

Egzamin (pierwszy termin) - Programowanie Strukturalne - Zestaw G004

Zadanie 1 i 2: po 10 pkt. Zadanie 3 i 4: po 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).

Zad.1. Dane są następujące wyrazy i znaki:

```
* int * fun , , ) ( ** w b n int int char
```

Ułóż je we właściwej kolejności, aby otrzymać nagłówek funkcji `fun`, która dostaje jako argumenty kolejno tablicę `tablic`, napis i liczbę całkowitą. Następnie dodaj dowolną implementację funkcji i stwórz dla niej przypadek testowy.

Zad.2. Napisz funkcję, która dostaje w argumencie dwa napisy i liczbę całkowitą `n`. Funkcja ma zwrócić 1 jeśli znak o indeksie `n` w pierwszym napisie jest wcześniej w tablicy ASCII niż znak o indeksie `n` w drugim napisie. W przeciwnym wypadku albo gdy któryś z napisów jest krótszy, to funkcja ma zwrócić 0. W zadaniu nie korzystaj z funkcji bibliotecznych. Stwórz przypadek testowy.

Zad.3. Napisz funkcję, której argumentem jest dwuwymiarowa tablica `tablic` i jej wymiary `n` i `m`. Funkcja ma zwrócić średnią elementów w kolumnie o indeksie 0. Stwórz przypadek testowy.

Zad.4. Napisz rekurencyjną funkcję, która otrzymuje jako argumenty listę z głową o elementach typu:

```
struct element {
    int z;
    struct element * next;
};
```

i zwraca liczbę elementów nieparzystych ujemnych na liście. Funkcja nierekurencyjna - 0 punktów. Stwórz jeden przypadek testowy.