

# Лабораторна робота №1

Тема роботи: агрегатні типи даних Objective-C

Мета роботи: ознайомитись с типами даних NSArray, NSMutableArray

## Теоретичні відомості

Мова Objective-C містить декілька класів для зберігання й обробки великої кількості об'єктів:

- масиви — NSArray, NSMutableArray
- словники — NSDictionary, NSMutableDictionary
- множини — NSSet, NSMutableSet

Ці класи мають дві загальні особливості:

- існують у двох формах: імутабельна (не може змінюватись) та мутабельна (можуть змінюватись, наприклад, кількість елементів)
- можуть містити лише об'єкти (показчики)

Останнє означає, що неможна створити, наприклад, об'єкт NSArray, що містить змінні типу int. Для цієї мети треба використовувати спеціальний клас NSNumber, який дозволяє створювати об'єкти, що зберігають числа різних базових типів:

```
NSNumber *ten = [NSNumber numberWithInt:10];
```

```
NSNumber *falseObj = [NSNumber numberWithBool:NO];
```

Також цей клас містить методи (властивості) для отримання значень базових типів з об'єктів NSNumber:

```
[ten floatValue]; або ten.floatValue;
```

```
[falseObj intValue];
```

Сучасна версія ObjC має спеціальні скорочені варіанти для створення об'єктів NSNumber, т.зв. літерали:

```
NSNumber *ten = @10;
```

```
float f = 0.5;
```

```
NSNumber *pointFive = @(f);
```

Клас NSArray містить ряд методів для створення об'єктів цього типу. Найбільш поширений варіант:

```
NSArray *arra = [NSArray arrayWithObjects:[NSNumber numberWithInt:10],  
[NSNumber numberWithInt:11], [NSNumber numberWithInt:12], nil];
```

Можна також використовувати літерали:

```
NSArray *array = @[10, 11, 12];
```

NSArray може зберігати в одній змінній будь-які об'єкти, оскільки будь-який клас в ObjC - це потомок базового класу NSObject, а сам NSArray зберігає покажчики на ці об'єкти. Але гарною практикою є безпосереднє позначення типу елементів масиву:

```
NSArray<NSNumber*> *aaa = @[10, 11];
```

Такий підхід дозволяє писати більш зрозумілі й безпасні програми.

Для отримання елементу масиву використовується метод objectAtIndex:

```
NSNumber *firstItem = [aaa objectAtIndex:0];
```

або літеральна форма:

```
NSNumber *firstItem = aaa[0];
```

або спеціальний метод:

```
NSNumber *firstItem = aaa.firstObject;
```

Інші корисні методи:

count

arrayByAddingObject

componentsJoinedByString

indexOfObject

та інші.

Об'єкти класу NSArray неможна змінювати, наприклад додавати нові елементи, або змінювати самі елементи. Наприклад, він не має методу для втсановлення елементу з заданим індексом. Робота з мутабельними масивами виконується за допомогою класу NSMutableArray.

Створювати такі об'єкти можна як мутабельні копії існуючих об'єктів NSArray:

```
NSMutableArray *mutableArray = [aaa mutableCopy];
```

Змінити об'єкт мутабельного масиву з індексом і можна наступним чином:

```
[mutableCopy replaceObjectAtIndex:0 withObject:@20];
```

```
mutableCopy[1] = @30;
```

Також клас NSMutableArray містить інші мметоди для модифікації масивів.

## Завдання

Розробити програму, що виконує над заданим масивом вказані дії. Номер варіанту визначається наступним чином:

номер\_варіанту = номер\_по\_списку % 5 + 1

Варіант	Вид обробки	Тип елементів
1	Знайти мінімальний елемент	int
2	Сортувати в зростаючому порядку	float
3	Знайти індекс елементу з заданим значенням	Рядок символів
4	Знайти суму додатних чисел	int
5	Знайти максимальне від'ємне число	float