

# Programowanie strukturalne (2024) - Kolokwium 1 - Zestaw A42

Zasady kolokwium:

- Obowiązuje regulamin zajęć.
- Czas: 90 minut (ew. jak zostanie czasu do końca zajęć, to można zostać).
- Łącznie do zdobycia max 40 punktów. Próg zaliczenia: 25 pkt (bez innych punktów).
- **Kolokwium należy wykonać na komputerach zamontowanych na stałe w pracowniach.**
- Student przysyłając rozwiązania oświadcza, że rozwiązał je samodzielnie.
- W trakcie kolokwium nie można korzystać z żadnych materiałów pomocniczych w żadnej formie. Wszelkie kody powinny być napisane manualnie bez wspomagania się dodatkami automatycznie generującymi kod (np. Copilot, chat GPT itp.).
- Publikowanie poleceń i rozwiązań w internecie jest zabronione do czasu napisania kolokwium przez wszystkie grupy ćw.
- Kod musi się kompilować, aby był sprawdzany.
- Kod zakomentowany nie będzie sprawdzany.
- W trakcie kolokwium zostanie udostępniony przez prowadzącego pendrive. Zawartość pendrive będzie może zawierać pliki pomocnicze do poleceń. Udostępniony będzie w celu zgrania rozwiązań. Umieszczenie poleceń na pendrive powinno odbyć się w czasie kolokwium.
- Rozwiązania po czasie mogą nie być sprawdzane.
- O ile nie zaznaczono w poleceniu inaczej, każdą z funkcji należy wywołać co najmniej jeden raz (może być bardzo trywialnie).
- Należy przestrzegać nazw funkcji i kolejności argumentów w poleceniach.
- Warto zwracać uwagę na typ zwracany funkcji — jeśli metoda ma „coś” zwrócić, będzie to wskazane w poleceniu.
- Po kartkach z poleceniami można pisać i traktować jako brudnopis.
- Format rozwiązania:
  - każde rozwiązania w osobnym pliku z rozszerzeniem.
  - nazwa plików: zad1.c, zad2.c, zad3.c, zad4.c, zad5.c.
  - Zadania powinny być umieszczone w archiwum .zip na udostępnionym pendrive.
  - Nazwa archiwum powinna być wg schematu NUMERZESTAWU\_NUMERALBUMU.zip gdzie numer zestawu znajduje się na górze kartki z poleceniami. np. A23\_123456.zip

1. W folderze DebugXY (XY - losowe znaki) na pendrive znajduje się projekt z kodem w języku C. W pliku main.c w niektórych liniach są komentarze. Twoim zadaniem jest wpisanie wartości odpowiednich zmiennych po wykonaniu konkretnej linii kodu. Dopisanie nowych linii czy zaburzenie struktury kodu oznacza zero punktów za polecenie.

*Punktacja: 5 pkt.*

2. Napisz program, który wczytuje ze standardowego wejścia dodatnią liczbę całkowitą  $n$  i wypisuje na standardowym wyjściu wartość sumy odwrotności wszystkich liczb naturalnych od 1 do  $n$ , tj.  $\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n}$ .

Przypadki testowe:

wejście	wyjście
1	1
10	2.928968

*Punktacja: 7 pkt.*

3. Napisz funkcję `sqValue`, która przyjmuje jako argument wskaźnik na zmienną typu `int`. Funkcja powinna obliczyć kwadrat zmiennej wskazywanej przez wskaźnik i zaktualizować wartość wskazywaną przez wskaźnik. Stwórz przypadek testowy w main tak, aby wyświetlić na konsoli wartość zaktualizowanej zmiennej.

Przykładowa sytuacja przed wywołaniem funkcji:

Nazwa zmiennej	Wartość zmiennej	Adres zmiennej
a	-4	0x00204

Po funkcji:

Nazwa zmiennej	Wartość zmiennej	Adres zmiennej
a	16	0x00204

*Punktacja: 8 pkt.*

4. Napisz rekurencyjną funkcję `sequence`, która dla otrzymanej w argumencie dodatniej liczby całkowitej  $n$  zwraca wartość elementu o indeksie  $n$  ciągu zdefiniowanego w następujący sposób:

$$\begin{aligned}
 e_1 &= -1 \\
 e_n &= 4 \cdot e_{n-1} + 3, n\text{-parzyste}, \\
 e_n &= e_{n-1} - (-1)^n, n\text{-nieparzyste}.
 \end{aligned}$$

Stwórz przypadek testowy dla funkcji.

*Punktacja: 10 pkt.*

5. Napisz funkcję `sumEvenIndex`, która otrzymuje dwa argumenty: dodatnią liczbę całkowitą `n` oraz `n`-elementową tablicę `tab` o elementach typu `int`. Funkcja powinna zwrócić sumę elementów na pozycjach parzystych. Stwórz przypadek testowy dla funkcji.

Przykład testowy: Dla tablicy `[3, -4, 5, 7, 3]` funkcja powinna zwrócić 11, bo na indeksach parzystych są 3, 5 i 3.

*Punktacja: 10 pkt - gdy rozwiązanie w całości oparte na wskaźnikach. 7 pkt - gdy rozwiązanie bazuje na notacji tablicowej (przez nawiasy kwadratowe). W przypadku rozwiązania mieszanego, maksymalna liczba punktów może być zmienna.*

