

Programowanie strukturalne (2024) - Przykładowe Kolokwium 1 - Zestaw S11

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 dwie liczby całkowite n i m (zakładamy, że $n < m$) i wypisuje na standardowym wyjściu iloczyn liczb nieparzystych z przedziału od $[n, m]$.

Przykładowe wejście:

3
7

Przykładowe wyjście:

105

Punktacja: 7 pkt.

3. Napisz funkcję `find_max_pointed_numbers`, która ma dwa argumenty. Pierwszym argumentem jest wskaźnik `num1` na stałą wartość typu `double`, a drugim argumentem jest stały wskaźnik `num2` na zmienną typu `double`. Funkcja `find_max_pointed_numbers` ma zwrócić liczbę zawierającą większą wartość spośród wartości wskazywanych przez pierwszy i drugi wskaźnik. Stwórz przypadek testowy dla funkcji.

Przykładowa sytuacja przed wywołaniem funkcji:

Nazwa zmiennej	Wartość zmiennej	Adres zmiennej
a	4.2	0x00204
b	-3.4	0x00338

Funkcja ma zwrócić dla powyższych danych 4.2.

Punktacja: 8 pkt.

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

$$\begin{aligned}d_0 &= 3 \\d_n &= d_{n-1} - 3, n\text{-parzyste}, \\d_n &= 2 \cdot d_{n-1} + 4, n\text{-nieparzyste}.\end{aligned}$$

Stwórz dwa przypadki testowe.

Początkowe wartości:

n	0	1	2	3	4	5
d_n	3	10	7	18	15	34

Punktacja: 10 pkt.

5. Napisz funkcję, która otrzymuje cztery argumenty: dodatnią liczbę całkowitą **n**, **n**-elementowe tablice **tab1** i **tab2** oraz **n**-elementową tablicę **tab3** o elementach typu **double**. Funkcja powinna obliczać iloczyn elementów tablic **tab1** i **tab2** o tych samych indeksach i zapisywać wyniki do tablicy **tab3**. Stwórz przypadek testowy dla funkcji.

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.

Przykładowe tabele przed wywołanie funkcji:

tab1

Indeks	0	1	2
Wartość	2.0	4.0	6.0

tab2

Indeks	0	1	2
Wartość	3.0	5.0	7.0

Przykładowe tabele po wywołaniu funkcji:

tab1

Indeks	0	1	2
Wartość	2.0	4.0	6.0

tab2

Indeks	0	1	2
Wartość	3.0	5.0	7.0

tab3

Indeks	0	1	2
Wartość	6.0	20.0	42.0

