



## Sprawozdanie zaliczeniowe z ćwiczeń laboratoryjnych Podstawy Informatyki

Maciej Waloszczyk  
Informatyka Techniczna (WIMiIP)

14 listopada 2024

### Wstęp

Wszystkie wymagane pliki są dostępne na platformie GitHub pod adresem:  
[https://github.com/maciejwaloszczyk/podstawy\\_informatyki](https://github.com/maciejwaloszczyk/podstawy_informatyki)

Zadania zostały wykonane na platformie:

**System:** MacOS Sequoia 15.1

**Mikroprocesor:** Apple M3 (ARM-64)

**Pamięć:** 16 GB

**Narzędzia:** Visual Studio Code, Draw.io, Xcode

**Kompilator:** Clang++ 16

**Język:** C++

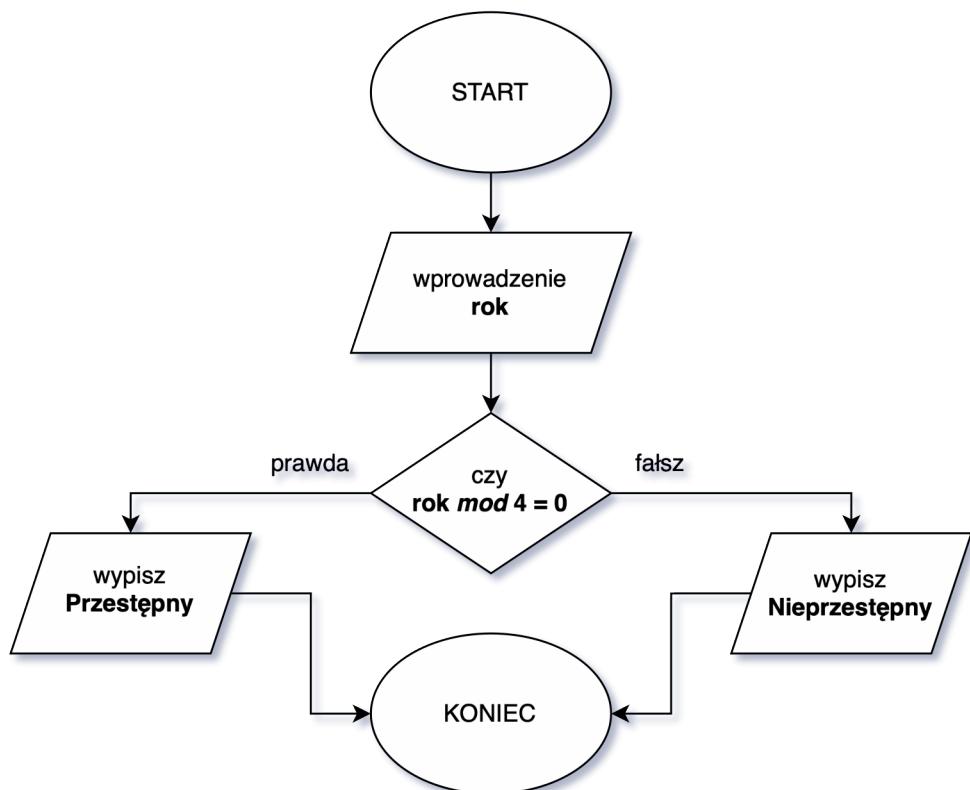
**Wykresy:** LaTeX

# 1 Instrukcje warunkowe, pętle, schematy blokowe

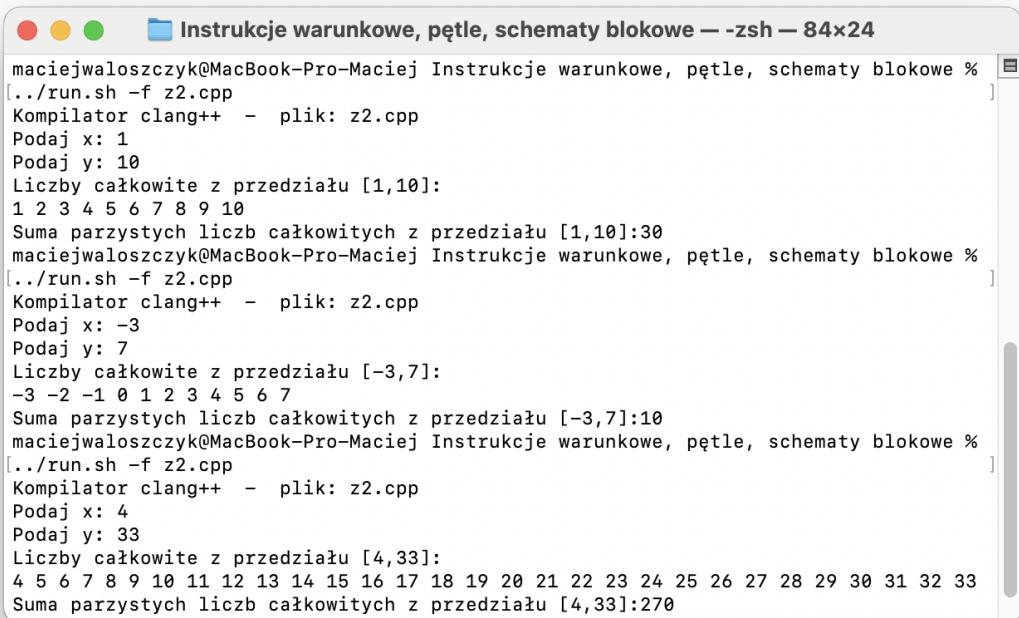
## Zadanie 1

```
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe -- zsh -- 80x24
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe] e % ./run.sh -f z1.cpp
Kompilator clang++ - plik: z1.cpp
Podaj rok: 2024
Przestępny.
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe] e % ./run.sh -f z1.cpp
Kompilator clang++ - plik: z1.cpp
Podaj rok: 2023
Nieprzestępny.
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe] e % ./run.sh -f z1.cpp
Kompilator clang++ - plik: z1.cpp
Podaj rok: 2019
Nieprzestępny.
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe] e % ./run.sh -f z1.cpp
Kompilator clang++ - plik: z1.cpp
Podaj rok: 2012
Przestępny.
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe
e %
```

