

Imię i nazwisko	Kierunek	Rok studiów i grupa
Patryk Kozłowski	Informatyka techniczna	1 rok grupa 2
Data zajęć	Numer i temat sprawozdania	
16.10.2024	3. Schematy blokowe (pętle, instrukcje warunkowe, funkcje)	

Niniejsze sprawozdanie dotyczy zagadnień programistycznych, obejmujących różnorodne zadania, które rozwijają umiejętności operowania na danych oraz pracy z funkcjami w języku programowania, takim jak Python. Głównym celem tych zadań jest kształtowanie myślenia algorytmicznego oraz umiejętności tworzenia funkcjonalnych programów rozwiązujących konkretne problemy.

Pierwsze zadanie polega na stworzeniu funkcji, która oblicza całkowitą liczbę nóg różnych zwierząt na farmie. Farmer hoduje kurczaki, krowy oraz świnie, a każdemu z tych zwierząt przypisana jest określona liczba nóg: kurczaki mają po 2 nogi, natomiast krowy i świnie po 4. Na podstawie liczby każdego gatunku podanej przez rolnika, program ma zsumować wszystkie nogi i wyświetlić wynik na ekranie. Jest to klasyczne zadanie z wykorzystaniem funkcji arytmetycznych, które pozwala zrozumieć, jak można efektywnie operować na liczbach, przypisując im odpowiednie wartości w kontekście rzeczywistego problemu.

Kolejne zadania obejmują m.in. algorytmy sprawdzające, czy dany rok jest przestępny, wypisywanie liczb z podanego przedziału, oraz pisanie prostych gier, takich jak „kamień, papier, nożyczki”. Dzięki nim można przećwiczyć takie aspekty programowania jak warunki logiczne, pętle, operacje na listach oraz interakcję z użytkownikiem.

Zadanie 1

```
In [ ]: #include <iostream>

int animals(int kurczaki, int krowy, int swinie){
    return (kurczaki * 2 + krowy * 4 + swinie * 4);
}

int main(void){
    int k, kr, s;
    std::cout << "podaj ilosc kurczakow, krow, swin oddzielając je spacją: ";
    std::cin >> k >> kr >> s;

    std::cout << "można zliczyć " << animals(k, kr, s) << " nog" << std::endl;

    return 0;
}
```

```
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/01$ g++ z01.cpp -o zadanie1
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/01$ ./zadanie1
podaj ilosc kurczakow, krow, swin oddzielajac je spacja: 1 2 3
mozna zliczyc 22 nog
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/01$
```

Zadanie 2

```
In [ ]: // 01.cpp

#include <iostream>

using namespace std;

void zad01(){
    int rok;
    cout << "podaj rok: ";
    cin >> rok;
    if ((rok % 4 == 0) && ((rok % 100 != 0) || (rok % 400 == 0))){
        cout << "jest przestepny" << endl;
    }
    else{
        cout << "nie jest przestepny" << endl;
    }
}
```

```
In [ ]: // 02.cpp

#include <iostream>

using namespace std;

void zad02(){
    int l1, l2;
    int sum = 0;
    cout << "podaj dwie liczby: ";
    cin >> l1 >> l2;

    if (l1 == l2){
        exit(0);
    }
    else if (l1 > l2){
        int tmp = l1;
        l1 = l2;
        l2 = tmp;
    }
    l1++;
    for (; l1 < l2; l1++){
        if (l1 % 2 == 0){
            sum += l1;
        }
        cout << l1 << endl;
    }

    cout << "suma wynosi " << sum << endl;
}
```

In []: // 03.cpp

```
#include <iostream>

using namespace std;

void zad03(){
    int dt;

    cout << "podaj numer dnia tygodnia <1, 7> : ";
    cin >> dt;

    if ((dt < 1) || (dt > 7)){
        cout << "nieprawidlowy numer" << endl;
        exit(1);
    }

    string dni_t[7] = {"poniedzialek", "wtorek", "sroda", "czwartek", "piatek",
    cout << "podany numer to " << dni_t[dt - 1] << endl;

}
```

In []: // 04.cpp

```
#include <iostream>

using namespace std;

void zad04() {
    int liczby[5];

    for (int i = 0; i < 5; i++){
        cout << "podaj " << i + 1 << " liczbe: ";
        cin >> liczby[i];
    }
    int min = liczby[0];
    int max = liczby[0];
    for (int i = 1; i < 5; i++){
        if (liczby[i] > max){
            max = liczby[i];
        }
        else if (liczby[i] < min){
            min = liczby[i];
        }
    }

    cout << "min to " << min << " max to " << max << " a ich roznica to " << (ma
}
```

