

**1. Изучить механизм интеропа между языками, попробовать у себя вызывать C/C++ (Не C++/CLI) код (суммы чисел достаточно) из Java и C#. В отчёте описать логику работы, сложности и ограничения этих механизмов.**

C#:

Код на C++:

- sum.h:

```
#pragma once
#include <stdint>

extern "C"
{
    __declspec(dllexport) int32_t __stdcall sum(int32_t, int32_t);
}
```

- sum.cpp:

```
#include "pch.h"
#include "sum.h"

int32_t sum(int32_t a, int32_t b)
{
    return a + b;
}
```

extern "C" – для того, чтобы имя функции при линковке (и в библиотеке) совпадало с именем в программе

При использовании ключевого слова \_\_declspec(dllexport) в объектный файл добавляется директива экспорта

\_\_stdcall указывает соглашение о вызовах

Заголовочный файл cstdint используется для типов данных, фиксированных по размеру (для корректного определения их в C#).

Класс на C#:

```
using System.Runtime.InteropServices;

namespace CSInterop
{
```

```

public class MathLibraryImport
{
    [DllImport("MyMathLibrary", CallingConvention = CallingConvention.StdCall, EntryPoint = "sum")]
    public static extern int Sum(int a, int b);
}

```

Для вызова неуправляемого кода необходимо подключить System.Runtime.InteropServices

Статичной функции с произвольным именем, но таким же набором аргументов, как и у библиотечной, должен быть добавлен атрибут DllImport, в котором должно быть указано название dll-библиотеки (в данном случае “MyMathLibrary”), соглашение о вызовах (stdcall), а также, название функции в библиотеке (“sum”).

Также необходимо ключевое слово extern – у метода есть внешняя реализация.

Java:

Класс Java:

```

package mainpackage;

public class CMyClass {
    public native int sum(int a, int b);

    static
    {
        System.LoadLibrary("MyMathLibrary");
    }

    public static void main(String[] args)
    {
        int a = 10;
        int b = 20;
        int c = new CMyClass().sum(a, b);
        System.out.println(c);
    }
}

```

```
}
```

Для вызова неуправляемого кода в статическом блоке необходимо вызвать `System.LoadLibrary` и передать аргументом название библиотеки (в данном случае “`MyMathLibrary`”), а также объявить статичную функцию с такой же сигнатурой, как и у библиотечной.

Также необходимо ключевое слова `native` – у метода есть внешняя реализация.

Сгенерировать файлы C++ можно с помощью компилятора `javac`, вызвав команду `javac -h . CMyClass.java`.

Сгенерированный код на C++:

- `mainpackage_CMyClass.h`:

```
/* DO NOT EDIT THIS FILE - it is machine generated */
#include <jni.h>
/* Header for class mainpackage_CMyClass */

#ifndef _Included_mainpackage_CMyClass
#define _Included_mainpackage_CMyClass
#ifdef __cplusplus
extern "C" {
#endif
/*
 * Class:      mainpackage_CMyClass
 * Method:     sum
 * Signature:  (II)I
 */
JNIEXPORT jint JNICALL Java_mainpackage_CMyClass_sum
    (JNIEnv *, jobject, jint, jint);

#ifdef __cplusplus
}
#endif
#endif
```

- `mainpackage_CMyClass.cpp`:

```
#include "pch.h"
#include "mainpackage_CMyClass.h"
```

```
jint Java_mainpackage_CMyClass_sum (JNIEnv*, jobject, jint a, jint b) {  
    return a + b;  
}
```

Заголовочный файл jni.h используется для определённых там типов данных, необходимых Java.

Дополнительные ограничения Java для интеропа – функции в C++ должны называться особым образом, а именно, Java\_*(название пакета)*\_*(название класса)*\_*(название функции)*, причём сигнатура функции в Java должна совпадать с сигнатурой C++ (кроме названий типов).

Плюсы интеропа: достаточно просто подключить код на более быстро исполняемом языке

Минусы: несмотря на простоту, есть определённые неудобства, связанные с ограничениями:

- dll нужно собирать отдельно под каждую архитектуру

- нельзя использовать произвольный код, написанный на C++, он должен подчиняться определённым правилам (необходимо задумываться о размерах типов данных, указывать экспорт имён в стиле C и добавлять специальную директиву в объектный файл, явно указывать соглашение о вызовах)

**2. Написать немного кода на Scala и F# с использованием уникальных возможностей языка - Pipe operator, Discriminated Union, Computation expressions и т.д. . Вызвать написанный код из обычных соответствующих ООП языков (Java и C#) и посмотреть во что превращается написанный ранее код после декомпиляции в них.**

**F#:**

```
// Learn more about F# at http://docs.microsoft.com/dotnet/fsharp
namespace FS
open System
module func =
    // Define a function to construct a message to print
    let from whom =
        sprintf "from %s" whom
    type Union = X of int | Y of string
    type LoggingBuilder() =
        let log p = printfn "expression is %A" p
        member this.Bind(x, f) =
            log x
            f x
        member this.Return(x) =
            x
    let printUnion union =
        match union with
        | X i -> printfn "%i" i
        | Y s -> printfn "%s" s
        | _ -> printfn "default"

    let logSumOfSqrs a b =
        let logger = new LoggingBuilder()
        let loggedWorkflow =
            logger
            {
                let! x = a * a
                let! y = b * b
                let! z = x + y
                return z
            }
        loggedWorkflow
[<EntryPoint>]
let main argv =
```

```

let message = "F#" |> from // Call the function
printfn "Hello world %s" message
let f = Union.Y "alkdfjs"
f |> printUnion
logSumOfSqrs 42 43
0 // return an integer exit code

```

### Использование в коде на C#:

```

using System;
using System.Collections;
using FS;
using Microsoft.FSharp.Collections;
namespace FSUsing
{
    class Program
    {
        static void Main(string[] args)
        {
            //Just a function
            Console.WriteLine(FS.func.from("a"));
            //Discriminated union
            FS.func.Union union = FS.func.Union.NewX(10);
            FS.func.printUnion(union);
            //Computation expression
            FS.func.logSumOfSqrs(10, 20);
        }
    }
}

```

### Декомпиляция:

#### Pipe operator:

```

namespace FS
open System
module func =
    // Define a function to construct a message to print
    let from whom =
        sprintf "from %s" whom
    [<EntryPoint>]
    let main argv =
        let message = "F#" |> from // Call the function

```

```
printfn "Hello world %s" message
0 // return an integer exit code
```

### Декомпилируется в:

```
using System;
using System.IO;
using System.Reflection;
using Microsoft.FSharp.Core;
[assembly: FSharpInterfaceDataVersion(2, 0, 0)]
[CompilationMapping(SourceConstructFlags.Module)]
public static class @_
{
    public static string from(string whom)
    {
        return PrintfModule.PrintFormatToStringThen(new PrintfFormat<FSharpFunc<string, string>, Unit, string, string, string>("from
%s")).Invoke(whom);
    }
    [EntryPoint]
    public static int Main(string[] argv)
    {
        string func = PrintfModule.PrintFormatToStringThen(new PrintfFormat<FSharpFunc<string, string>, Unit, string, string, string>("from
%s")).Invoke("F#");
        PrintfFormat<FSharpFunc<string, Unit>, TextWriter, Unit, Unit> format = new PrintfFormat<FSharpFunc<string, Unit>, TextWriter,
Unit, Unit, string>("Hello world %s");
        PrintfModule.PrintFormatLineToTextWriter(Console.Out, format).Invoke(func);
        return 0;
    }
}
```

