

```
/*
Popis:
Vybrané ledky budou blikat cca 2x za vteřinu

Ledky které mají takto blikat jsou vybrány konstantou "LED" a čas svícení je dán
děličkou časovače a šířkou jeho registru.
V tomto příkladu bude použit čítač/časovač "0" který má šířku registru 8 bitů
to znamená že počítá do 255.
Čítač bude mít zapnutou děličku hodinového signálu s dělitelem 1024.
Výsledná frekvence tedy bude 1 000 000/1024/255 = 3,8 Hz - při každém přetečení
budeme negovat stav ledek, takže se rozsvítí a zhasnou přibližně 2x za sekundu.
Přetečení čítače je signalizováno nultým bitem registru "TIFR"

*/

#include <avr/io.h> //Vlož knihovnu vstupů a výstupů (PORT, DDR, PIN)

#define LED 0xff; //Ledky, které se mají blikat (všechny)

/*****HLAVNÍ FUNKCE*****/

int main (void) //hlavní funkce
{
    DDRB = 0xff; //(0xff = 0b11111111) -> Piny 0 - 7 portu "B" budou výstupní
    PORTB = LED; //rozsvít' vybrané ledky

    TCCR0 = 5; //spust' časovač "0" s předděličkou 1024

    while (1) // Nekonečná smyčka (dokud 1 = pořád)
    {
        if((TIFR & 0b00000001)==0)//pokud je nultý bit "0" - ještě nepřetekl
        {
            ; //nedělej nic (prázdný příkaz)
        }
        else //pokud není nultý bit "0" - časovač časovač už přetekl
        {
            PORTB = ~PORTB; //vyměň na portu "B" všechny nuly za jedničky a naopak
            TIFR=TIFR | 0b00000001; //Nuluj příznak přetečení v registru TIFR
                //(zapsáním "1" -poněkud nelogické, ale je to tak)
        }

    } //Konec cyklu "while (1)" - program skočí zpět na jeho začátek

} //konec funkce main() - sem se program nikdy nedostane
//(závorka tu však musí být jinak by překladač nahlásil chybu)
```