In []: // 05.cpp

```
#include <iostream>

using namespace std;

void zad05(){
    int n;
```

```

cout << "podaj ilosc liczb: ";
cin >> n;

int tab[n];

for (int i = 0; i < n; i++){
    cout << "podaj " << i +1 << " liczbe: ";
    cin >> tab[i];
}

int sum = 0;

for (int i = 0; i < n; i++){
    sum += tab[i];
}
int s = sum / n;
cout << "srednia wynosi " << s << endl;

for (int i = 0; i < s; i++){
    cout << "0";
}
cout << endl;

}

```

In []: `// funkcje.h`

```

#pragma once

void zad01();

void zad02();

void zad03();

void zad04();

void zad05();

```

In []: `#include "funkcje.h"`

```

#include <iostream>

using namespace std;

int main(void){
    int program;
    cout << "podaj program od <1, 5> : ";
    cin >> program;
    switch (program)
    {
        case 1:
            zad01();
            break;
        case 2:
            zad02();
            break;
        case 3:
            zad03();
            break;
    }
}

```

```

        case 4:
            zad04();
            break;
        case 5:
            zad05();
            break;
        default:
            cout << "nieprawidlowy numer" << endl;
            return 1;
    }
    return 0;
}

```

```

patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$ g++ z02.cpp 01.cpp 02.cpp 03.cpp 04.cpp 05.cpp -o zadanie2
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$ ./zadanie2
podaj program od <1, 5> : 2
podaj dwie liczby: 2 3
suma wynosi 0
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$ ./zadanie2
podaj program od <1, 5> : 6
nieprawidlowy numer
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$ ./zadanie2
podaj program od <1, 5> : 1
podaj rok: 2024
jest przestepny
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$ ./zadanie2
podaj program od <1, 5> : 5
podaj ilosc liczb: 4
podaj 1 liczbe: 1
podaj 2 liczbe: 2
podaj 3 liczbe: 3
podaj 4 liczbe: 4
srednia wynosi 2
00
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/02$

```

Zadanie 3

```

In [ ]: #include <iostream>
#include <stdlib.h>

using namespace std;

typedef struct {
    int x;
    int y;
} position;

void get_final_position(position &pos, int n,int (&ruchy)[4]){
    for (int i = 0; i < n; i++){
        switch (i % 4)
        {
            case 0:
                pos.y += ruchy[i];
                cout << ruchy[i] << " kroków na polnoc, teraz o (" << pos.x << " , "
                break;
            case 1:
                pos.x += ruchy[i];
                cout << ruchy[i] << " kroków na wschod, teraz o (" << pos.x << " , "
                break;
            case 2:
                pos.y -= ruchy[i];
                cout << ruchy[i] << " kroków na poludnie, teraz o (" << pos.x << " , "
                break;
            case 3:

```

```

        pos.x -= ruchy[i];
        cout << ruchy[i] << " kroków na zachod, teraz o (" << pos.x << " , "
        break;
    }
}

int main(int argc, char* argv[]){
    int ruchy[argc - 1];

    int ruchy_iterator = 0;
    for (int i = 1; i < argc; i++){
        ruchy[ruchy_iterator] = strtol(argv[i], NULL, 10);
        ruchy_iterator++;
    }

    position pos = {0, 0};

    get_final_position(pos, argc - 1, ruchy);

    cout << "i skonczy na wspolrzednych (" << pos.x << " , " << pos.y << ")" <<

    return 0;
}

```

```

patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/03$ ./zadanie3 20 30 40 50 60 70
20 kroków na polnoc, teraz o (0 , 20)
30 kroków na wschod, teraz o (30 , 20)
40 kroków na poludnie, teraz o (30 , -20)
50 kroków na zachod, teraz o (-20 , -20)
60 kroków na polnoc, teraz o (-20 , 40)
70 kroków na wschod, teraz o (50 , 40)
i skonczy na wspolrzednych (50 , 40)
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/03$ ./zadanie3 20
20 kroków na polnoc, teraz o (0 , 20)
i skonczy na wspolrzednych (0 , 20)
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/03$ ./zadanie3 0 0 30 40 20
0 kroków na polnoc, teraz o (0 , 0)
0 kroków na wschod, teraz o (0 , 0)
30 kroków na poludnie, teraz o (0 , -30)
40 kroków na zachod, teraz o (-40 , -30)
20 kroków na polnoc, teraz o (-40 , -10)
i skonczy na wspolrzednych (-40 , -10)
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/03$

