

Programowanie iPhone - trudne początki



Kamil Brzeziński
Monika Miśniakiewicz
Gadu-Gadu S.A.



- ☺ O co ten cały szum?
- ☺ Co to jest program dla developerów
- ☺ Zapisanie się do programu
- ☺ Problemy
- ☺ Przydatne kontakty
- ☺ iPhone SDK
- ☺ Objective-C



- ☀️ Jeśli nie wiesz co to iPhone to zwolnij miejsce komuś innemu na tej sali :)
- ☀️ iPhone to nie jest telefon
- ☀️ iPhone to urządzenie multimedialne z funkcją telefonu
- ☀️ Podobno najbardziej pożądany gadżet 2007 i 2008 roku (ponad 10mln sprzedanych urządzeń)
- ☀️ Dwie wersje: iPhone i iPhone 3G
- ☀️ oraz iPod Touch

 developer.apple.com/iphone

iPhone Developer Program

The fastest path from code to customer.

The iPhone Developer Program provides a complete and integrated process for developing applications for iPhone and iPod touch.

[Learn more ▶](#)



- ☀ SDK
- ☀ Filmy instruktażowe iTunes
- ☀ Dokumentacja
- ☀ How-To
- ☀ Przykładowe aplikacje
- ☀ Certyfikat do testów na urządzeniach
































- ☀ Kupno Mac'a
- ☀ Stworzenie Apple ID
- ☀ Rejestracja jako osoba lub firma
- ☀ Czekanie kilka dni albo dłużej :)
- ☀ Zadzwoń do Apple UK lub US
- ☀ Pobranie SDK
- ☀ Rejestracja do właściwego programu
- ☀ Czekanie od 2 tygodni do 2 miesięcy
- ☀ Zadzwoń do Apple UK lub US

Purchase your iPhone Developer Program

Please select your country to complete this transaction in your local currency. If you do not see your country listed here, please [contact us](#) for further assistance.

Standard Program

- | | | |
|--|--|---|
|  United States |  Ireland |  Schweiz |
|  Australia |  Italia |  Singapore |
|  België |  日本
Japan |  한국
South Korea |
|  Belgique |  Luxembourg |  Suisse |
|  Canada (English) |  Mexico |  Suomi |
|  Danmark |  Nederland |  Sverige |
|  Deutschland |  New Zealand |  Taiwan |
|  España |  Norge |  United Kingdom |
|  France |  Österreich |  If you do not see your country listed here, please contact us. |
|  Hong Kong |  Portugal | |



- ☀ Apple Developer Care Center US
+1 (800) 633-2152
- ☀ Apple Developer Care Center UK
+44 800 505094
+44 131 345 9940
- ☀ Apple Developer Program e-mail
devprograms@apple.com
- ☀ FAX do działu billingowego (karta kredytowa)
+1 (408) 862-7602
- ☀ FAX do przesłania dokumentów
+1 (408) 974-1053
- ☀ Pamiętajcie o numerze referencyjnym Follow-Up



☀️ Książki :-)

