

Sprawozdanie zadanie 5

“Programator mikroprocesora”

Organizacja systemów komputerowych

Adrian Nowogrodzki 184332
Oskar Nowak 184289

The screenshot shows a Windows application window titled "Form1". At the top, there is a text instruction: "wybierz z listy lub wprowadź liczbę całkowitą dodatnią" (choose from the list or enter a positive integer). Below this, there are three dropdown menus for selecting an operation and registers. The first dropdown is set to "MOV", the second to "CH", and the third to "BX". To the right of these is an "OK" button. Below the dropdowns is a list box containing the instruction "1 ADD BX 7" and "MOV CH BX". To the right of the list box is a table showing the current values of registers AX, BX, CX, and DX in decimal (DEC), hexadecimal (NH), and binary (NL) formats. Below the table are five buttons: "wykonaj program", "wykonaj krok", "wyczyszc", "zapisz do pliku", and "odczytaj z pliku". At the bottom left, there is a "zresetuj program" button.

	DEC	NH	BIN	NL
AX	0	00000000	00000000	00000000
BX	7	00000000	00000111	00000111
CX	0	00000000	00000000	00000000
DX	0	00000000	00000000	00000000

Program umożliwia wykonywanie operacji ADD, SUB, MOV. Możliwa jest praca krokowa lub wykonanie całego programu na raz. Kod można zapisać do pliku tekstowego oraz można odczytać zawartość pliku do kodu programu. W odpowiednich miejscach wyświetlane są wartości poszczególnych rejestrów oraz ich ekwiwalent dziesiętny. Kod można dodawać za pomocą wprowadzania go w ComboBoxach i zatwierdzając "OK" lub w pliku tekstowym. Program posiada zabezpieczenia przepelniające oraz przed wprowadzeniem liczb ujemnych.

Po wciśnięciu “wykonaj program” - program linijka po linijce będzie wykonywał operacje

```
for (int i = 0; i < il_rozkazow; i++)  
{  
    Krok_działania_programu(i);  
}
```

```
private void Krok_działania_programu(int ktora_linia)  
{  
    wyraz_linii_programu = textBox1.Lines[ktora_linia].Split(' ');  
  
    if (wyraz_linii_programu[0] == "ADD")    // na podstawie pierwszej  
    {  
        ADD_function();  
    }  
    if (wyraz_linii_programu[0] == "SUB")  
    {  
        SUB_function();  
    }  
    if (wyraz_linii_programu[0] == "MOV")  
    {  
        MOV_function();  
    }  
    Odswiez();  
}
```

i w razie wystąpienia błędu ustawi flagę.

```
private void ADD_function()  
{  
    Pobieranie_temp_i();    // pob  
    Sumowanie();            // dod  
  
    if (temp_i < 0 || A_i < 0 || A  
    {  
        blad = true;    // jeśli w  
    }  
}
```

Flaga ta uniemożliwia wykonanie się programu.

Bazową formą zapisu danych jest int,

```
else if (wyraz_linii_programu[2] == "DH")
{
    temp_i = D_i / 256;
}
else if (wyraz_linii_programu[2] == "AL")
{
    temp_i = A_i % 256;    // _L oznacza
```

który później jest transformowany do formy bitowej a następnie na _L i _H,

```
bajty = BitConverter.GetBytes(A_i);
druga_tablica = bajty[0];
pierwsza_tablica = bajty[1];
```

które można wyświetlić.

```
label_ah_bin.Text = Convert.ToString(pierwsza_tablica, 2).PadLeft(8, '0'); // wyświetlanie AH
label_al_bin.Text = Convert.ToString(druga_tablica, 2).PadLeft(8, '0');    // wyświetlanie AL
```