Schemat blokowy

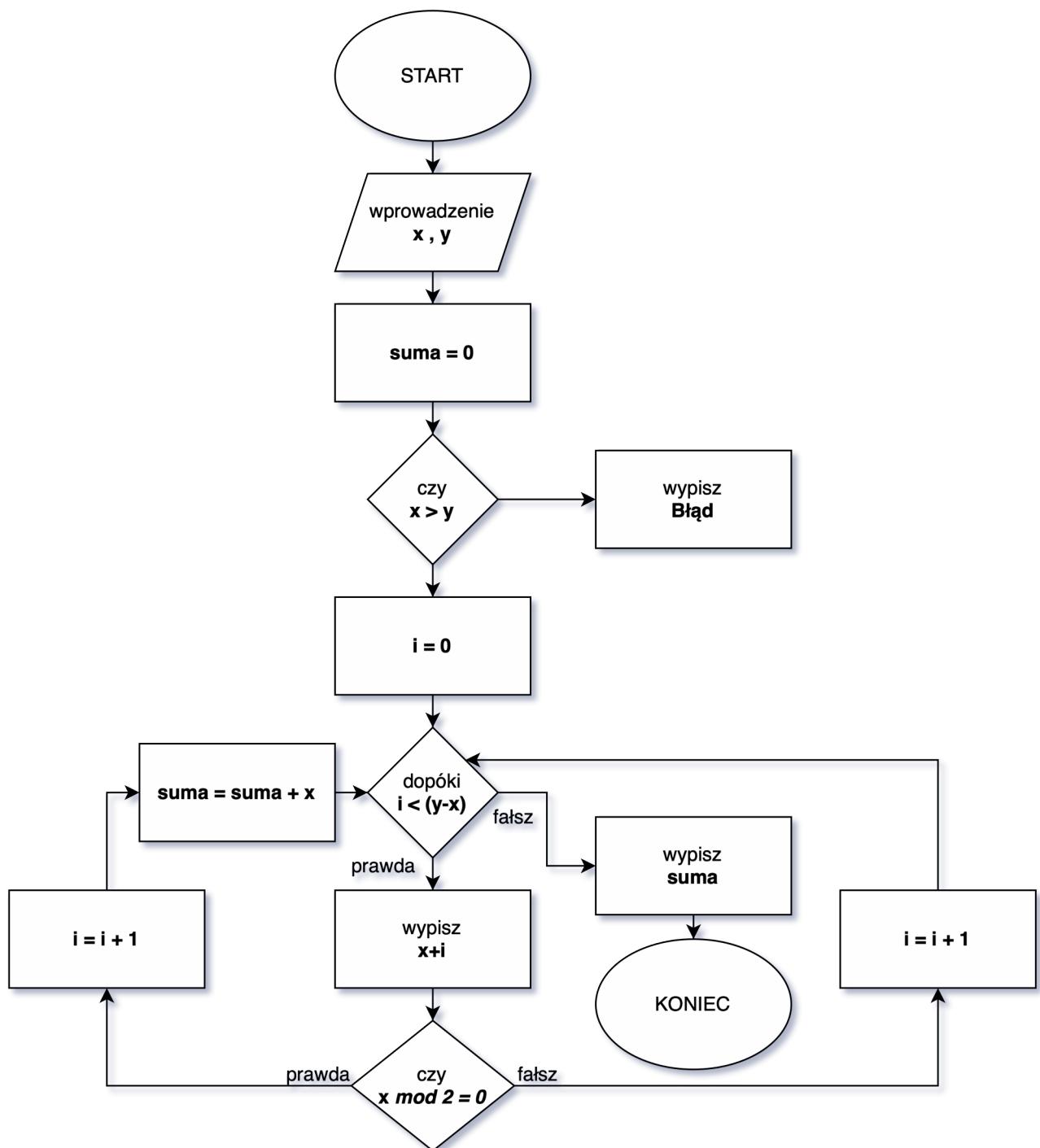


## Zadanie 2



```
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe %
[./run.sh -f z2.cpp
Komplator clang++ - plik: z2.cpp
Podaj x: 1
Podaj y: 10
Liczby całkowite z przedziału [1,10]:
1 2 3 4 5 6 7 8 9 10
Suma parzystych liczb całkowitych z przedziału [1,10]:30
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe %
[./run.sh -f z2.cpp
Komplator clang++ - plik: z2.cpp
Podaj x: -3
Podaj y: 7
Liczby całkowite z przedziału [-3,7]:
-3 -2 -1 0 1 2 3 4 5 6 7
Suma parzystych liczb całkowitych z przedziału [-3,7]:10
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe %
[./run.sh -f z2.cpp
Komplator clang++ - plik: z2.cpp
Podaj x: 4
Podaj y: 33
Liczby całkowite z przedziału [4,33]:
4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
Suma parzystych liczb całkowitych z przedziału [4,33]:270
```