### Discriminated Union и Pattern matching:

```
namespace FS
open System
type Union = X of int | Y of string
let printUnion union =
    match union with
    | X i -> printfn "%i" i
    | Y s -> printfn "%s" s
    | _ -> printfn "default"
[<EntryPoint>]
let main argv =
    let f = Union.Y "alkdfjs"
```

```
f |> printUnion
0 // return an integer exit code
```

### Декомпилируется в:

```
using System;
using System.Collections;
using System.Diagnostics;
using System.IO;
using System.Reflection;
using System.Runtime.CompilerServices;
using System.Runtime.InteropServices;
using Microsoft.FSharp.Core;
[assembly: FSharpInterfaceDataVersion(2, 0, 0)]
[assembly: AssemblyVersion("0.0.0.0")]
[CompilationMapping(SourceConstructFlags.Module)]
public static class @_
{
    [Serializable]
    [StructLayout(LayoutKind.Auto, CharSet = CharSet.Auto)]
    [DebuggerDisplay("{__DebugDisplay(),nq}")]
    [CompilationMapping(SourceConstructFlags.SumType)]
    public abstract class Union : IEquatable<Union>, IStructuralEquatable, IComparable<Union>, IComparable, IStructuralComparable
    {
        public static class Tags
        {
            public const int X = 0;

            public const int Y = 1;
        }
        [Serializable]
        [SpecialName]
        [DebuggerTypeProxy(typeof(X@DebugTypeProxy))]
        [DebuggerDisplay("{__DebugDisplay(),nq}")]
        public class X : Union
        {
            [DebuggerBrowsable(DebuggerBrowsableState.Never)]
            [CompilerGenerated]
            [DebuggerNonUserCode]
            internal readonly int item;

            [CompilationMapping(SourceConstructFlags.Field, 0, 0)]
```



```

[CompilerGenerated]
[DebuggerNonUserCode]
public int Item
{
    [CompilerGenerated]
    [DebuggerNonUserCode]
    get
    {
        return item;
    }
}
[CompilerGenerated]
[DebuggerNonUserCode]
internal X(int item)
{
    this.item = item;
}
}
[Serializable]
[SpecialName]
[DebuggerTypeProxy(typeof(Y@DebugTypeProxy))]
[DebuggerDisplay("__DebugDisplay(),nq")]
public class Y : Union
{
    [DebuggerBrowsable(DebuggerBrowsableState.Never)]
    [CompilerGenerated]
    [DebuggerNonUserCode]
    internal readonly string item;

    [CompilationMapping(SourceConstructFlags.Field, 1, 0)]
    [CompilerGenerated]
    [DebuggerNonUserCode]
    public string Item
    {
        [CompilerGenerated]
        [DebuggerNonUserCode]
        get
        {
            return item;
        }
    }
}

```

```

        [CompilerGenerated]
        [DebuggerNonUserCode]
        internal Y(string item)
        {
            this.item = item;
        }
    }
    [SpecialName]
    internal class X@DebugTypeProxy
    {
        [DebuggerBrowsable(DebuggerBrowsableState.Never)]
        [CompilerGenerated]
        [DebuggerNonUserCode]
        internal X _obj;

        [CompilationMapping(SourceConstructFlags.Field, 0, 0)]
        [CompilerGenerated]
        [DebuggerNonUserCode]
        public int Item
        {
            [CompilerGenerated]
            [DebuggerNonUserCode]
            get
            {
                return _obj.item;
            }
        }
    }

    [CompilerGenerated]
    [DebuggerNonUserCode]
    public X @DebugTypeProxy(X obj)
    {
        _obj = obj;
    }
}
[SpecialName]
internal class Y@DebugTypeProxy
{
    [DebuggerBrowsable(DebuggerBrowsableState.Never)]
    [CompilerGenerated]
    [DebuggerNonUserCode]
    internal Y _obj;

```

```

[CompilationMapping(SourceConstructFlags.Field, 1, 0)]
[CompilerGenerated]
[DebuggerNonUserCode]
public string Item
{
    [CompilerGenerated]
    [DebuggerNonUserCode]
    get
    {
        return _obj.item;
    }
}
[CompilerGenerated]
[DebuggerNonUserCode]
public Y @DebugTypeProxy(Y obj)
{
    _obj = obj;
}
}
[CompilerGenerated]
[DebuggerNonUserCode]
[DebuggerBrowsable(DebuggerBrowsableState.Never)]
public int Tag
{
    [CompilerGenerated]
    [DebuggerNonUserCode]
    get
    {
        return (this is Y) ? 1 : 0;
    }
}
[CompilerGenerated]
[DebuggerNonUserCode]
[DebuggerBrowsable(DebuggerBrowsableState.Never)]
public bool IsX
{
    [CompilerGenerated]
    [DebuggerNonUserCode]
    get
    {
        return this is X;
    }
}

```

```

    }
}
[CompilerGenerated]
[DebuggerNonUserCode]
[DebuggerBrowsable(DebuggerBrowsableState.Never)]
public bool IsY
{
    [CompilerGenerated]
    [DebuggerNonUserCode]
    get
    {
        return this.is Y;
    }
}
[CompilerGenerated]
[DebuggerNonUserCode]
internal Union()
{
}
[CompilationMapping(SourceConstructFlags.UnionCase, 0)]
public static Union NewX(int item)
{
    return new X(item);
}
[CompilationMapping(SourceConstructFlags.UnionCase, 1)]
public static Union NewY(string item)
{
    return new Y(item);
}
[SpecialName]
[CompilerGenerated]
[DebuggerNonUserCode]
internal object __DebugDisplay()
{
    return ExtraTopLevelOperators.PrintFormatToString(new PrintfFormat<FSharpFunc<Union, string>, Unit, string, string, string>("%+0.8A")).Invoke(this);
}
[CompilerGenerated]
public override string ToString()
{
    return ExtraTopLevelOperators.PrintFormatToString(new PrintfFormat<FSharpFunc<Union, string>, Unit, string, string, Union>("%+A")).Invoke(this);
}

```

```

}
[CompilerGenerated]
public sealed override int CompareTo(Union obj)
{
    if (this != null)
    {
        if (obj != null)
        {
            int num = ((this is Y) ? 1 : 0);
            int num2 = ((obj is Y) ? 1 : 0);
            if (num == num2)
            {
                IComparer genericComparer;
                if (this is X)
                {
                    X x = (X)this;
                    X x2 = (X)obj;
                    genericComparer = LanguagePrimitives.GenericComparer;
                    int item = x.item;
                    int item2 = x2.item;
                    if (item < item2)
                    {
                        return -1;
                    }
                    return (item > item2) ? 1 : 0;
                }
                Y y = (Y)this;
                Y y2 = (Y)obj;
                genericComparer = LanguagePrimitives.GenericComparer;
                return string.CompareOrdinal(y.item, y2.item);
            }
            return num - num2;
        }
        return 1;
    }
    if (obj != null)
    {
        return -1;
    }
    return 0;
}
[CompilerGenerated]

