
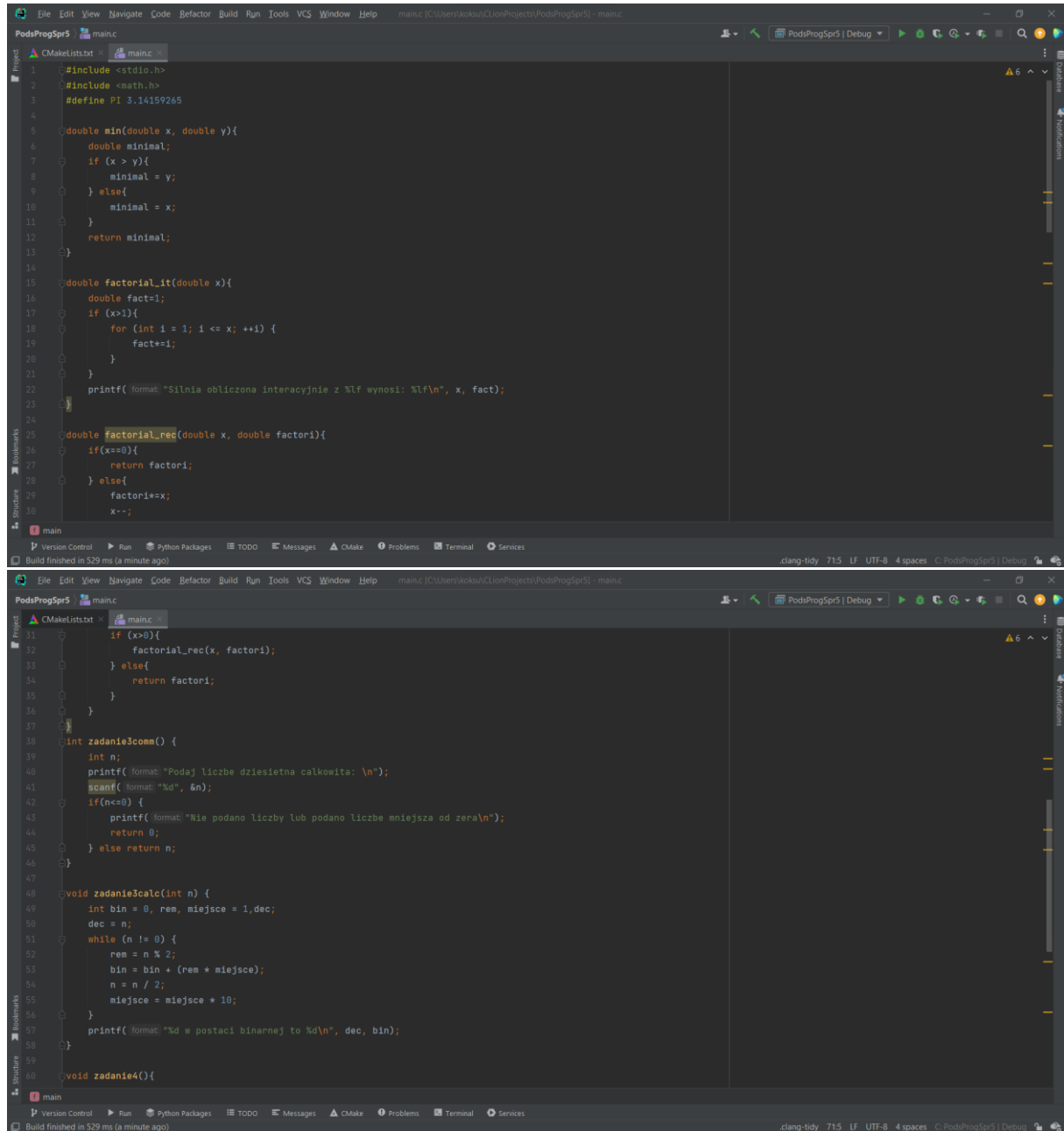


## SPRAWOZDANIE NR 5

Nazwa ćwiczenia	Funkcje i rekurencja.		 <b>POLITECHNIKA BYDGOSKA</b> Wydział Telekomunikacji, Informatyki i Elektrotechniki
Przedmiot	Podstawy programowania – laboratorium		
Student grupa	Marcin Ogórkiewicz, grupa 7		
Data ćwiczeń	30.11.2022	07.12.2022	Data oddania sprawozdania



```
#include <stdio.h>
#include <math.h>
#define PI 3.14159265

double min(double x, double y){
    double minimal;
    if (x > y){
        minimal = y;
    } else{
        minimal = x;
    }
    return minimal;
}

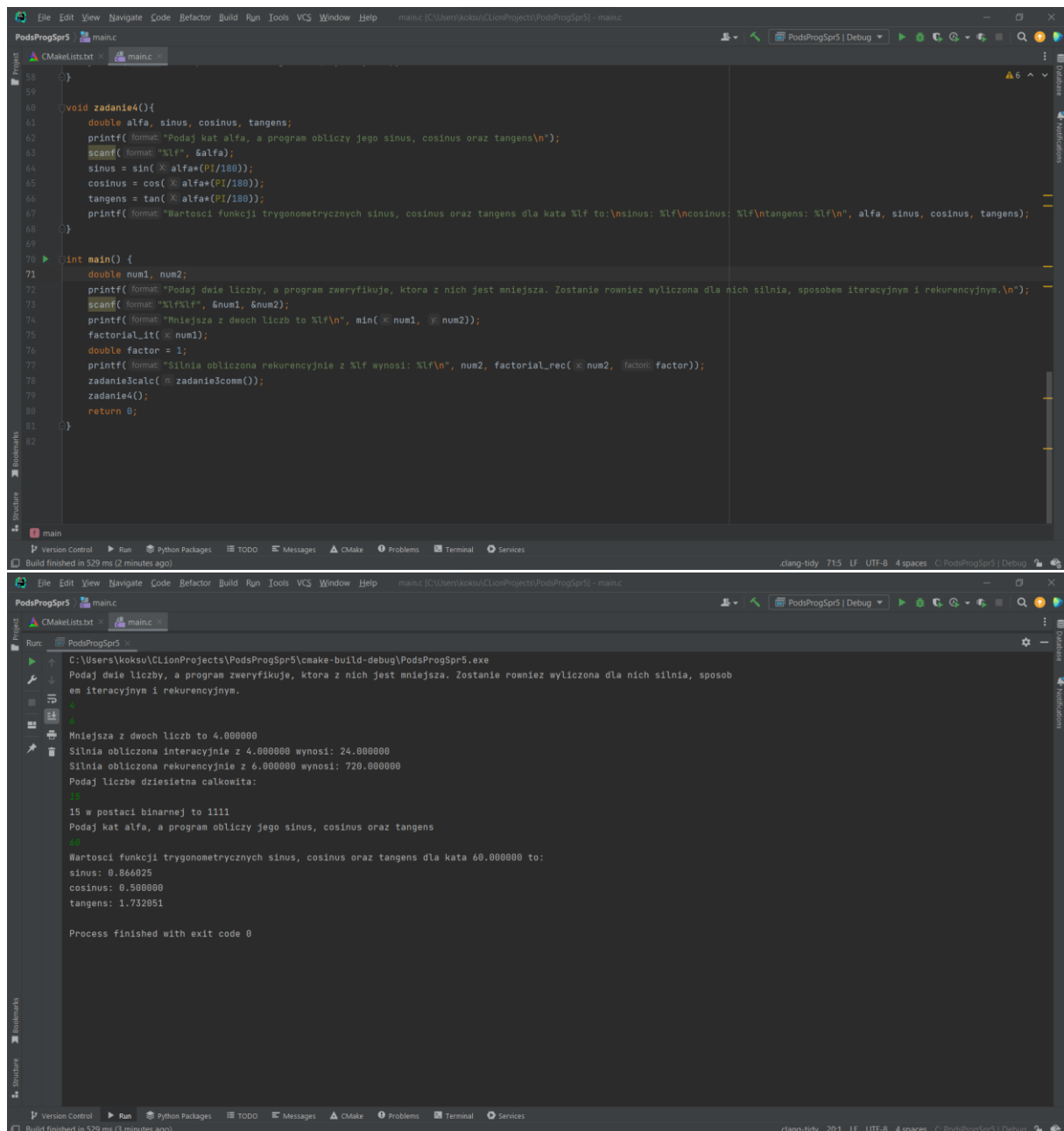
double factorial_it(double x){
    double fact=1;
    if (x>1){
        for (int i = 1; i <= x; ++i) {
            fact*=i;
        }
    }
    printf("Silnia obliczona iteracyjnie z %lf wynosi: %lf\n", x, fact);
}

double factorial_rec(double x, double factori){
    if(x==0){
        return factori;
    } else{
        factori*=x;
        x--;
    }
}

int zadanie3comm() {
    int n;
    printf("Podaj liczbe dziesietna calkowita: \n");
    scanf(" %d", &n);
    if(n<=0) {
        printf("Nie podano liczby lub podano liczbe mniejsza od zera\n");
        return 0;
    } else return n;
}

void zadanie3calc(int n) {
    int bin = 0, rem, miejsce = 1, dec;
    dec = n;
    while (n != 0) {
        rem = n % 2;
        bin = bin + (rem * miejsce);
        n = n / 2;
        miejsce = miejsce * 10;
    }
    printf(" %d w postaci binarnej to %d\n", dec, bin);
}

void zadanie4(){
}
```



## Zadanie 1

W części głównej programu, użytkownik zostaje poproszony o wprowadzenie dwóch liczb w celu sprawdzenia, która z nich jest mniejsza oraz zostaje poinformowany o wykorzystaniu tych liczb do obliczenia silni sposobem iteracyjnym i rekurencyjnym. Zostają nimi nadpisane kolejno uprzednio zainicjowane zmienne `num1` oraz `num2`. Funkcja `min` pobiera wartości podane przez użytkownika oraz weryfikuje, która jest mniejsza, prostą instrukcją warunkową `if`, po czym zwraca wartość mniejszą. Program następnie wyświetla informację o tym, która liczba jest mniejsza, przy użyciu wartości zwróconej przez funkcję `min`.