## Schemat blokowy do zadania 2



## Zadanie 3

```
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x24]
e % ..../run.sh -f z3.cpp
Komplator clang++ - plik: z3.cpp
Podaj 1 liczbę: 1
Podaj 2 liczbę: 3
Podaj 3 liczbę: 5
Podaj 4 liczbę: 7
Podaj 5 liczbę: 9

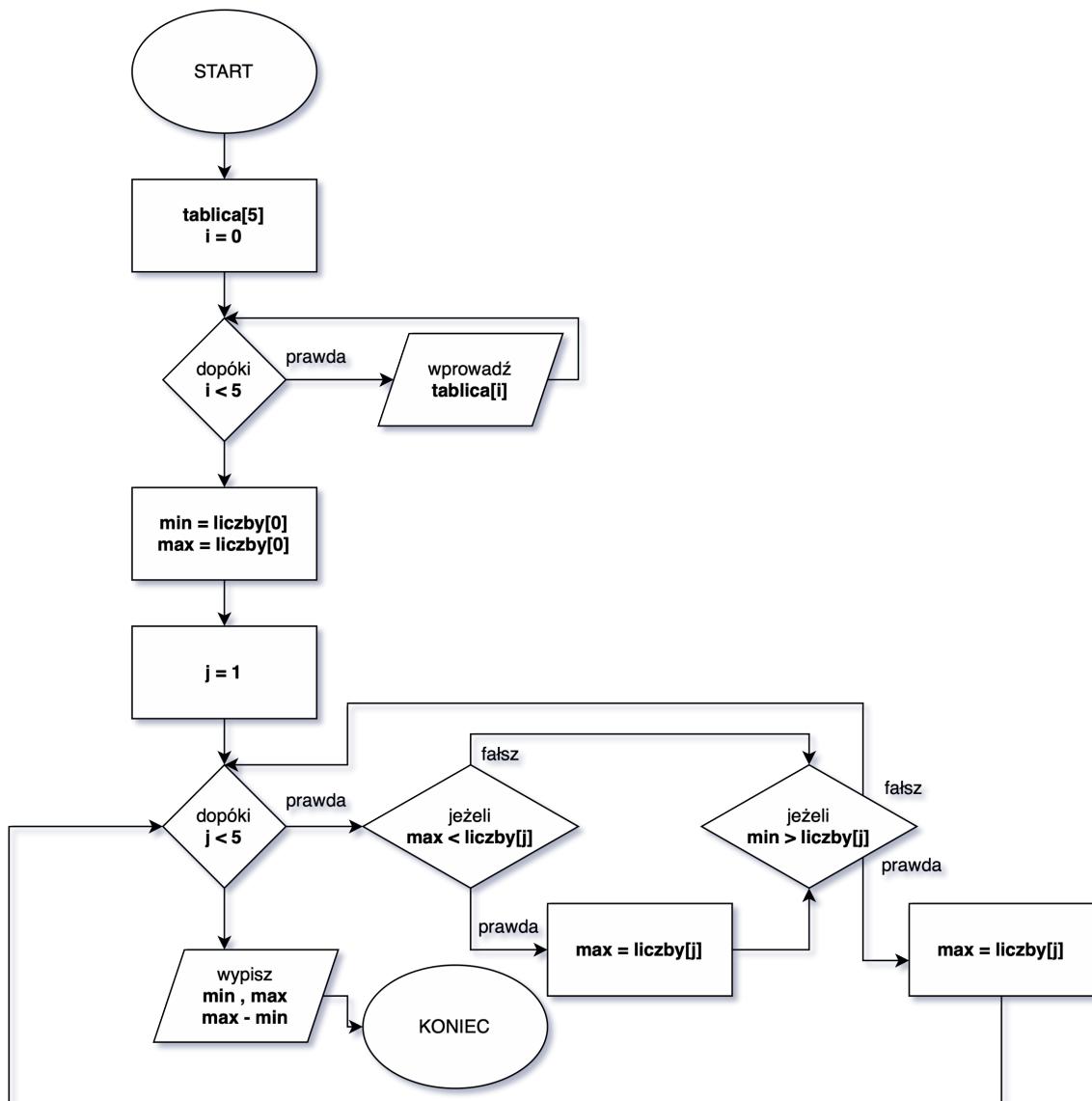
Najmniejsza liczba: 1
Największa liczba: 9
Różnica liczb: 8
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x24]
e % ..../run.sh -f z3.cpp
Komplator clang++ - plik: z3.cpp
Podaj 1 liczbę: 12
Podaj 2 liczbę: 34
Podaj 3 liczbę: 56
Podaj 4 liczbę: 78
Podaj 5 liczbę: 90

Najmniejsza liczba: 12
Największa liczba: 90
Różnica liczb: 78
```

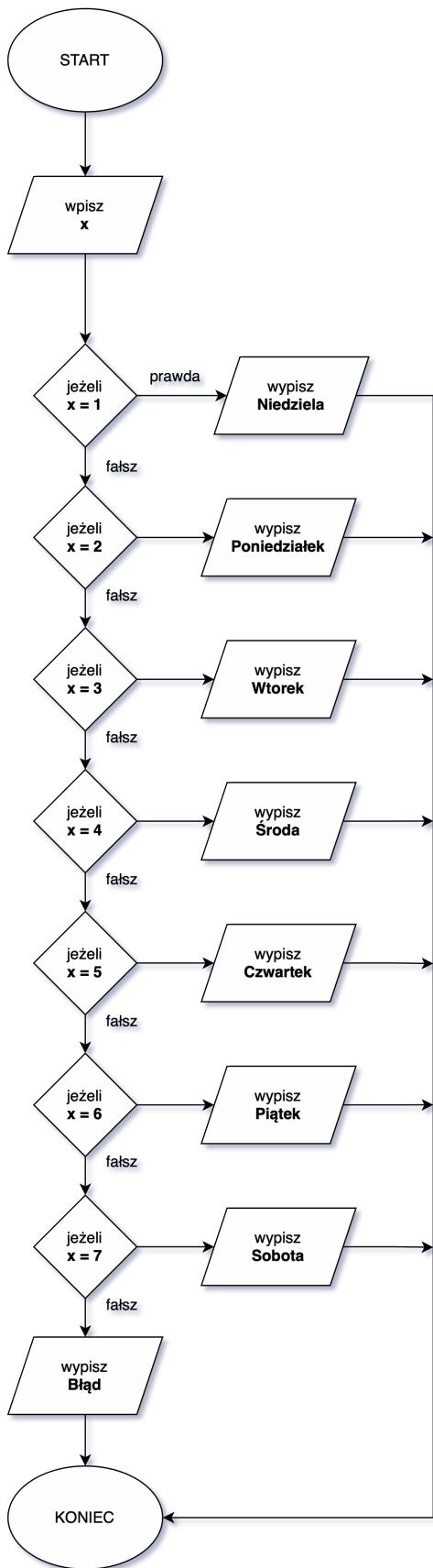
## Zadanie 4

```
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x21]
e % ..../run.sh -f z4.cpp
Komplator clang++ - plik: z4.cpp
Podaj numer dnia tygodnia: 1
Niedziela
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x21]
e % ..../run.sh -f z4.cpp
Komplator clang++ - plik: z4.cpp
Podaj numer dnia tygodnia: 3
Wtorek
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x21]
e % ..../run.sh -f z4.cpp
Komplator clang++ - plik: z4.cpp
Podaj numer dnia tygodnia: 4
Środa
[maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe --zsh-- 80x21]
e % ..../run.sh -f z4.cpp
Komplator clang++ - plik: z4.cpp
Podaj numer dnia tygodnia: 7
Sobota
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe
```