```

```

public sealed override int CompareTo(object obj)
{
    return CompareTo((Union)obj);
}
[CompilerGenerated]
public sealed override int CompareTo(object obj, IComparer comp)
{
    Union union = (Union)obj;
    if (this != null)
    {
        if ((Union)obj != null)
        {
            int num = ((this is Y) ? 1 : 0);
            Union union2 = union;
            int num2 = ((union2 is Y) ? 1 : 0);
            if (num == num2)
            {
                if (this is X)
                {
                    X x = (X)this;
                    X x2 = (X)union;
                    int item = x.item;
                    int item2 = x2.item;
                    if (item < item2)
                    {
                        return -1;
                    }
                    return (item > item2) ? 1 : 0;
                }
                Y y = (Y)this;
                Y y2 = (Y)union;
                return string.CompareOrdinal(y.item, y2.item);
            }
            return num - num2;
        }
        return 1;
    }
    if ((Union)obj != null)
    {
        return -1;
    }
    return 0;
}

```

```

}
[CompilerGenerated]
public sealed override int GetHashCode(IEqualityComparer comp)
{
    if (this != null)
    {
        int num = 0;
        if (this is X)
        {
            X x = (X)this;
            num = 0;
            return -1640531527 + (x.item + ((num << 6) + (num >> 2)));
        }
        Y y = (Y)this;
        num = 1;
        string item = y.item;
        return -1640531527 + (((item != null) ? item.GetHashCode() : 0) + ((num << 6) + (num >> 2)));
    }
    return 0;
}
[CompilerGenerated]
public sealed override int GetHashCode()
{
    return GetHashCode(LanguagePrimitives.GenericEqualityComparer);
}
[CompilerGenerated]
public sealed override bool Equals(object obj, IEqualityComparer comp)
{
    if (this != null)
    {
        Union union = obj as Union;
        if (union != null)
        {
            int num = ((this is Y) ? 1 : 0);
            Union union2 = union;
            int num2 = ((union2 is Y) ? 1 : 0);
            if (num == num2)
            {
                if (this is X)
                {
                    X x = (X)this;
                    X x2 = (X)union;

```

```

        return x.item == x2.item;
    }
    Y y = (Y)this;
    Y y2 = (Y)union;
    return string.Equals(y.item, y2.item);
}
return false;
}
return false;
}
return obj == null;
}
[CompilerGenerated]
public sealed override bool Equals(Union obj)
{
    if (this != null)
    {
        if (obj != null)
        {
            int num = ((this is Y) ? 1 : 0);
            int num2 = ((obj is Y) ? 1 : 0);
            if (num == num2)
            {
                if (this is X)
                {
                    X x = (X)this;
                    X x2 = (X)obj;
                    return x.item == x2.item;
                }
                Y y = (Y)this;
                Y y2 = (Y)obj;
                return string.Equals(y.item, y2.item);
            }
            return false;
        }
        return false;
    }
    return obj == null;
}
[CompilerGenerated]
public sealed override bool Equals(object obj)
{

```



```

    Union union = obj as Union;
    if (union != null)
    {
        return Equals(union);
    }
    return false;
}
}
[EntryPoint]
public static int main(string[] argv)
{
    PrintfFormat<FSharpFunc<string, Unit>, TextWriter, Unit, Unit> format = new PrintfFormat<FSharpFunc<string, Unit>, TextWriter, Unit,
Unit, string>("%s");
    PrintfModule.PrintFormatLineToTextWriter(Console.Out, format).Invoke("alkdfjs");
    return 0;
}
}

```

### Computation Expression:

```

namespace FS
open System

module func =
    // Define a function to construct a message to print
    type LoggingBuilder() =
        let log p = printfn "expression is %A" p

        member this.Bind(x, f) =
            log x
            f x

        member this.Return(x) =
            x

let logSumOfSqrs a b =
    let logger = new LoggingBuilder()
    let loggedWorkflow =
        logger
        {
            let! x = a * a
            let! y = b * b

```

```

        let! z = x + y
        return z
    }
    loggedWorkflow

[<EntryPoint>]
let main argv =
    logSumOfSqrs 42 43
    0 // return an integer exit code

```

### Декомпилируется в:

```

using System;
using System.IO;
using System.Reflection;
using System.Runtime.CompilerServices;
using Microsoft.FSharp.Core;
[assembly: FSharpInterfaceDataVersion(2, 0, 0)]
[assembly: AssemblyVersion("0.0.0.0")]
[CompilationMapping(SourceConstructFlags.Module)]
public static class @_
{
    [Serializable]
    [CompilationMapping(SourceConstructFlags.ObjectType)]
    public class LoggingBuilder
    {
        public c Bind<b, c>(b x, FSharpFunc<b, c> f)
        {
            log(x);
            return f.Invoke(x);
        }
        public a Return<a>(a x)
        {
            return x;
        }
        [CompilerGenerated]
        internal void log<a>(a p)
        {
            PrintfFormat<FSharpFunc<a, Unit>, TextWriter, Unit, Unit> format = new PrintfFormat<FSharpFunc<a, Unit>, TextWriter, Unit,
Unit, a>("expression is %A");
            PrintfModule.PrintFormatLineToTextWriter(Console.Out, format).Invoke(p);
        }
    }
}

```

```

    }
    [CompilerGenerated]
    internal static int f@8(LoggingBuilder logger, int x)
    {
        logger.log(43);
        return f@8-1(logger, x, 43);
    }
    [CompilerGenerated]
    internal static int f@8-1(LoggingBuilder logger, int x, int y)
    {
        int num = x + y;
        logger.log(num);
        return num;
    }
    [EntryPoint]
    public static int main(string[] argv)
    {
        LoggingBuilder loggingBuilder = new LoggingBuilder();
        loggingBuilder.log(42);
        int num = f@8(loggingBuilder, 42);
        return 0;
    }
}

```

## Scala:

```

import scala.collection.immutable
import scala.util.chaining.scalaUtilChainingOps
object ScalaFeatures {
    def from(a : String) : String = {
        "from " + a
    }
    def sqrts(xs: List[Double]) : List[Double] = {
        for (x <- xs if 0 <= x)
            yield Math.sqrt(x)
    }
    implicit class Pipe[T](val v : T) extends AnyVal {
        def |>[U](f : T => U): U = f(v)
    }
    abstract class Notification
    case class Email(sender: String, title: String, body: String) extends Notification
    case class SMS(caller: String, message: String) extends Notification
}

```

```

case class VoiceRecording(contactName: String, link: String) extends Notification
def showNotification(notification: Notification): String = {
  notification match {
    case Email(sender, title, _) =>
      s"You got an email from $sender with title: $title"
    case SMS(number, message) =>
      s"You got an SMS from $number! Message: $message"
    case VoiceRecording(name, link) =>
      s"You received a Voice Recording from $name! Click the link to hear it: $link"
  }
}
def main(args : Array[String]): Unit = {
  val a = "Scala"
  print(a.>(x => from(x)))
  print(a.pipe(from))
  val lst = List(1.0, 2.0, 3.0)
  val newLst = sqrts(lst)
  print(newLst)
  val someSms = SMS("12345", "Are you there?")
  val someVoiceRecording = VoiceRecording("Tom", "voicerecording.org/id/123")
  println(showNotification(someSms))
  println(showNotification(someVoiceRecording))
}

