

Kolokwium MRJP 21.12.2015

Zadanie 1. Wygeneruj kod JVM (Jasmin) dla poniższej klasy w Javie:

```
public class Col {  
    public static int f(int x, int y) {  
        int z;  
        if((x<0 || y<0)&&(x>0 || y>0))  
            z = 1;  
        else  
            z = 0;  
        return z;  
    }  
}
```

Wskazówka: skoki warunkowe wykonywane gdy szczyt stosu jest (>0), (<0) to ifgt i iflt.

Zadanie 2. Przetłumacz na język assemblera x86 (32 bit), używając protokołu wywołań x86/libc następujący fragment kodu w C:

```
int f(int *x, int a);  
int g(int a, int* b) {  
    int y=1, z;  
    f(&z, f(&y, f(&z, a)));  
    return (*b)/y;  
}
```

Zadanie 3. Dla jakiego kodu w C kompilator mógł wygenerować poniższy kod LLVM?

```
define i32 @f(i32* %a, i32 %n) {  
; <label>:0  
    %1 = getelementptr i32* %a, i32 0  
    %2 = load i32* %1, align 4  
    br label %3  
; <label>:3                                ; preds = %13, %0  
    %r.0 = phi i32 [ %2, %0 ], [ %r.1, %13 ]  
    %i.0 = phi i32 [ 0, %0 ], [ %14, %13 ]  
    %4 = icmp slt i32 %i.0, %n  
    br i1 %4, label %5, label %15  
; <label>:5                                ; preds = %3  
    %6 = getelementptr i32* %a, i32 %i.0  
    %7 = load i32* %6  
    %8 = icmp sgt i32 %r.0, %7  
    br i1 %8, label %9, label %12  
; <label>:9                                ; preds = %5  
    %10 = getelementptr i32* %a, i32 %i.0  
    %11 = load i32* %10  
    br label %12  
; <label>:12                               ; preds = %9, %5  
    %r.1 = phi i32 [ %11, %9 ], [ %r.0, %5 ]  
    br label %13  
; <label>:13                               ; preds = %12  
    %14 = add i32 %i.0, 1  
    br label %3  
  
; <label>:15                                ; preds = %3  
    ret i32 %r.0  
}
```