### Schemat blokowy do zadania 3



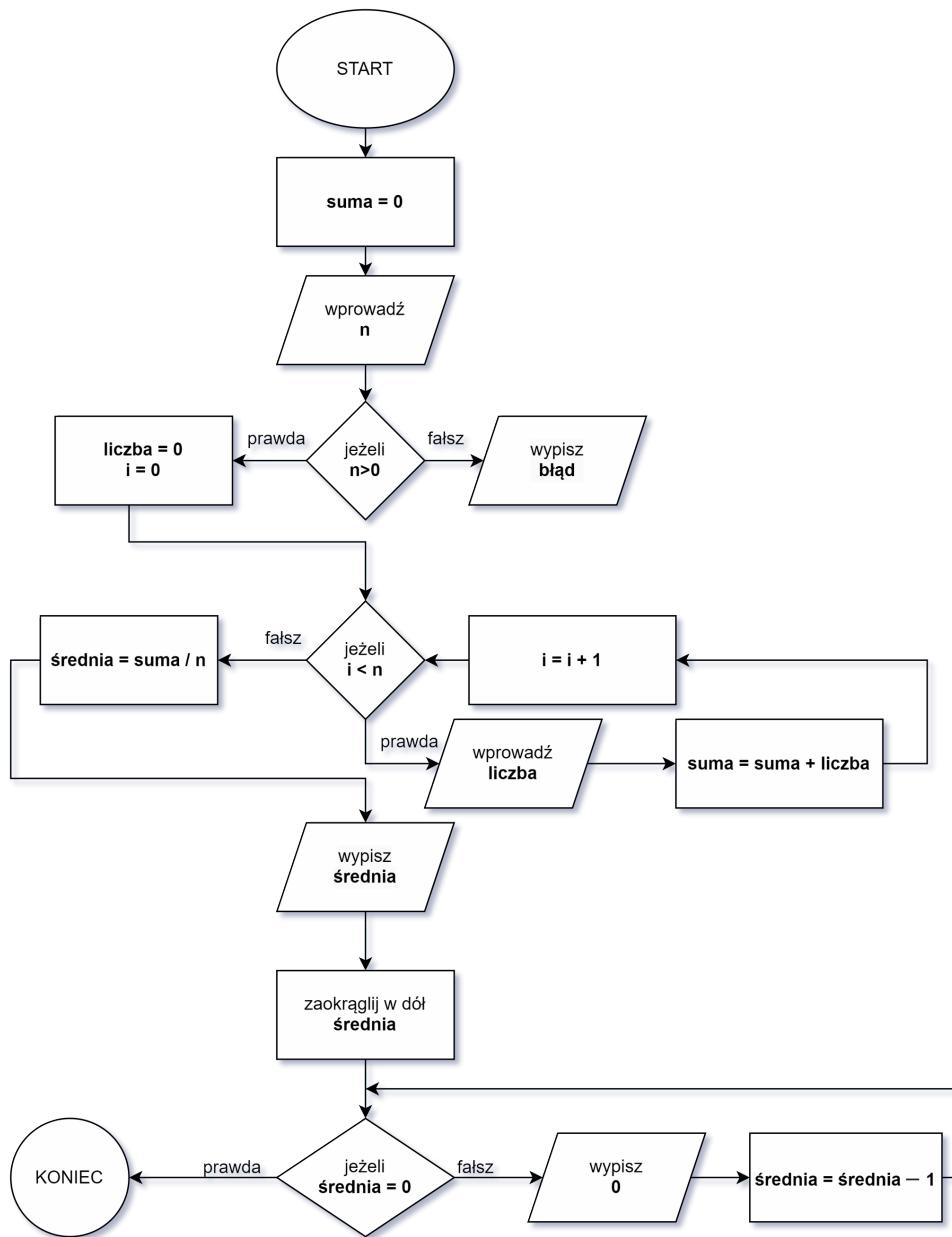
## Schemat blokowy do zadania 4



## Zadanie 5

```
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe % ./run.sh -f z5.cpp
[Kompilator clang++ - plik: z5.cpp
Podaj ile liczb chcesz wprowadzić: 6
Podaj liczbę 1: 1
Podaj liczbę 2: 3
Podaj liczbę 3: 5
Podaj liczbę 4: 7
Podaj liczbę 5: 9
Podaj liczbę 6: 11
Średnia arytmetyczna wprowadzonych liczb: 6
Liczba 0 wypisana 6 razy:
0 0 0 0 0
maciejwaloszczyk@MacBook-Pro-Maciej Instrukcje warunkowe, pętle, schematy blokowe %
```

Schemat blokowy



## 2 Funkcje

### Zadanie 1

```
Funkcje -- zsh -- 80x43
Komplator clang++ - plik: z1.cpp
Wybierz program [1-5]: 1

Podaj rok: 1013
Nieprzestepny.
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z1.cpp
Komplator clang++ - plik: z1.cpp
Wybierz program [1-5]: 2

Podaj x: 2
Podaj y: 14
Liczby calkowite z przedzialu [2,14]:
2 3 4 5 6 7 8 9 10 11 12 13 14
Suma parzystych liczb calkowitych z przedzialu [2,14]:56
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z1.cpp
Komplator clang++ - plik: z1.cpp
Wybierz program [1-5]: 3

Podaj 1 liczbe: 2
Podaj 2 liczbe: 4
Podaj 3 liczbe: 6
Podaj 4 liczbe: 8
Podaj 5 liczbe: 11

Najmniejsza liczba: 2
Najwiecejsza liczba: 11
Rownica liczby: 9
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z1.cpp
Komplator clang++ - plik: z1.cpp
Wybierz program [1-5]: 4

Podaj numer dnia tygodnia: 4
Sroda
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z1.cpp
Komplator clang++ - plik: z1.cpp
Wybierz program [1-5]: 5

Podaj ile liczb chcesz wprowadzic: 2
Podaj liczbe 1: 6
Podaj liczbe 2: 8
Srednia arytmetyczna wprowadzonych liczb: 7
Liczba 0 wypisana 7 razy:
0 0 0 0 0 0 0
```