```

### Использование в коде на Java:

```

import java.util.*;
import scala.collection.JavaConverters;
import scala.collection.mutable.*;
public class ScalaUsing {
  public static void main(String[] args) {
    //just a function
    String a = "Java";
    System.out.println(ScalaFeatures.from(a));
    //for comprehension
    List<Object> lst = new ArrayList<>();
    lst.add(1.0);
    lst.add(2.0);
    lst.add(3.0);
    System.out.println(ScalaFeatures.sqrts(JavaConverters.asScala(lst).toList()));
  }
}

```

```

        //pattern matching
        ScalaFeatures.SMS email = new ScalaFeatures.SMS("12345", "Are you there?");
        System.out.println(ScalaFeatures.showNotification(email));
    }
}

```

### Pipe operator:

```

object ScalaFeatures {
    def from(a : String) : String = {
        "from " + a
    }

    implicit class Pipe[T](val v : T) extends AnyVal {
        def |>[U](f : T => U): U = f(v)
    }

    def main(args : Array[String]): Unit = {
        val a = "Scala"
        print(a.|>(x => from(x)))
    }
}

```

### Декомпилируется в:

```

//decompiled from ScalaFeatures$.class
import java.lang.invoke.SerializedLambda;
import scala.Predef.;

public final class ScalaFeatures$ {
    public static final ScalaFeatures$ MODULE$ = new ScalaFeatures$();

    public String from(final String a) {
        return (new StringBuilder(5)).append("from ").append(a).toString();
    }

    public Object Pipe(final Object v) {
        return v;
    }

    public void main(final String[] args) {
        String a = "Scala";
        MODULE$.print(ScalaFeatures.Pipe$.MODULE$.bar$greater$extension(this.Pipe(a), (x) -> {

```

```

        return MODULE$.from(x);
    }));
}

private ScalaFeatures$() {

    // $FF: synthetic method
    private static Object $deserializeLambda$(SerializedLambda var0) {
        return var0.lambdaDeserialize<invokedynamic>(var0);
    }
}

//decompiled from ScalaFeatures.class
import scala.Function1;
import scala.reflect.ScalaSignature;
import scala.runtime.BoxesRunTime;

@ScalaSignature(
    bytes =
        "\u0006\u0005\u0005\u0015r!B\n\u0015\u0011\u00039b!B\r\u0015\u0011\u0003Q\u0002\"B\u0011\u0002\t\u0003\u0011\u0003\"B\u0012\u0002\t\u0003!c\u0001\u0002\u001a\u0002\u0007MB\u0001\u0002\u000f\u0003\u0003\u0006\u0004%\t!\u000f\u0005\t\u000b\u0012\u0011\t\u0011)A\u0005u!)\u0011\u0005\u0002C\u0001\r\")! \n\u0002C\u0001\u0017\"9Q\u000bBA\u0001\n\u00032\u0006b\u0002.\u0005\u0003\u0003%\teW\u0004\bC\u0006\t\t\u0011#\u0001c\r\u001d\u0011\u0014!!A\t\u0002\rDQ!\t\t\u0007\u0005\u0002\u0011DQ!\u001a\u0007\u0005\u0006\u0019DqA\u001d\u0007\u0002\u0002\u0013\u00151\u000fC\u0004z\u0019\u0005\u0005IQ\u0001>\t\t\u0011\u0005\u0011\u0011!C\u0002\u0003\u000bAq!!\u0005\u0002\t\u0003\t\u0019\"A\u0007TG\u0006d\u0017MR3biV\u0014Xm\u001d\u0006\u0002+\u00059A(Z7qift4\u0001\u0001\t\u00031\u0005i\u0011\u0001\u0006\u0002\u000e'\u000e\fG.\u0019$fCR,(/Z:\u0014\u0005\u0005Y\u0002C\u0001\u000f \u001b\u0005i\"\" \u0001\u0010\u0002\u000bM\u001c\u0017\r\\1\n\u0005\u0001j\"AB!osJ+g-\u0001\u0004=S:LGO\u0010\u000b\u0002/\u0005!aM]8n)\t)\u0003\u0007\u0005\u0002'[9\u0011qe\u000b\t\u0003Qui\u0011!\u000b\u0006\u0003UY\ta\u0001\u0010:p_Rt\u0014B\u0001\u0017\u001e\u0003\u0019\u0001&/\u001a3fM&\u0011af\f\u0002\u0007'R\u0014\u0018N\\4\u000b\u00051j\u0002\"B\u0019\u0004\u0001\u0004)\u0013!A1\u0003\tAK\u0007/Z\u000b\u0003iq\u001a\" \u0001B\u001b\u0011\u0005q1\u0014BA\u001c\u001e\u0005\u0019\te.\u001f,bY\u0006\ta/F\u0001;! \tYD\b\u0004\u0001\u0005\u000bu\"!\u0019\u0001\u0003\u0003Q\u000b\"a\u0010\" \u0011\u0005q\u0001\u0015BA!\u001e\u0005\u001dqu\u000e\u001e5jj\u001e\u0004\" \u0001H\" \n\u0005\u0011k\"aA!os\u0006\u0011a\u000f\t\u000b\u0003\u000f&\u00032\u0001\u0013\u0003;\u001b\u0005\t\u0001\" \u0002\u001d\b\u0001\u0004Q\u0014\u0001\u0004\u0013cCJ$S]3bi\u0016\u0014XC\u0001'O)\ti\u0005\u000b\u0005\u0002<\u001d\u0012)q\n\u0003b\u0001}\t\tQ\u000bC\u0003R\u0011\u0001\u0007!+A\u0001g!\u0011a2KO'\n\u0005Qk\"!\u0003$ v]\u000e$\u0018n\u001c82\u0003!A\u0017m\u001d5D_\u0012,G#A,\u0011\u0005qA\u0016BA-\u001e\u0005\u0005rIe\u000e^\u0001\u0007KF,\u0018\r\\:\u0015\u0005q{\u0006C\u0001\u000f^\u0013\tqVDA\u0004C_>dW-\u00198\t\t\u000f\u0001T\u0011\u0011!a\u0001\u0005\u0006\u0019\u0001\u0010J\u0019\u0002\tAK\u0007/\u001a\t\t\u0003\u00112\u0019\" \u0001D\u000e\u0015\u0003\t\fa\u0003\n2be\u0012:'/Z1uKJ$S\r\u001f;f]NLwN\\ \u000b\u0004O*tGC\u00015p)\tI7\u000e\u0005\u0002<U\u0012)qJ\u0004b\u0001!)\u0011K\u0004a\u0001YB!AdU7j!\tYd\u000eB\u0003>\u001d\t\t\u0007a\bC\u0003q\u001d\u0001\u0000