```

Zadanie 4

```

In [ ]: #include <iostream>
#include <cstdlib>
#include <ctime>

enum Choice { ROCK = 1, PAPER, SCISSORS };

std::string choiceToString(int choice) {
    switch (choice) {
        case ROCK: return "kamien";
        case PAPER: return "papier";
        case SCISSORS: return "nozyce";
        default: return "nieprawidlowe";
    }
}

```

```

int knp(int userChoice, int computerChoice) {
    if (userChoice == computerChoice) return 0; // Draw
    if ((userChoice == ROCK && computerChoice == SCISSORS) ||
        (userChoice == PAPER && computerChoice == ROCK) ||
        (userChoice == SCISSORS && computerChoice == PAPER)) {
        return 1; // wygrasz
    }
    return -1; // przegrywasz
}

int main() {
    std::srand(std::time(0));
    int userScore = 0, computerScore = 0;

    for (int round = 1; round <= 3; ++round) {
        int userChoice;
        std::cout << "Runda " << round << ": Wybierz (1 - kamien, 2 - papier, 3 - nozyce) ";
        std::cin >> userChoice;

        if (userChoice < 1 || userChoice > 3) {
            std::cout << "wybierz 1/2/3 \n";
            --round; // powtorz runde
            continue;
        }

        int computerChoice = std::rand() % 3 + 1;
        std::cout << "komputer : " << choiceToString(computerChoice) << "\n";

        int result = knp(userChoice, computerChoice);
        if (result == 1) {
            std::cout << "Wygrasz!\n";
            ++userScore;
        } else if (result == -1) {
            std::cout << "Przegrywasz!\n";
            ++computerScore;
        } else {
            std::cout << "Remis!\n";
        }
    }

    std::cout << "\nWYNIKI: Ty " << userScore << " - " << computerScore << " Komputer\n";
    if (userScore > computerScore) {
        std::cout << "gratki! wygrales!\n";
    } else if (userScore < computerScore) {
        std::cout << "ilosc sekund od 1970 cie pokonala :/ \n";
    } else {
        std::cout << "remisik\n";
    }

    return 0;
}

```

```

Runda 2: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 2
komputer : papier
Remis!
Runda 3: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 3
komputer : kamien
Przegrywasz!

WYNIKI: Ty 1 – 1 Komputer
remisik
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/04$ ./zadanie4
Runda 1: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 3
komputer : papier
Wygrywasz!
Runda 2: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 3
komputer : papier
Wygrywasz!
Runda 3: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 3
komputer : papier
Wygrywasz!

WYNIKI: Ty 3 – 0 Komputer
gratki! wygrales!
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/04$ ./zadanie4
Runda 1: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 1
komputer : papier
Przegrywasz!
Runda 2: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 1
komputer : kamien
Remis!
Runda 3: Wybierz (1 – kamien, 2 – papier, 3 – nozyczki): 1
komputer : kamien
Remis!

WYNIKI: Ty 0 – 1 Komputer
ilosc sekund od 1970 cie pokonala :/
patryk@DESKTOP-IRMHL7I:~/podstawy_inf_sprawozdania/sem2/04$

```

Podsumowanie i Wnioski

Repozytorium github link: https://github.com/UmarlyPoeta/pi_sem_1_bin

Zadania przedstawione w sprawozdaniu obejmują szeroki zakres problemów programistycznych, od prostych operacji arytmetycznych, po bardziej złożone algorytmy związane z analizą danych i tworzeniem funkcji. Każde zadanie miało na celu rozwinięcie konkretnych umiejętności, takich jak efektywne zarządzanie danymi wejściowymi, obliczenia matematyczne, a także praca z pętlami i warunkami logicznymi.

Pierwsze zadanie, dotyczące liczenia nóg zwierząt, pozwala na ćwiczenie umiejętności pracy z funkcjami i arytmetyką w kontekście rzeczywistego problemu. Zadania związane z obliczaniem lat przestępnych oraz analizą przedziałów liczb uczą stosowania instrukcji warunkowych i pętli, co jest podstawą wielu algorytmów. Natomiast stworzenie gry „kamień, papier, nozyczki” wymagało użycia funkcji losujących, co dodatkowo rozwija umiejętności związane z interakcją komputera z użytkownikiem.

Podsumowując, wszystkie zadania były wartościowe pod względem nauki kluczowych elementów programowania, takich jak logika, struktury danych oraz interakcja z użytkownikiem. Rozwiązywanie ich pozwoliło na wzmocnienie umiejętności praktycznych i teoretycznych, niezbędnych w pracy nad bardziej zaawansowanymi projektami programistycznymi.