program, který já sám neporazím. To se mi ovšem nepovedlo. Se svým programem jsem několikrát hrál a zjistil jsem, že je mi rovnocenným soupeřem, ale i přesto jsem ho schopný porazit. Sice jsem nesplnil svůj druhý cíl, ale i tak mám z finální verze programu dobrý pocit, i když si uvědomuji, že je co zlepšovat.

**Co zlepšit?**

Ke zlepšení je toho na tomto programu více. Nejprve to co mě trápí nejvíce a tím je fakt, že program sice odpovídá na tahy, ale grafickým výstup je pouze šachovnice a tah, který počítač zahrál. Nelze tedy hrát způsobem, že kliknu na figurku a táhnu tak, kam chci aby figurka hrála.

Další věcí ke zlepšení je doprogramování alfa-beta ořezávání u minimaxového algoritmu a následné prohledávání stromu hry do větší hloubky a tím zvýšení obtížnosti počítače.

Další možností jak zvýšit obtížnost počítače je vylepšení statické ohodnocovací funkce tak, aby vracela přesnější ohodnocení pozice. Avšak já osobně si myslím že je tato funkce napsaná velmi dobře.

Samozřejmě je možné hru naprogramovat jiným způsobem, nepoužívat třídy či úlohu uchopit za úplně jiný konec. Já jsem zvolil tento způsob a jsem si vědom toho, že to nemusí být ten nejlepší.