```

7\u0011/A\u0003%i\"L7\u000fE\u0002I\t5\f!\u0003[1tQ\u000e{G-\u001a\u0013fqR,gn]5p]V\u0011A\u000f\u001f\u000b\u0003-  
VDQ\u0001]\bA\u0002Y\u00042\u0001\u0013\u0003x!\tY\u0004\u0010B\u0003>\u001f\t\u0007a(\u0001\tfcV\fGn\u001d\u0013fqR,gn]5  
p]V\u001910a\u0001\u0015\u0005qtHC\u0001/~\u0011\u001d\u0001\u0007#!AA\u0002\tCQ\u0001\u001d\tA\u0002}\u0004B\u0001\u0013  
\u0003\u0002\u0002A\u00191(a\u0001\u0005\u000bu\u0002\"\\u0019\u0001  
\u0016\t\u0005\u001d\u0011Q\u0002\u000b\u0005\u0003\u0013\ty\u0001\u0005\u0003I\t\u0005-  
\u0001cA\u001e\u0002\u000e\u0011)Q(\u0005b\u0001}!1\u0001(\u0005a\u0001\u0003\u0017\tA!\\1j]R!\u0011QCA\u000e!\ra\u0012qC  
\u0005\u0004\u00033i\"\\u0001B+oSRDq!!\b\u0013\u0001\u0004\ty\"\\u0001\u0003be\u001e\u001c\b\u0003\u0002\u000f\u0002\"\\u001  
5Jl!a\t\u001e\u0005\u0015\t%O]lz\u0001"

)

```
public final class ScalaFeatures {
    public static void main(final String[] args) {
        ScalaFeatures$.MODULE$.main(args);
    }

    public static Object Pipe(final Object v) {
        return ScalaFeatures$.MODULE$.Pipe(v);
    }

    public static String from(final String a) {
        return ScalaFeatures$.MODULE$.from(a);
    }

    public static final class Pipe {
        private final Object v;

        public Object v() {
            return this.v;
        }

        public Object $bar$greater(final Function1 f) {
            return ScalaFeatures.Pipe$.MODULE$. $bar$greater$extension(this.v(), f);
        }

        public int hashCode() {
            return ScalaFeatures.Pipe$.MODULE$.hashCode$extension(this.v());
        }

        public boolean equals(final Object x$1) {
            return ScalaFeatures.Pipe$.MODULE$.equals$extension(this.v(), x$1);
        }
    }
}
```

```

    public Pipe(final Object v) {
        this.v = v;
    }
}

public static class Pipe$ {
    public static final ScalaFeatures.Pipe$ MODULE$ = new ScalaFeatures.Pipe$();

    public final Object $bar$greater$extension(final Object $this, final Function1 f) {
        return f.apply($this);
    }

    public final int hashCode$extension(final Object $this) {
        return $this.hashCode();
    }

    public final boolean equals$extension(final Object $this, final Object x$1) {
        boolean var3;
        if (x$1 instanceof ScalaFeatures.Pipe) {
            var3 = true;
        } else {
            var3 = false;
        }

        boolean var10000;
        if (var3) {
            Object var5 = x$1 == null ? null : ((ScalaFeatures.Pipe)x$1).v();
            if (BoxesRunTime.equals($this, var5)) {
                var10000 = true;
                return var10000;
            }
        }

        var10000 = false;
        return var10000;
    }
}
}

```

**Pipe operator:**



```
object ScalaFeatures {
  def from(a : String) : String = {
    "from " + a
  }

  def main(args : Array[String]): Unit = {
    val a = "Scala"
    print(a.pipe(from))
  }
}
```

**Декомпилируется в:**

```
//decompiled from ScalaFeatures$.class
import java.lang.invoke.SerializedLambda;
import scala.Predef.;

public final class ScalaFeatures$ {
  public static final ScalaFeatures$ MODULE$ = new ScalaFeatures$();

  public String from(final String a) {
    return (new StringBuilder(5)).append("from ").append(a).toString();
  }

  public void main(final String[] args) {
    String a = "Scala";

    MODULE$.print(scala.util.ChainingOps..MODULE$.pipe$extension(scala.util.package.chaining..MODULE$.scalaUtilChainingOps(a), (ax) -> {
      return MODULE$.from(ax);
    }));
  }

  private ScalaFeatures$() {
  }

  // $FF: synthetic method
  private static Object $deserializeLambda$(SerializedLambda var0) {
    return var0.lambdaDeserialize<invokedynamic>(var0);
  }
}
```

```

}

//decompiled from ScalaFeatures.class
import scala.reflect.ScalaSignature;

@ScalaSignature(
  bytes =
    "\u0006\u00059:Q!\u0002\u0004\t\u0002%1Qa\u0003\u0004\t\u00021AQaE\u0001\u0005\u0002QAQ!F\u0001\u0005\u0002YAQ\u0001J\u0001\u0005\u0002\u0015\u0015\u00bU2bY\u00064U-\u0019;ve\u0016\u001c(\\"A\u0004\u0002\u000fQ*W\u000e\u001d;z}\r\u0001\u0001C\u0001\u0006\u0002\u001b\u00051!!D*dC2\fg)Z1ukJ,7o\u0005\u0002\u0002\u001bA\u0011a\"E\u0007\u0002\u001f)\t\u0001#A\u0003tG\u0006d\u0017-\u0003\u0002\u0013\u001f\t1\u0011I\\=SK\u001a\fa\u0001P5oSRtD#A\u0005\u0002\t\u0019\u0014x.\u001c\u000b\u0003/\t\u0002\"\\u0001G\u0010\u000f\u0005ei\u0002C\u0001\u000e\u0010\u001b\u0005Y\"B\u0001\u000f\t\u0003\u0019a$o\\8u}%\u0011adD\u0001\u0007!J,G-\u001a4\n\u0005\u0001\n#AB*ue&twM\u0003\u0002\u001f\u001f!)1e\u0001a\u0001/\u0005\t\u0011-\u0001\u0003nC&tGC\u0001\u0014*!\tqq%\u0003\u0002)\u001f\t!QK\\5u\u0011\u0015QC\u0001\u0001,\u0003\u0011\t'oZ:\u0011\u00079as#\u0003\u0002.\u001f\t)\u0011I\u001d:bs\u0002"
)
public final class ScalaFeatures {
  public static void main(final String[] args) {
    ScalaFeatures$.MODULE$.main(args);
  }

  public static String from(final String a) {
    return ScalaFeatures$.MODULE$.from(a);
  }
}

```

### For comprehension:

```

import scala.collection.immutable
import scala.util.chaining.scalaUtilChainingOps

object ScalaFeatures {

  def sqrts(xs: List[Double]) : List[Double] = {
    for (x <- xs if 0 <= x)
      yield Math.sqrt(x)
  }

  def main(args : Array[String]): Unit = {

```

```

    val lst = List(1.0, 2.0, 3.0)
    val newLst = sqrts(lst)
    print(newLst)
  }
}

```

### Декомпилируется в:

```

//decompiled from ScalaFeatures$.class
import java.lang.invoke.SerializedLambda;
import scala.collection.immutable.List;
import scala.package.;

public final class ScalaFeatures$ {
    public static final ScalaFeatures$ MODULE$ = new ScalaFeatures$();

    public List sqrts(final List xs) {
        return (List)xs.withFilter((x) -> {
            return (double)0 <= x;
        }).map((x) -> {
            return Math.sqrt(x);
        });
    }

    public void main(final String[] args) {
        List lst = (List).MODULE$.List().apply(scala.runtime.ScalaRunTime..MODULE$.wrapDoubleArray(new double[]{1.0D, 2.0D, 3.0D}));
        List newLst = this.sqrts(lst);
        scala.Predef..MODULE$.print(newLst);
    }

    private ScalaFeatures$() {
    }

    // $FF: synthetic method
    private static Object $deserializeLambda$(SerializedLambda var0) {
        return Class.lambdaDeserialize<invokedynamic>(var0);
    }
}

//decompiled from ScalaFeatures.class