### Zadanie 2

```
Funkcje -- zsh -- 80x16
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z2.cpp
Komplator clang++ - plik: z2.cpp
Wpisz pierwszy ruch: 20
Wpisz drugi ruch: 30
Wpisz trzeci ruch: 10
Wpisz czwarty ruch: 40

Robocik poruszył sie na pole x,y = {-10,10}
|maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ../run.sh -f z2.cpp
Komplator clang++ - plik: z2.cpp
Wpisz pierwszy ruch: 5
Wpisz drugi ruch: 7
Wpisz trzeci ruch: 12
Wpisz czwarty ruch: 21

Robocik poruszył sie na pole x,y = {-14,-7}
```

## Zadanie 3

```
Funkcje -- zsh -- 80x24
Kompilator clang++ - plik: z3.cpp
Wpisz pierwszy wyraz ciągu (a1): 2
Wpisz różnicę ciągu (r): 6

Ciąg an:
Wyraz 1: 2; Wyraz 2: 8; Wyraz 3: 14; Wyraz 4: 20; Wyraz 5: 26; Wyraz 6: 32; Wyra
z 7: 38; Wyraz 8: 44; Wyraz 9: 50; Wyraz 10: 56; Wyraz 11: 62; Wyraz 12: 68; Wyr
az 13: 74; Wyraz 14: 80; Wyraz 15: 86; Wyraz 16: 92; Wyraz 17: 98; Wyraz 18: 104
; Wyraz 19: 110; Wyraz 20: 116; Wyraz 21: 122; Wyraz 22: 128; Wyraz 23: 134; Wyr
az 24: 140; Wyraz 25: 146; Wyraz 26: 152; Wyraz 27: 158; Wyraz 28: 164; Wyraz 29
: 170; Wyraz 30: 176; Wyraz 31: 182; Wyraz 32: 188; Wyraz 33: 194; Wyraz 34: 200
; Wyraz 35: 206; Wyraz 36: 212; Wyraz 37: 218; Wyraz 38: 224; Wyraz 39: 230; Wyr
az 40: 236; Wyraz 41: 242; Wyraz 42: 248; Wyraz 43: 254; Wyraz 44: 260; Wyraz 45
: 266; Wyraz 46: 272; Wyraz 47: 278; Wyraz 48: 284; Wyraz 49: 290; Wyraz 50: 296
; Wyraz 51: 302; Wyraz 52: 308; Wyraz 53: 314; Wyraz 54: 320; Wyraz 55: 326; Wyr
az 56: 332; Wyraz 57: 338; Wyraz 58: 344; Wyraz 59: 350; Wyraz 60: 356; Wyraz 61
: 362; Wyraz 62: 368; Wyraz 63: 374; Wyraz 64: 380; Wyraz 65: 386; Wyraz 66: 392
; Wyraz 67: 398; Wyraz 68: 404; Wyraz 69: 410; Wyraz 70: 416; Wyraz 71: 422; Wyr
az 72: 428; Wyraz 73: 434; Wyraz 74: 440; Wyraz 75: 446; Wyraz 76: 452; Wyraz 77
: 458; Wyraz 78: 464; Wyraz 79: 470; Wyraz 80: 476; Wyraz 81: 482; Wyraz 82: 488
; Wyraz 83: 494; Wyraz 84: 500; Wyraz 85: 506; Wyraz 86: 512; Wyraz 87: 518; Wyr
az 88: 524; Wyraz 89: 530; Wyraz 90: 536; Wyraz 91: 542; Wyraz 92: 548; Wyraz 93
: 554; Wyraz 94: 560; Wyraz 95: 566; Wyraz 96: 572; Wyraz 97: 578; Wyraz 98: 584
; Wyraz 99: 590; Wyraz 100: 596;
```

## Zadanie 4

```
Funkcje -- zsh -- 80x28
[maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ..../run.sh -f z4.cpp
Kompilator clang++ - plik: z4.cpp
Zgadnij liczbę od 1 do 100!
Podaj liczbę: 5
ZA MAŁO!
Podaj liczbę: 50
ZA MAŁO!
Podaj liczbę: 70
ZA MAŁO!
Podaj liczbę: 90
ZA DUŻO!
Podaj liczbę: 85
Gratulacje! Zgadłeś liczbę w 5 próbach.
[maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ..../run.sh -f z4.cpp
Kompilator clang++ - plik: z4.cpp
Zgadnij liczbę od 1 do 100!
Podaj liczbę: 10
ZA MAŁO!
Podaj liczbę: 40
ZA DUŻO!
Podaj liczbę: 20
ZA MAŁO!
Podaj liczbę: 30
ZA MAŁO!
Podaj liczbę: 35
ZA MAŁO!
Podaj liczbę: 39
Gratulacje! Zgadłeś liczbę w 6 próbach.
```

## Zadanie 5

```
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z5.cpp
Kompilator clang++ - plik: z5.cpp
Totolotek:
Liczba nr 1: 42
Liczba nr 2: 18
Liczba nr 3: 19
Liczba nr 4: 12
Liczba nr 5: 41
Liczba nr 6: 6
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z5.cpp
Kompilator clang++ - plik: z5.cpp
Totolotek:
Liczba nr 1: 42
Liczba nr 2: 48
Liczba nr 3: 32
Liczba nr 4: 20
Liczba nr 5: 32
```

```
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z5.cpp
Kompilator clang++ - plik: z5.cpp
Totolotek:
Liczba nr 1: 42
Liczba nr 2: 18
Liczba nr 3: 19
Liczba nr 4: 12
Liczba nr 5: 41
Liczba nr 6: 6
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z5.cpp
Kompilator clang++ - plik: z5.cpp
Totolotek:
Liczba nr 1: 42
Liczba nr 2: 48
Liczba nr 3: 32
Liczba nr 4: 20
Liczba nr 5: 32
Liczba nr 6: 22
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z5.cpp
Kompilator clang++ - plik: z5.cpp
Totolotek:
Liczba nr 1: 42
Liczba nr 2: 5
```

