Systemy Operacjyne 2

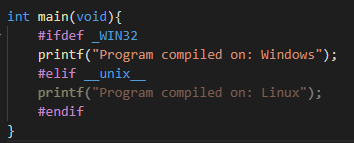
Andriy Adamovych

Informatyka 3 rok lab1

**Laboratorium 7**

**Zad. 1**

**Napisz program check sprawdzający na jakim systemie został skompilowany. Wykorzystać kompilację warunkową.**



Wynik:



**Zad. 2**

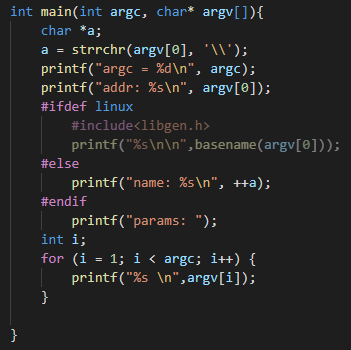
**Napisz program arguments wypisujący kolejno:**

- wartość zmiennej argc

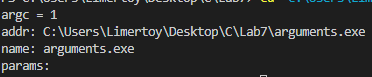
- adres programu

- nazwę programu \*

- przekazane parametry

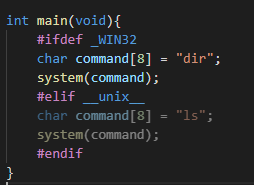


Wynik:



**Zad. 3**

**Napisz program list wywołujący komendę dir dla systemu Windows lub ls dla systemu Linux. Parametry podane przy uruchomieniu programu są przekazywane do obu komend.**



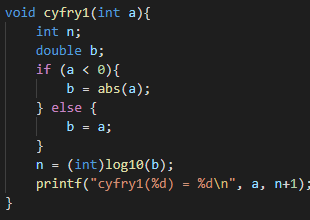
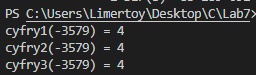
Wynik:



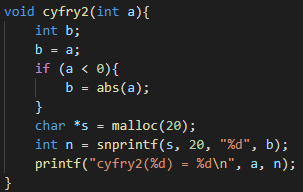
**Zad. 4**

**Napisz program cyfry wyliczający liczbę cyfr danej liczby całkowitej przy pomocy funkcji:**

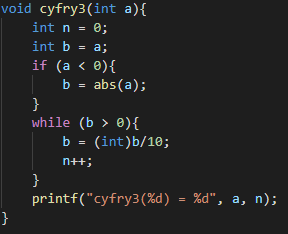
**- funkcja cyfry1 wykorzystuje funkcję log10. \* Wynik:**

****

**- funkcja cyfry2 wykorzystuje funkcję snprintf.**

****

**- funkcja cyfry3 wykorzystuje dzielenie przez 10. \***



**Zad. 5, 6**

**Napisz program dodawanie realizujący dodawanie pisemne. Interfejs programu musi wyglądać dokładnie tak samo, jak w programie dodawanie.exe na stronie autora. Zakładamy, że dane wejściowe mają postać:**

Z: a = 0, 1, 2, ...

b = 0, 1, 2, ...

**- w jakich przypadkach program może dawać niepoprawne wyniki?**

Kiedy podaję liczby mniejsze od 0, wtedy jedynki nad dodawaniem nie pokazują się.

**Dodaj do programu dodawanie kontrolę poprawności danych wejściowych i wyniku dla przypadków:**

**- dane wejściowe pobierane są jako liczby całkowite**

**- dane wejściowe pobierane są jako łańcuchy znaków**

#include <ctype.h>

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main(void)

{

    int a, b, i = 9;

    int k, j, res;

    printf("a = ");

    scanf("%d", &a);

    printf("b = ");

    scanf("%d", &b);

    res = a + b;

    if (a > 0 && b > 0)

    {

        int tab[4][10] = {{}, {}, {}};

        for (k = 0; k < 4; k++)

        {

            for (j = 0; j < 10; j++)

            {

                tab[k][j] = -1;

            }

        }

        while (a != 0)

        {

            tab[1][i] = a % 10;

            a /= 10;

            i--;

        }

        i = 9;

        while (b != 0)

        {

            tab[2][i] = b % 10;

            b /= 10;

            i--;

        }

        i = 9;

        while (res != 0)

        {

            tab[3][i] = res % 10;

            res /= 10;

            i--;

        }

        for (j = 9; j > 0; j--)

        {

            if ((tab[1][j] + tab[2][j]) >= 10)

            {

                tab[0][j - 1] = 1;

            }

            else

            {

                tab[0][j - 1] = -1;

            }

        }

        for (k = 0; k < 4; k++)

        {

            if (k == 0 || k == 1 || k == 3)

            {

                printf(" ");

            }

            else if (k == 2)

            {

                printf("+");

            }

            for (j = 0; j < 10; j++)

            {

                if (tab[k][j] != -1)

                    printf("%d", tab[k][j]);

                else

                    printf(" ");

            }

            if (k == 2)

            {

                printf("\n     --------");

            }

            printf("\n");

        }

    }

    else

    {

        printf("  %d\n", a);

        printf("+ %d\n", b);

        printf("-------");

        printf("\n");

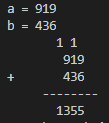
        printf("  %d", res);

    }

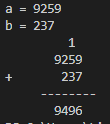
}

**Wynik:**

**Taki sam rozmiar liczb:**

****

**Różny rozmiar liczb:**

****