```

```

import scala.collection.immutable.List;
import scala.reflect.Signature;

@ScalaSignature(
  bytes =
    "\u0006\u0005i:Q!\u0002\u0004\t\u0002%1Qa\u0003\u0004\t\u00021AQaE\u0001\u0005\u0002QAQ!F\u0001\u0005\u0002YAQ\u0001K\u0001\u0005\u0002%\nQbU2bY\u00064U-\u0019;ve\u0016\u001c(\\"A\u0004\u0002\u000fQ*W\u000e\u001d;z}\r\u0001\u0001C\u0001\u0006\u0002\u001b\u00051!!D*dC2\fg)Z1ukJ,7o\u0005\u0002\u0002\u001bA\u0011a\\"E\u0007\u0002\u001f)\t\u0001#A\u0003tG\u0006d\u0017-\u0003\u0002\u0013\u001f\t1\u0011I\\=SK\u001a\fa\u0001P5oSRtD#A\u0005\u0002\u000bM\fb(\u000f^\u0015\u0005]1\u0003c\u0001\r!G9\u0011\u0011D\b\b\u00035ui\u0011a\u0007\u0006\u00039!\ta\u0001\u0010:p_Rt\u0014\\"\u0001\t\n\u0005}y\u0011a\u00029bG.\fw-Z\u0005\u0003C\t\u0012A\u0001T5ti*\u0011qd\u0004\t\u0003\u001d\u0011J!!J\b\u0003\r\u0011{WO\u00197f\u0011\u001593\u0001\u0001\u0018\u0003\tA8/\u0001\u0003nC&tGC\u0001\u0016.!\tq1&\u0003\u0002-\u001f\t!QK\5u\u0011\u0015qC\u0001\u0001\u0003\u0011\t'oZ:\u0011\u00079\u0001$\u0003\u00022\u001f\t)\u0011I\u001d:bsB\u0011lg\u000e\b\u0003iU\u0002\\"AG\b\n\u0005Yz\u0011A\u0002) sK\u0012,g-\u0003\u00029s\t11\u000b\u001e:jj\u001eT!AN\b"
)
public final class ScalaFeatures {
  public static void main(final String[] args) {
    ScalaFeatures$.MODULE$.main(args);
  }

  public static List sqrts(final List xs) {
    return ScalaFeatures$.MODULE$.sqrts(xs);
  }
}

```

### Pattern matching:

```

object ScalaFeatures {

  abstract class Notification

  case class Email(sender: String, title: String, body: String) extends Notification

  case class SMS(caller: String, message: String) extends Notification

  case class VoiceRecording(contactName: String, link: String) extends Notification

  def showNotification(notification: Notification): String = {

```

```

notification match {
  case Email(sender, title, _) =>
    s"You got an email from $sender with title: $title"
  case SMS(number, message) =>
    s"You got an SMS from $number! Message: $message"
  case VoiceRecording(name, link) =>
    s"You received a Voice Recording from $name! Click the link to hear it: $link"
}

def main(args : Array[String]): Unit = {
  val someSms = SMS("12345", "Are you there?")
  val someVoiceRecording = VoiceRecording("Tom", "voicerecording.org/id/123")
  println(showNotification(someSms))
  println(showNotification(someVoiceRecording))
}

```

**Декомпилируется в:**

```

//decompiled from ScalaFeatures$.class
import scala.MatchError;
import scala.Predef.;

public final class ScalaFeatures$ {
  public static final ScalaFeatures$ MODULE$ = new ScalaFeatures$();

  public String showNotification(final ScalaFeatures.Notification notification) {
    String var2;
    if (notification instanceof ScalaFeatures.Email) {
      ScalaFeatures.Email var4 = (ScalaFeatures.Email)notification;
      String sender = var4.sender();
      String title = var4.title();
      var2 = (new StringBuilder(35)).append("You got an email from ").append(sender).append(" with title:
").append(title).toString();
    } else if (notification instanceof ScalaFeatures.SMS) {
      ScalaFeatures.SMS var7 = (ScalaFeatures.SMS)notification;
      String number = var7.caller();
      String message = var7.message();
      var2 = (new StringBuilder(31)).append("You got an SMS from ").append(number).append("! Message:
").append(message).toString();
    } else {

```

```

        if (!(notification instanceof ScalaFeatures.VoiceRecording)) {
            throw new MatchError(notification);
        }

        ScalaFeatures.VoiceRecording var10 = (ScalaFeatures.VoiceRecording)notification;
        String name = var10.contactName();
        String link = var10.link();
        var2 = (new StringBuilder(65)).append("You received a Voice Recording from ").append(name).append("! Click the
link to hear it: ").append(link).toString();
    }

    return var2;
}

public void main(final String[] args) {
    ScalaFeatures.SMS someSms = new ScalaFeatures.SMS("12345", "Are you there?");
    ScalaFeatures.VoiceRecording someVoiceRecording = new ScalaFeatures.VoiceRecording("Tom",
"voicerecording.org/id/123");
    .MODULE$.println(this.showNotification(someSms));
    .MODULE$.println(this.showNotification(someVoiceRecording));
}

private ScalaFeatures$() {
}
}

//decompiled from ScalaFeatures.class
import java.io.Serializable;
import scala.Option;
import scala.Product;
import scala.Some;
import scala.Tuple2;
import scala.Tuple3;
import scala.collection.Iterator;
import scala.reflect.Signature;
import scala.runtime.AbstractFunction2;
import scala.runtime.AbstractFunction3;
import scala.runtime.ModuleSerializationProxy;
import scala.runtime.Statics;

@ScalaSignature(