## Zadanie 2

### Sposób iteracyjny

Do wyliczenia silni iteracyjnie oraz rekurencyjnie funkcje wykorzystują zmienne `num1` oraz `num2` z poprzedniego zadania. W funkcji liczącej sposobem iteracyjnym zostaje zdefiniowana zmienna `fact` =

1. Będzie ona służyła do przechowywania wartości silni, ponieważ najmniejsza silnia, którą da się obliczyć wynosi 1. Następnie, w pętli for, zmienna ta zostaje przemnożona przez zmienną i, która pozwoli na przemnożenie zmiennej fact przez każdą liczbę całkowitą ze zbioru <1; liczba podana przez użytkownika>. Po każdej iteracji pętli, i jest zwiększane. Gdy i osiągnie wartość liczby podanej przez użytkownika, program wypisuje wartość silni i kończy działanie.

### **Sposób rekurencyjny**

Nazewnictwo zmiennych w tej części programu jest nieco dziwne, ponieważ bałem się, iż nadanie tych samych nazw zmiennych mogłoby zaburzyć działanie programu. Funkcja rekurencyjna wykorzystuje liczbę podaną przez użytkownika w zadaniu poprzednim oraz zmienną factor=1, która przechowuje wynik w funkcji głównej. Jeżeli liczba wynosi 0, program od razu zwraca wartość 1 i wypisuje ją dla użytkownika w części głównej. W przeciwnym razie, program przemnaża zmienną factor (w funkcji nazywa się factori) przez liczbę podaną przez użytkownika, a następnie zmniejsza liczbę podaną przez użytkownika o 1. Następnie, program sprawdza czy liczba podana przez użytkownika jest nadal większa od 0. Jeśli tak, funkcja zostaje ponownie wywołana z parametrami x(liczba podana przez użytkownika) oraz factori, które przechowuje wynik. Gdy warunek nie jest spełniony, funkcja zwraca factori. Na koniec, w głównej funkcji, program wyświetla wynik działania funkcji rekurencyjnej użytkownikowi.

### **Zadanie 3**

#### **Komunikacja z użytkownikiem**

Program prosi użytkownika o wpisanie liczby całkowitej dziesiętnej, po czym pobiera ją i przechowuje we wcześniej utworzonej zmiennej n. Na koniec weryfikuje czy liczba jest mniejsza bądź równa 0. Jeśli tak, program zwraca komunikat, iż nie podano liczby bądź jest ona mniejsza od 0, po czym zwraca 0, a w przeciwnym razie zwraca liczbę podaną przez użytkownika.

#### **Konwersja liczby dziesiętnej na binarną**

Program definiuje zmienne typu int bin=0, miejsce=1 oraz dec=n, a także inicjuje zmienną rem. Następnie rozpoczyna się działanie pętli while, która będzie funkcjonowała dopóki n nie jest równe 0. Zmienna rem przyjmuje wartość reszty z dzielenia n przez 2, po czym do bin zostaje dodany iloczyn rem i miejsce. Następnie n zostaje dwukrotnie pomniejszone, a miejsce zostaje 10-krotnie powiększone. Po zakończeniu działania pętli, program wyświetla użytkownikowi komunikat z liczbą podaną przez niego w systemie dziesiętnym, oraz tą samą liczbą, ale w systemie binarnym. Obydwie te wartości są przechowywane odpowiednio przez zmienne dec oraz bin.

### **Zadanie 4**

Na początku kodu definiuję wartość liczby PI, z której będzie korzystała funkcja. Będzie ona potrzebna do przekonwertowania wartości kąta ze stopni na radiany, ponieważ funkcje trygonometryczne z biblioteki math.h działają na radianach. Następnie program inicjuje zmienne typu double: alfa, sinus, cosinus, tangens, po czym pobiera od użytkownika wartość kąta, którą nadpisuje zmienną alfa. Później program wylicza i przypisuje wartości odpowiednich funkcji trygonometrycznych do wcześniej podanych zmiennych i w międzyczasie zachodzi konwersja na radiany. Na koniec, program wypisuje wartości kąta alfa oraz wyliczonych funkcji trygonometrycznych.