S. Kochan: Programming in Objective-C

E. Sadun: The iPhone Developer's Cookbook

☀️ Forum myapple.pl

☀️ Oficjalne forum dla developerów Apple

☀️ Oficjalne forum beta dla developerów iPhone

☀️ hackint0sh.org







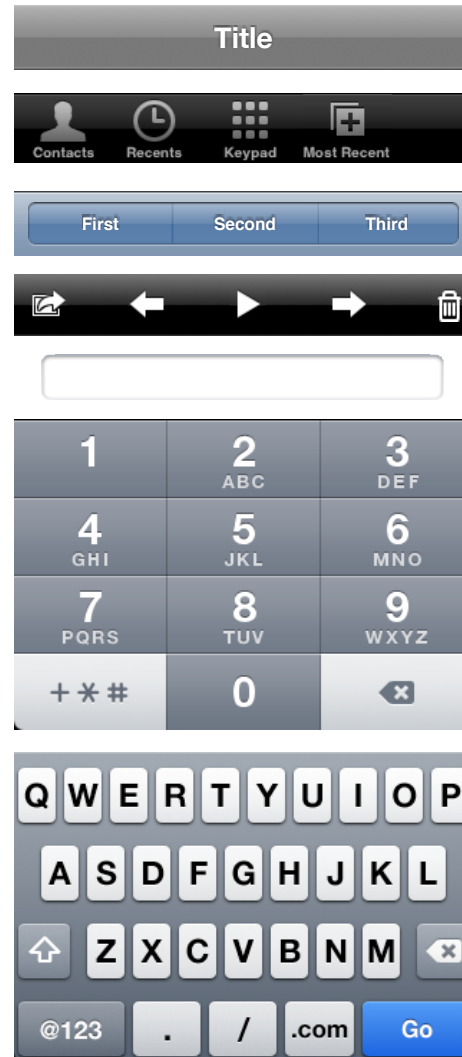


The screenshot shows the Instruments application window on a Mac OS X desktop. The Instruments window is titled "Instruments" and has a "TouchFinger" target selected. The main area displays several monitoring graphs: "Sampler", "CPU Monitor" (with sub-graphs for % System Load, % User Load, and % Total Load), "OpenCL Driver", "Leaks", and "Memory Monitor". Below the graphs is a "Memory Monitor" table with the following data:

Process ID	Process Name	User Name	% CPU	# Threads	Res Memory
0	kernel_task	root	2.2	64	299.15 MB
1	launchd	root	0	3	792.00 KB
10	kernel	root	0	2	1,000.00 KB
11	notepad	root	0	2	392.00 KB
12	xylogd	root	0	4	4.52 MB
14	update	root	0	1	200.00 KB
15	systemd	root	0	3	180.00 KB
18	securityd	root	0	1	1.33 MB
20	mds	root	0	16	42.67 MB
21	hdparm	admin	0.1	2	968.00 KB
22	loginwindow	admin	0.8	4	1.30 MB
23	kernel	root	0	2	496.00 KB
25	hold	root	0	2	372.00 KB
26	Server	root	0	23	1.17 MB

The desktop background is a blue abstract pattern. The dock at the bottom contains various application icons. A virtual iPhone simulator is overlaid on the Instruments window, displaying a space-themed game with a spaceship and a starry background.





Objective-C

- ☀️ Fundamenty Objective-C
- ☀️ Zarządzanie pamięcią
- ☀️ Krótkie porównanie C++ z Objective-C





☀️ wszystkie konstrukcje specyficzne dla Objective-C zaczynają się od małej



```
@interface
    { /* ... definicje pól ... */ }
    /* ... prototypy metod ... */
@end
```



```
@implementation
    /* ... ciała metod ... */
@end
```

C++

```
char * stringFromInt(int value);
```

Objective-C

```
- (char *) stringFromInt: (int) value;
```

Wywołanie zadeklarowanej metody

```
zmienna = [instancjaKlasy stringFromInt:514];
```

Kilka parametrów

```
/* deklaracja */  
- (void) setWidth:(int) width andHeight:(int) height;  
  
/* i wywołujemy */  
[obiekt setWidth:15 andHeight:14];
```

- ☀ sprawdzanie dostępności metody w czasie działania programu (nie podczas kompilacji)

```
if ([obiekt respondsToSelector:@selector(wyswietl:)])  
    [obiekt wyswietl:"Działa!"];
```

- ☀ możemy pominąć określenie typu zwracanego przez jakąś metodę - kompilator domyślnie przyjmie, że zwraca wartość typu id.

Deklaracja

```
@protocol JakisProtokol
    /* ... prototypy metod ... */
@end
```

Implementacja

```
@interface KlasaKtoraObslugujeJedenProtokol :
    KlasaBazowa <JakisProtokol>
@end
/*****/
@interface KlasaKtoraObslugujeKilkaProtokolow :
    KlasaBazowa <Protokol1, Protokol2, Protokol3>
@end
```

☀ sprawdzanie zgodności z protokołem
w czasie działania programu

```
if ([obiekt conformsTo:@protocol(JakisProtokol)])  
    [obiekt jakasMetodaProtokolu];
```

```
NSString* jakisString = [[NSString alloc] init];
```

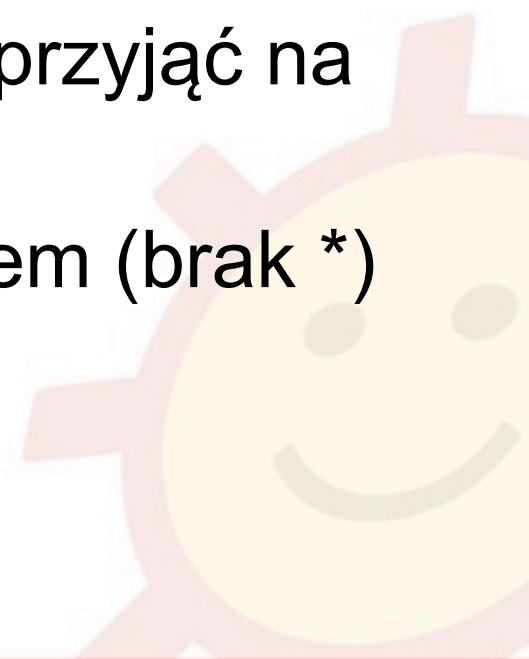
```
NSNumber* wartosc = [[NSNumber alloc] initWithFloat:1.0];
```

```
// utworzenie i zwolnienie obiektu klasy NSString  
// trzeba zwolnić jakisString jak nie będzie już potrzebny  
  
NSString* jakisString = [[NSString alloc] init];  
[jakisString release];
```

```
- (id) init {  
    if ( self = [super init] ) {  
        [self setCaption:@"Jakis tekst"];  
        [self setPhotographer:@"Jakis fotograf"];  
    }  
    return self; }  
}
```

```
- (void) dealloc {  
    [caption release];  
    [photographer release];  
    [super dealloc];  
}
```