```

bytes =

"\u0006\u0005\t%u!B+W\u0011\u0003If!B.W\u0011\u0003a\u0006\"B2\u0002\t\u0003!g!B3\u0002\u0003\u00031\u0007\"B2\u0004\t\u00039g\u0001\u00026\u0002\u0001.D\u0001b\_\u0003\u0003\u0016\u0004%\t\u0001  
\u0005\n\u0003\u0017)! \u0011#Q\u0001\u0001\u0011\"!\u0004\u0006\u0005+\u0007I\u0011\u0001?\t\u0013\u0005=QA!E!\u0002\u0013i  
\b\"CA\t\u000b\tU\r\u0011\" \u0001}\u0011%\t\u0019\" \u0002B\tB\u0003Q\u0010\u0003\u0004d\u000b\u0011\u0005\u0011Q\u0003\u0005\n\u0003?) \u0011\u0011!C\u0001\u0003CA\u0011\"!\u000b\u0006#\u0003%\t!a\u000b\t\u0013\u0005\u0005S!%A\u0005\u0002\u0005-  
\u0002\"CA\" \u000bE\u0005I\u0011AA\u0016\u0011%\t)%BA\u0001\n\u0003\n9\u0005C\u0005\u0002X\u0015\t\t\u0011\" \u0001\u0002Z  
!I\u0011\u0011M\u0003\u0002\u0002\u0013\u0005\u00111\r\u0005\n\u0003\_\*\u0011\u0011!C!\u0003cB\u0011\"a  
\u0006\u0003\u0003%\t!!!\t\u0013\u0005-  
U!A\u0005B\u00055\u0005\"CAI\u000b\u0005\u0005I\u0011IAJ\u0011%\t)\*BA\u0001\n\u0003\n9\nnC\u0005\u0002\u001a\u0016\t\t\u0001\u0011\" \u0011\u0002\u001c\u001eI\u0011qT\u0001\u0002\u0002#\u0005\u0011\u0011\u0015\u0004\tU\u0006\t\t\u0011#\u0001\u0002\$\n\"11m\u0007C\u0001\u0003wC\u0011\"!&\u001c\u0003\u0003)%a&\t\u0013\u0005u6\$!A\u0005\u0002\u0006}\u0006\"Cad7\u0005\u0005I  
\u0011QAe\u0011%\tYnGA\u0001\n\u0013\tiN\u0002\u0004\u0002f\u0006\u0001\u0015q\u001d\u0005\n\u0003S\f#Q3A\u0005\u0002qD\u0011\"a;\n\u0005#\u0005\u000b\u0011B?\t\u0013\u00055\u0018E!f\u0001\n\u0003a\b\"CAxC\tE\t\u0015!\u0003~\u0011\u0019\u0019  
\u0017\u0005\" \u0001\u0002r\"I\u0011qD\u0011\u0002\u0002\u0013\u0005\u0011\u0011  
\u0005\n\u0003S\t\u0013\u0013\u0013!C\u0001\u0003WA\u0011\"!\u0011\"#\u0003%\t!a\u000b\t\u0013\u0005\u0015\u0013%!A\u0005B\u0005  
5\u001d\u0003\"CA,C\u0005\u0005I\u0011AA-  
\u0011%\t\t'IA\u0001\n\u0003\tty\u0010C\u0005\u0002p\u0005\n\t\u0011\" \u0011\u0002r!I\u0011qP\u0011\u0002\u0002\u0013\u0005  
5!1\u0001\u0005\n\u0003\u0017\u000b\u0013\u0011!C!\u0005\u000fA\u0011\"!%\" \u0003\u0003%\t%a%\t\u0013\u0005U\u0015%!A\u0005  
05B\u0005]\u0005\"CAMC\u0005\u0005I\u0011\tB\u0006\u000f%\u0011y!AA\u0001\u0012\u0003\u0011\tBB\u0005\u0002f\u0006\t\t\u0001  
011#\u0001\u0003\u0014!11\r\u000eC\u0001\u00057A\u0011\"!&5\u0003\u0003)%a&\t\u0013\u0005uF'!A\u0005\u0002\n\u0001\"Cad  
i\u0005\u0005I\u0011\u0011B\u0012\u0011%\tY\u000eNA\u0001\n\u0013\tiN\u0002\u0004\u00030\u0005\u0001%\u0011\u0007\u0005\n\u0005gQ\$Q3A\u0005\u0002qD\u0011B!\u000e;\u0005#\u0005\u000b\u0011B?\t\u0013\tj\"H!f\u0001\n\u0003a\b\" \u0003B\u001du\tE\t  
\u0015!\u0003~\u0011\u0019\u0019'\b\" \u0001\u0003<!I\u0011q\u0004\u001e\u0002\u0002\u0013\u0005!1\t\u0005\n\u0003SQ\u0014\u0013!C\u0001\u0003WA\u0011\"!\u0011;#\u0003%\t!a\u000b\t\u0013\u0005\u0015#(!A\u0005B\u0005\u001d\u0003\"CA,u\u0005\u0005I\u0011AA-  
\u0011%\t\tGOA\u0001\n\u0003\u0011I\u0005C\u0005\u0002pi\n\t\u0011\" \u0011\u0002r!I\u0011q\u0010\u001e\u0002\u0002\u0013\u0005  
u0005!Q\n\u0005\n\u0003\u0017S\u0014\u0011!C!\u0005#B\u0011\"!%; \u0003\u0003%\t%a%\t\u0013\u0005U%(!A\u0005B\u0005]\u0005  
\"CAMu\u0005\u0005I\u0011\tB+\u000f%\u0011I&AA\u0001\u0012\u0003\u0011YFB\u0005\u00030\u0005\t\t\u0011#\u0001\u0003^!11-  
\u0014C\u0001\u0005CB\u0011\"!&N\u0003\u0003)%a&\t\u0013\u0005uV\*!A\u0005\u0002\n\r\u0004\"Cad\u001b\u0006\u0005I\u0011\u0011B5\u0011%\tY.TA\u0001\n\u0013\ti\u000eC\u0004\u0003n\u0005!\tAa\u001c\t\u000f\tU\u0014\u0001\" \u0001\u0003x\u0005i1k  
Y1mC\u001a+\u0017\r^;sKNT\u0011aV\u0001\b\u00151\u0007\u000f^=?\u0007\u0001\u0001\"AW\u0001\u000e\u0003Y\u0013QbU2bY\u0005  
064U-\u0019;ve\u0016\u001c8CA\u0001^!\tq\u0016-  
D\u0001` \u0015\u0005\u0001\u0017!B:dc2\f\u0017B\u0002` \u0005\u0019\te. \u001f\*fM\u00061A(\u001b8jiz\" \u0012! \u0017\u0002\  
r\u001d>\$\u0018NZ5dCRLwN\\ \u0003\u0007u#\u0012\u0001\u001b\t\u0003S\u000ei\u0011! \u0001\u0002\u0006\u000b6\f\u0017\u0005  
e\\ \u0005\u000b!dw\u000e\u0005\u0002\_ [&\u0011an\u0018\u0002\b!J|G-  
^2u!\t\u0001\bP\u0004\u0002rm:\u0011!/^ \u0007\u0002g\*\u0011A\u000fW\u0001\u0007yI|w\u000e\u001e  
\n\u0003\u0001L!a^0\u0002\u000fA\f7m[1hK&\u0011\u0011P\u001f\u0002\r' \u0016\u0014\u0018. \u00197ju\u0006\u0014G. \u001a\u0006\u0003o~\u000bbaa]3oI\u0016\u0014X#A?\u0011\u0007y\f)AD\u0002\u0000\u0003\u0003\u0001\"A]0\n\u0007\u0005\rq, \u0001\u0004  
Qe\u0016\$WMZ\u0005\u0005\u0003\u000f\tIA\u0001\u0004TiJLgn\u001a\u0006\u0004\u0003\u0007y\u0016aB:f] \u0012, ' \u000fI\u0001