## Zadanie 6

```
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z6.cpp
Kompilator clang++ - plik: z6.cpp
Wpisz liczbę Pa: 1234

Konwerter:
[1] Pa => bar:
[2] Pa => tor:
[3] Pa => psi:

Opcja >> 1

1234[Pa] = 0.01234[bar]
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z6.cpp
Kompilator clang++ - plik: z6.cpp
Wpisz liczbę Pa: 349

Konwerter:
[1] Pa => bar:
[2] Pa => tor:
[3] Pa => psi:

Opcja >> 2

349[Pa] = 2.6175[tor]
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z6.cpp
Kompilator clang++ - plik: z6.cpp
Wpisz liczbę Pa: 999

Konwerter:
[1] Pa => bar:
[2] Pa => tor:
[3] Pa => psi:

Opcja >> 3

999[Pa] = 0.144893[psi]
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje %
```

## Zadanie 7

```
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z7.cpp
Kompilator clang++ - plik: z7.cpp
Wpisz liczbę a: 1
Wpisz liczbę b: 2
Wpisz liczbę c: 3

0
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z7.cpp
Kompilator clang++ - plik: z7.cpp
Wpisz liczbę a: 3
Wpisz liczbę b: 4
Wpisz liczbę c: 5

1
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje % ./run.sh -f z7.cpp
Kompilator clang++ - plik: z7.cpp
Wpisz liczbę a: 5
Wpisz liczbę b: 6
Wpisz liczbę c: 7

0
maciejwaloszczyk@MacBook-Pro-Maciej Funkcje %
```

### 3 Logika

Zadanie 1

Podpunkt A

Wyrażenie  $(p \vee q \Rightarrow r) \Rightarrow (p \Rightarrow r) \vee (q \Rightarrow r)$  jest tautologią.

Podpunkt B

Wyrażenie  $(p \vee q) \wedge (p \Rightarrow q) \Rightarrow q \Rightarrow p$  **nie** jest tautologią.

Kontrprzykład:  $p = 0$  i  $q = 1$ .

Podpunkt C

Wyrażenie  $\neg(p \Rightarrow q) \wedge (q \Rightarrow p) \Rightarrow (p \wedge \neg q)$  jest tautologią.

Zadanie 2

```
maciejwaloszczyk@MacBook-Pro-Maciej Logika % ./run.sh -f z2.cpp
Kompilator clang++ - plik: z2.cpp
0
0
1
1
1
maciejwaloszczyk@MacBook-Pro-Maciej Logika % ./run.sh -f z3.cpp
Kompilator clang++ - plik: z3.cpp
a) p ∧ q: 0
b) p ∨ q: 1
c) ¬p ∧ (p ∨ q): 0
d) (p ∧ r) → q: 0
e) ¬(p ∨ (q ∨ r)): 0
f) [(p → r) ∨ ¬q] → [p → (r ∧ ¬q)]: 1
g) [¬(¬r ∨ q) ∨ ¬(q ∧ r)] → [¬(q → p)]: 0
maciejwaloszczyk@MacBook-Pro-Maciej Logika %
```

Zadanie 3

Warunki:  $p = 1, q = 1, r = 0$ .

Zrzut ekranu prezentujący działanie programu razem z zadaniem 2.

Podpunkt A

$p \wedge q$	$p \vee q$
1	prawda

Podpunkt B

$p \vee q$	$p \vee q$
1	prawda

Podpunkt C

$\neg p$	$p \vee q$	$\neg p \wedge (p \vee q)$
0	1	fałsz

Podpunkt D

$p \wedge r$	$(p \wedge r) \Rightarrow q$
0	prawda

Podpunkt E

$q \vee r$	$p \Leftrightarrow (q \vee r)$	$\neg(p \Leftrightarrow (q \vee r))$
1	1	fałsz

Podpunkt F

$p \Rightarrow r$	$(p \Rightarrow r) \vee \neg q$	$r \wedge \neg q$	$p \Rightarrow (r \wedge \neg q)$	$[(p \Rightarrow r) \vee \neg q] \Leftrightarrow [p \Rightarrow (r \wedge \neg q)]$
0	0	0	0	prawda

Podpunkt G

$\neg r \vee q$	$\neg(q \wedge r)$	$(\neg r \vee q) \vee \neg(q \wedge r)$	$\neg(q \Rightarrow p)$	$[(\neg r \vee q) \vee \neg(q \wedge r)] \Rightarrow [\neg(q \Rightarrow p)]$
1	1	1	0	fałsz

## 4 Struktury danych – Grafy

Podpunkt A

Liczba wierzchołków: 6 (od  $v_0$  do  $v_5$ ), Liczba krawędzi: 8 (od  $e_0$  do  $e_7$ )

Pary wierzchołków, gdzie  $a$  i  $b$  łączy  $e$ :

$$(a, b) \in \{(v_0, v_1); (v_1, v_2); (v_2, v_2); (v_1, v_3); (v_3, v_1); (v_2, v_4); (v_4, v_0); (v_4, v_3)\};$$

Podpunkt B

```
maciejwaloszczyk@MacBook-Pro-Maciej Struktury danych – Grafy -- zsh -- 84x21
[maciejwaloszczyk@MacBook-Pro-Maciej Struktury danych – Grafy % ./run.sh -f z2.cpp ]
Komplilator clang++ – plik: z2.cpp
Podaj liczbe wierzcholkow: 6
Podaj liczbe krawedzi: 8
Podaj pary wierzcholokow (u v) dla kazdej krawedzi:
0 1
1 2
2 2
3 1
1 3
2 4
4 0
4 3
Macierz sasiedztwa:
0 1 0 0 0 0
0 0 1 1 0 0
0 0 1 0 1 0
0 1 0 0 0 0
1 0 0 1 0 0
0 0 0 0 0 0
maciejwaloszczyk@MacBook-Pro-Maciej Struktury danych – Grafy %
```

### Podpunkt C

```
1 (...)  
2     macierzSasiedztwa[u][v] = 1;  
3     macierzSasiedztwa[v][u] = 1;  
4 (...)
```

Wystarczy dodać linijkę,  
aby  $macierzSasiedztwa[i][j] = macierzSasiedztwa[j][i] = 1$ .