- ☀️ Objective-C jest dynamicznie typowany
- ☀️ instancję klasy zawsze przypisujemy do wskaźnika
- ☀️ typ wbudowany id, który może przyjąć na siebie instancję dowolnej klasy
- ☀️ id samo w sobie jest wskaźnikiem (brak *)



```
@interface KlasaPochodna : KlasaBazowa
```

- ☀️ brak wielodziedziczenia
- ☀️ dziedziczymy zawsze publicznie
- ☀️ możliwa implementacja wielu protokołów



- ☀ zmienna "self" ma dokładnie takie samo działanie, jak operator "this" w C++

```
[self jakasMetoda]
```

- ☀ self jest również metodą

```
[jakisObiekt self]
```

- ☀ zmienna "super" pozwala na wywołanie metody z klasy bazowej, pełni podobną funkcję, jak kwalifikator "NazwaKlasyBazowej::" w C++, nadpisując metodę jakasMetoda i chcąc wywołać jej poprzednią wersję, piszemy:

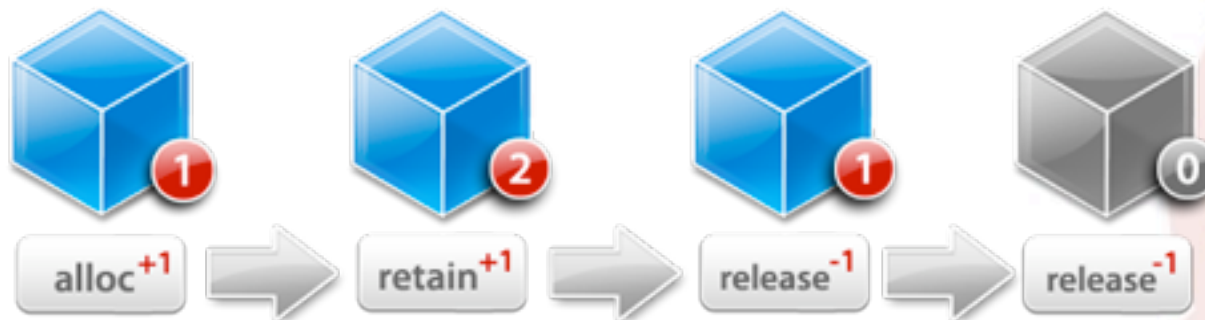
```
[super jakasMetoda]
```

```
// alokacja i inicjalizacja wartosc1
NSNumber* wartosc1 = [[NSNumber alloc] initWithFloat:8.75];

[self setTotal:wartosc1];

// autorelease automatycznie, nie trzeba zwalniać
NSNumber* wartosc2 = [NSNumber numberWithFloat:14.78];
[self setTotal:wartosc2];

// zwalniamy wartosc1
[wartosc1 release];
```



☀ trzystopniowa kontrola dostępności ale tylko dla pól klasy

```
@interface JakasKlasa
{
    /* pola zadeklarowane tu będą zastrzeżone (protected) */
    @private
    /* pola zadeklarowane tu będą prywatne (private) */
    @protected
    /* pola zadeklarowane tu będą zastrzeżone (protected) */
    @public
    /* pola zadeklarowane tu będą publiczne (public) */
}
/* prototypy metod */
@end
```



- ☀ brak przeładowywania operatorów
- ☀ brak szablonów (templates)
- ☀ wskaźniki zamiast referencji





```
#include "header.h"
```

```
#ifndef HEADER_H  
#define HEADER_H  
... zawartość pliku header.h ...  
#endif
```

```
#import "header.h"
```

lub można też

```
#include "header.h"  
i  
#pragma once
```

Pytania ?



Dziękujemy za uwagę!

