## Лаб: Масиви

Тествайте решението в Judge: <https://judge.softuni.bg/Contests/3171/Additional-Exercises>.

## Ден от седмицата

Въведете **ден от седмицата** [1…7] и отпечатайте **името** на деня (на английски) или **"Invalid day!"**

### Примерен вход

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 1 | Monday |
| 2 | Wednesday |
| 10 | Invalid day! |

## Числа в обратен ред

Прочетете **n** на брой числа и ги принтирайте в **обратен ред.**

### Примерен вход

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Вход** | **Изход** |
| 3  10  20  30 | 30 20 10 | 1  10 | 10 |

### Подсказки

Първо трябва да прочетем **n** от конзолата

Graphical user interface, text, application

Description automatically generated

След това **създаваме масив** от числа с размер **n**.

Graphical user interface, text, application

Description automatically generated

Четем **n** числата чрез for-цикъл.

Graphical user interface, text, application

Description automatically generated

**Задаваме** стойност на елементите на масива.

Graphical user interface, text, application

Description automatically generated

Принтираме масива в **обратен ред**.

Graphical user interface, text, application, email

Description automatically generated

## Закръглени числа

Прочетете масив от реални числа (разделени с интервали), закръглети ги до най-близкото число и принтирайте числата.

### Примерен вход

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 0.9 1.5 2.4 2.5 3.14 | 0.9 => 1  1.5 => 2  2.4 => 2  2.5 => 3  3.14 => 3 |
| -5.01 -1.599 -2.5 -1.50 0 | -5.01 => -5  -1.599 => -2  -2.5 => -3  -1.50 => -2  0 => 0 |

## Обърнат масив от низове

Прочетете **масив от низове** (разделени с **интервал**), **обърнете** го и **принтирайте** елементите:

### Примерен вход

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| a b c d e | e d c b a |
| -1 hi ho w | w ho hi -1 |

## Сума

**Прочетете масив** от конзолата и отпечатайте **сумата** на **четните числа.**

### Примерен вход

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 1 2 3 4 5 6 | 12 |
| 3 5 7 9 | 0 |
| 2 4 6 8 10 | 30 |

### Подсказки

Първо трябва да **четем масива**.

Text

Description automatically generated

Създаваме **променлива** за **сумата.**

A picture containing logo

Description automatically generated

Създаваме **for-цикъл,** за да минем през елементите от масива.

A picture containing graphical user interface

Description automatically generated

Проверяваме дали числото на **сегашният индекс** е **четно**.

Text

Description automatically generated

Принтираме **сумата**.

A picture containing logo

Description automatically generated

## Разлика на четните и нечетните числа

Напишете програма, която **пресмята разликата** на **сумата на четните** и **сумата на нечетните числа** в **масив**.

### Примерен вход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 1 2 3 4 5 6 | 3 | Четни: 2 + 4 + 6 = 12  Нечетни: 1 + 3 + 5 = 9  Резултат: 12 – 9 = 3 |
| 3 5 7 9 | -24 | Четни: 0  Нечетни: 3 + 5 + 7 + 9 = 24  Резултат: 0 – 24 = -24 |
| 2 4 6 8 10 | 30 | Четни: 2 + 4 + 6 + 8 + 10 = 30  Нечетни: 0  Резултат: 30 – 0 = 30 |

### Подсказки

Първо трябва да **прочетем масива**.

Text

Description automatically generated

Създаваме **две променливи** за сумата на **четните** и **нечетните** числа.

A picture containing text

Description automatically generated

Минаваме през всички елементи от масива чрез **for-цикъл**

A picture containing graphical user interface

Description automatically generated

Проверяваме **сегашното число** - ако числото е **четно** го добавяме към **сумата** на **четните числа**, в противен случай го добавяме към **сумата** на **нечетните числа.**

Graphical user interface, text

Description automatically generated

**Принтираме разликата**.

A screenshot of a text message

Description automatically generated with low confidence

## Съкращаване на масив от числа

Напишете програма, която прочита масив от цели числа и ги **съкращава** чрез сумиране на съседни двойки елементи, докато се получи едно **цяло число**. Например, ако имаме 3 елемента {2, 10, 3}, ние сумираме първите две числа и вторите две числа и получаваме {2+10, 10+3} = {12, 13}, след това отново сумираме всички съседни елементи и получаваме {12+13} = {25}.

### Примерен вход

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 2 10 3 | 25 | 2 10 3 à 2+10 10+3 à 12 13 à 12 + 13 à 25 |
| 5 0 4 1 2 | 35 | 5 0 4 1 2 à 5+0 0+4 4+1 1+2 à 5 4 5 3 à 5+4 4+5 5+3 à 9 9 8 à 9+9 9+8 à 18 17 à 18+17 à 35 |
| 1 | 1 | Има само едно число |

### Подсказки

Докато имаме повече от 1 елемент в масива nums[] повтаряме следните стъпки:

* Създаваме нов масив condensed[] с размер nums.Length-1.
* Събираме числата от nums[] в condensed[]
  + condensed[i] = nums[i] + nums[i+1]
* nums[] = condensed[]

Този процес може да се види на илюстрацията:

A picture containing diagram

Description automatically generated

A picture containing graphical user interface

Description automatically generated