## 5 Systemy i reprezentacja liczb

### Zadanie 1

$$(113)_{10} = (01110001)_2$$

$$(432)_{10} = (000110110000)_2$$

$$(555)_{10} = (22B)_{16}$$

$$(8736)_{10} = (2220)_{16}$$

$$(10011)_2 = (2220)_{10}$$

$$(101001011)_{10} = (331)_{10}$$

$$(D5E7)_{16} = (54759)_{10}$$

$$(F01A33)_{16} = (15735347)_{10}$$

$$(752)_8 = (1EA)_{16}$$

$$(2641)_8 = (5A1)_{16}$$

### Zadanie 2

```
maciejwaloszczyk@MacBook-Pro-Maciej Systemy i reprezentacja liczb % ./run.sh -f z2.cpp  
Kompilator clang++ - plik: z2.cpp  
a): 101100  
b): 2C  
c): 5  
d): 2C  
e): 1001101  
maciejwaloszczyk@MacBook-Pro-Maciej Systemy i reprezentacja liczb %
```

## 6 Złożoność obliczeniowa

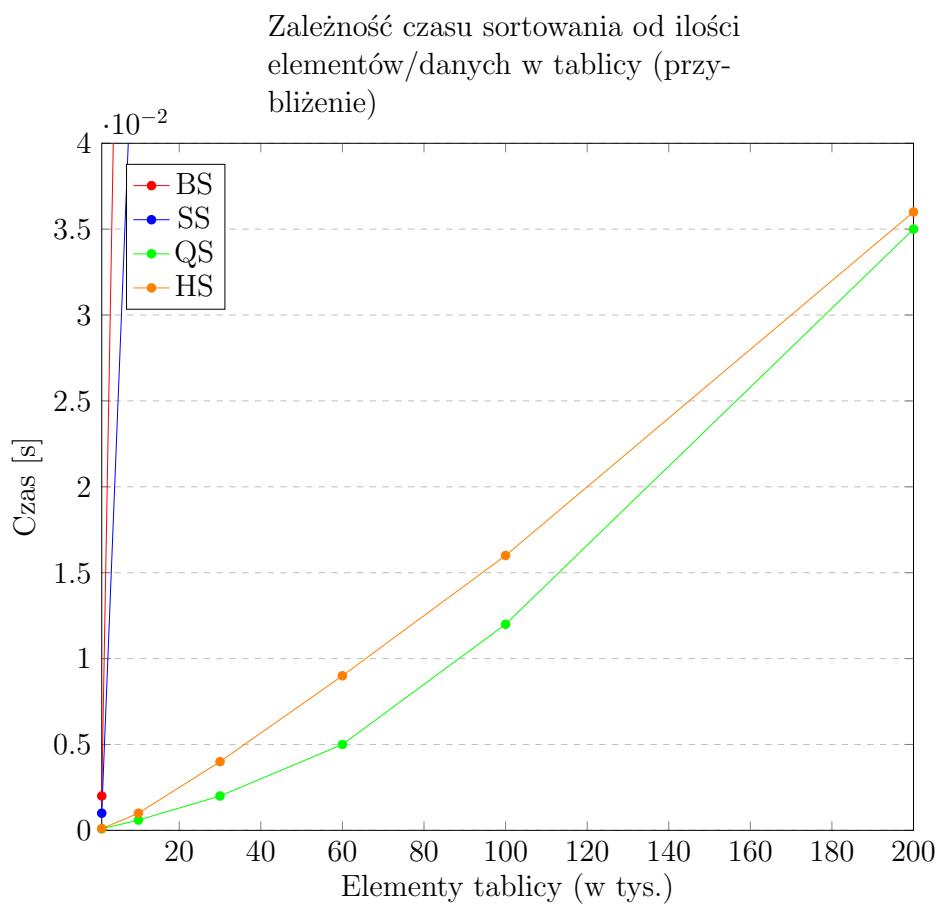
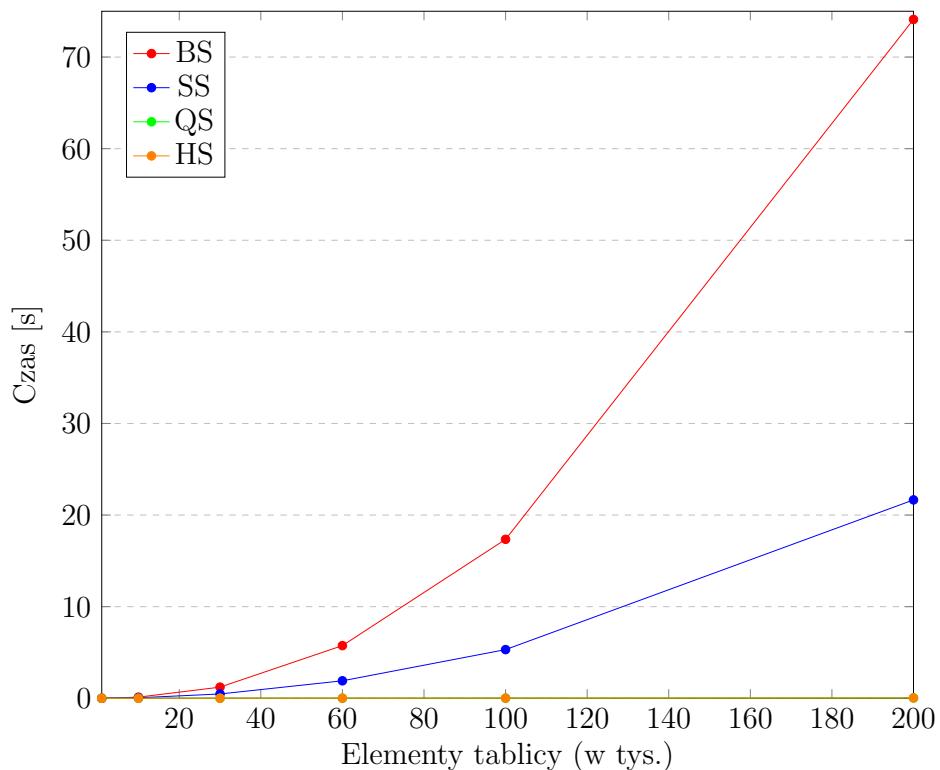
### Zadanie 1 Podpunkt A

```
Złożoność obliczeniowa -- zsh -- 88x36
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-1000.cpp]
Komplilator clang++ - plik: z1-1000.cpp
Tablica 1000 elementowa.
    Sortowanie bąbelkowe czas: 0.00185308 sekund
    Sortowanie przez wybór czas: 0.000905417 sekund
    Sortowanie szybkie czas: 8.5708e-05 sekund
    Sortowanie przez kopcowanie czas: 0.000153792 sekund
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-10000.cpp]
Komplilator clang++ - plik: z1-10000.cpp
Tablica 10000 elementowa.
    Sortowanie bąbelkowe czas: 0.122986 sekund
    Sortowanie przez wybór czas: 0.0545074 sekund
    Sortowanie szybkie czas: 0.000622458 sekund
    Sortowanie przez kopcowanie czas: 0.00125763 sekund
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-30000.cpp]
Komplilator clang++ - plik: z1-30000.cpp
Tablica 30000 elementowa.
    Sortowanie bąbelkowe czas: 1.21418 sekund
    Sortowanie przez wybór czas: 0.46918 sekund
    Sortowanie szybkie czas: 0.00220287 sekund
    Sortowanie przez kopcowanie czas: 0.00418679 sekund
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-60000.cpp]
Komplilator clang++ - plik: z1-60000.cpp
Tablica 60000 elementowa.
    Sortowanie bąbelkowe czas: 5.74856 sekund
    Sortowanie przez wybór czas: 1.90557 sekund
    Sortowanie szybkie czas: 0.00550267 sekund
    Sortowanie przez kopcowanie czas: 0.00907308 sekund
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-100000.cpp]
Komplilator clang++ - plik: z1-100000.cpp
Tablica 100000 elementowa.
    Sortowanie bąbelkowe czas: 17.3513 sekund
    Sortowanie przez wybór czas: 5.31284 sekund
    Sortowanie szybkie czas: 0.0121165 sekund
    Sortowanie przez kopcowanie czas: 0.0160198 sekund
maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa %
```

