

# Egzamin (pierwszy termin) - Programowanie strukturalne - Zestaw 2

*Zadanie 1: 10 pkt. Zadanie 2: 12 pkt. Zadanie 3: 13 pkt. Zadanie 4: 15 pkt.*

*Punktacja: 46-50 pkt - bdb(5,0); 41-45 pkt - db+(4,5); 36-40 pkt - db(4,0); 31-35 pkt - dst+(3,5); 26-30 pkt - dst(3,0); 0-25 pkt - ndst (2,0).*

1. Dane są następujące wyrazy i znaki:

```
int char char * * * ( ) napis1 napis2 , foo
```

Ułóż je we właściwej kolejności, aby otrzymać nagłówek funkcji foo, która dostaje jako argumenty dwa napisy oraz zwraca wskaźnik na `int`. Następnie dodaj dowolną implementację funkcji i stwórz dla niej przypadek testowy.

2. Napisz funkcję rekurencyjną, której argumentem jest dodania liczba całkowita  $n$  oraz  $n$ -elementowa jednowymiarowa tablica o elementach typu `int`. Funkcja ma zwrócić sumę elementów z tablicy. Stwórz przypadek testowy.
3. Napisz funkcję, której argumentem jest dwuwymiarowa tablica tablic (zawierająca zmienne typu `float`) oraz jej wymiary  $n$  i  $m$ . Funkcja ma zwrócić średnią elementów na głównej przekątnej. Stwórz przypadek testowy.
4. Napisz funkcję, która porównuje dwie listy z głową o elementach typu:

```
struct node {  
    int i;  
    struct node * next;  
};
```

i zwraca 1 jeśli ostatnie elementy na liście są równe oraz 0 w pozostałych przypadkach (także wtedy gdy któraś z list lub obie są puste). Stwórz jeden przypadek testowy.