### Podpunkt B

```
Złożoność obliczeniowa -- zsh -- 87x8
[maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa % ..../run.sh -f z1-200000.cpp]
Komplilator clang++ - plik: z1-200000.cpp
Tablica 200000 elementowa.
    Sortowanie bąbelkowe czas: 74.1074 sekund
    Sortowanie przez wybór czas: 21.6538 sekund
    Sortowanie szybkie czas: 0.035048 sekund
    Sortowanie przez kopcowanie czas: 0.0358706 sekund
maciejwaloszczyk@MacBook-Pro-Maciej Złożoność obliczeniowa %
```

Podpunkt C  
Zależność czasu sortowania od ilości  
elementów/danych w tablicy



## 7 Kryptografia

### Zadanie 1

```
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia % ./run.sh -f z1.cpp
Komplikator clang++ - plik: z1.cpp
Funkcje szyfrujące:
[1] Szyfr Cezara
[2] Szyfr Morse'a

Funkcje deszyfrujące:
[3] Szyfr Cezara
[4] Szyfr Morse'a

Opcja >> 1

Podaj wyraz składający się z dużych liter: MACIEJ
Podaj klucz z przedziału [-26..26]: 7
Po zaszyfrowaniu: THJPLQ
Zapis do pliku: maciej.txt
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia % ./run.sh -f z1.cpp
Komplikator clang++ - plik: z1.cpp
Funkcje szyfrujące:
[1] Szyfr Cezara
[2] Szyfr Morse'a

Funkcje deszyfrujące:
[3] Szyfr Cezara
[4] Szyfr Morse'a

Opcja >> 3

Wpisz nazwę pliku: maciek.txt
Podaj klucz tajny z przedziału [-26..26]: 7
Pierwsza linia jako tablica znaków: THJPLQ
Po rozszyfrowaniu: MACIEJ
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia %
```

```
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia % ./run.sh -f z1.cpp
Komplikator clang++ - plik: z1.cpp
Funkcje szyfrujące:
[1] Szyfr Cezara
[2] Szyfr Morse'a

Funkcje deszyfrujące:
[3] Szyfr Cezara
[4] Szyfr Morse'a

Opcja >> 2

Podaj napis składający się z dużych liter: WALOSZCZYKPIOTR
Zaszyfrowany tekst (Morse): .-- .-.. --- ... --- -. --. --- .-. . --- - .-
Zapis do pliku: piotrek.txt
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia % ./run.sh -f z1.cpp
Komplikator clang++ - plik: z1.cpp
Funkcje szyfrujące:
[1] Szyfr Cezara
[2] Szyfr Morse'a

Funkcje deszyfrujące:
[3] Szyfr Cezara
[4] Szyfr Morse'a

Opcja >> 4

Wpisz nazwę pliku: piotrek.txt
Odszyfrowany tekst: WALOSZCZYKPIOTR
maciejwaloszczyk@MacBook-Pro-Maciej Kryptografia %
```

Zawartość pliku *maciek.txt*

1 THJPLQ

Zawartość pliku *piotrek.txt*

1 .-- .- .-.. --- ... --- -. --- .-. --- .-. . --- - .-