\u0006i&\$H.Z\u0001\u0007i&\$H.\u001a\u0011\u0002\t\t|G-  
\_\u0001\u0006E>\$\u0017\u0010\t\u000b\t\u0003/\tI\"a\u0007\u0002\u001eA\u0011\u0011.\u0002\u0005\u0006w2\u0001\r!  
\u0005\u0007\u0003\u001ba\u0001\u0019A?\t\r\u0005EA\u00021\u0001~\u0003\u0011\u0019w\u000e]=\u0015\u0011\u0005]\u0011EA\  
u0013\u0003OAqa\_\u0007\u0011\u0002\u0003\u0007Q\u0010\u0003\u0005\u0002\u000e5\u0001\n\u0011\u0001~\u0011!\t\t\"u0004I\  
u0001\u0002\u0004i\u0018AD2paf\$C-  
\u001a4bk2\$H%M\u000b\u0003\u0003[Q3!`A\u0018W\t\t\t\u0004\u0005\u0003\u00024\u0005uRBAA\u001b\u0015\u0011\t9\$!\u000f\u000  
2\u0013Ut7\r[3dW\u0016\$'bAA\u001e?\u0006Q\u0011M\u001c8pi\u0006\$\u0018n\u001c8\n\t\u0005}\u0012Q\u0007\u0002\u0012k:\u001  
c\u0007.Z21K\u00124\u0016M]5b]\u000e,\u0017AD2paf\$C-  
\u001a4bk2\$HEM\u0001\u000fG>\u0004\u0018\u0010\n3fM\u0006,H\u000e\u001e\u00134\u00035\u0001(o\u001c3vGR\u0004&/\u001a4jqV  
\u0011\u0011\u0011\n\t\u0005\u0003\u0017\n)&\u0004\u0002\u0002N)! \u0011qJA) \u0003\u0011a\u0017M\\4\u000b\u0005\u0005M\u00  
13\u0001\u00026bm\u0006LA!a\u0002\u0002N\u0005a\u0001O]8ek\u000e\$\u0018I]5usV\u0011\u0011\u0011\f\t\u0004=\u0006u\u0013bAA0?\n  
\u0019\u0011J\u001c;\u0002\u001dA\u0014x\u000eZ;di\u0016cW-  
\\3oiR!\u0011QMA6!\rq\u0016qM\u0005\u0004\u0003Sz&aA!os\"I\u0011QN\n\u0002\u0002\u0003\u0007\u0011L\u0001\u0004q\u0012\n  
\u0014a\u00049s\_\u0012,8\r^%uKJfGo\u001c:\u0016\u0005\u0005M\u0004CBA;\u0003w\n)' \u0004\u0002\u0002x) \u0019\u0011\u0011P  
0\u0002\u0015\r|G\u000e\\3di&|g.\u0003\u0003\u0002~\u0005]\$ \u0001C%uKJfGo\u001c:\u0002\u0011\rfg.R9vC2\$B!a!\u0002\nB\u00  
019a,!\" \u0007\u0005\u001duLA\u0004C\_>dW-  
\u00198\t\u0013\u00055T#!AA\u0002\u0005\u0015\u0014A\u00059s\_\u0012,8\r^#mK6,g\u000e\u001e(b[\u0016\$B!!\u0013\u0002\u0010  
\"I\u0011Q\u000e\f\u0002\u0002\u0003\u0007\u0011L\u0001\tQ\u0006\u001c\bnQ8eKR\u0011\u0011L\u0001\ti>\u001cFO]5oOR\u001  
1\u0011\u0011J\u0001\u0007KF,\u0018\r\\:\u0015\t\u0005\r\u0015Q\u0014\u0005\n\u0003[J\u0012\u0011!a\u0001\u0003K\nQ!R7bS2  
\u0004\"! [\u000e\u0014\u000bm\t)+!-  
\u0011\u0013\u0005\u001d\u0016QV?~{\u0006]QBAAU\u0015\r\tYkX\u0001\beVtG/[7f\u0013\u0011\tty+!+\u0003#\u0005\u00137\u000f\  
u001e:bGR4UO\\2uS>t7\u0007\u0005\u0003\u00024\u0006eVBAA[\u0015\u0011\t9,! \u0015\u0002\u0005%|\u0017bA=\u00026R\u0011\u00  
11\u0011U\u0001\u0006CB\u0004H.\u001f\u000b\t\u0003/\t\t-  
a1\u0002F\")1P\ba\u0001{\\"1\u0011Q\u0002\u0010A\u0002uDa!!\u0005\u001f\u0001\u0004i\u0018aB;oCB\u0004H.\u001f\u000b\u0005  
\u0003\u0017\f9\u000eE\u0003\_\u0003\u001b\f\t.C\u0002\u0002P~\u0013aa\u00149uS>t\u0007C\u00020\u0002Tv1X0C\u0002\u0002V~\  
u0013a\u0001V;qY\u0016\u001c\u0004\"CAm?\u0005\u0005\t\u0019AA\f\u0003\rAH\u0005M\u0001\roJLG/\u001a\*fa2\f7-  
\u001a\u000b\u0003\u0003?\u0004B!a\u0013\u0002b&!\u0011]A'\u0005\u0019y%M[3di\n\u00191+T\*\u0014\t\u0005BGn\\\u0001\u0007  
G\u0006dG.\u001a:\u0002\u000f\r\fG\u000e\\3sA\u00059Q.Z:tC\u001e,\u0017\u0001C7fgNfw-  
\u001a\u0011\u0015\r\u0005M\u0018Q\_A]!\tI\u0017\u0005\u0003\u0004\u0002j\u001a\u0002\r!  
\u0005\u0007\u0003[4\u0003\u0019A?\u0015\r\u0005M\u00181`A\u007f\u0011!\tIo\nI\u0001\u0002\u0004i\b\u0002CAwOA\u0005\t\u00  
019A?\u0015\t\u0005\u0015\$\u0011\u0001\u0005\n\u0003[b\u0013\u0011!a\u0001\u00037\"B!a!\u0003\u0006!I\u0011Q\u000e\u0018\  
u0002\u0002\u0003\u0007\u0011Q\r\u000b\u0005\u0003\u0013\u0012I\u0001C\u0005\u0002n=\n\t\u0011\u0001\u0002\\Q!\u0011\u00  
011B\u0007\u0011%tiGMA\u0001\u0002\u0004\t)'A\u0002T\u001bN\u0003\"!\u001b\u001b\u0014\u000bQ\u0012)\"!-  
\u0011\u0011\u0005\u001d&qC?~\u0003gLAA!\u0007\u0002\*\n\t\u0012IY:ue\u0006\u001cGOR;oGRLwN\u001c\u001a\u0015\u0005\tEACBA  
z\u0005?\u0011\t\u0003\u0003\u0004\u0002j^ \u0002\r!  
\u0005\u0007\u0003[<\u0004\u0019A?\u0015\t\t\u0015\"Q\u0006\t\u0006=\u00065'q\u0005\t\u0006=\n%R0`\u0005\u0004\u0005Wy&A\  
u0002+va2,' \u0007C\u0005\u0002Zb\n\t\u0011\u0001\u0002t\ngak\\5dKJ+7m\u001c:eS:<7\u0003\u0002\u001eiY>\f1bY8oi\u0006\u00  
1cGOT1nK\u0006a1m\u001c8uC\u000e\$H\*Y7fA\u0005!A.\u001b81\u0003\u0015a\u0017N\\6!)\u0019\u0011iDa\u0010\u0003BA\u0011\u001  
1N\u000f\u0005\u0007\u0005gy\u0004\u0019A?\t\r\t]r\b1\u0001~)\u0019\u0011iD!\u0012\u0003H!A!1\u0007!\u0011\u0002\u0003\u00  
007Q\u0010\u0003\u0005\u00038\u0001\u0003\n\u0011\u0001~)\u0011\t)Ga\u0013\t\u0013\u00055T)!AA\u0002\u0005mC\u0003BAB\u00  
005\u001fB\u0011\"!\u001cH\u0003\u0003\u0005\r!!\u001a\u0015\t\u0005%#1\u000b\u0005\n\u0003[B\u0015\u0011!a\u0001\u00037\



```

"B!a!\u0003X!I\u0011QN&\u0002\u0002\u0003\u0007\u0011QM\u0001\u000f->L7-\u001a*fG>\u0014H-
\u001b8h!\tIWjE\u0003N\u0005?\n\t\f\u0005\u0005\u0002(\n]Q0
B\u001f)\t\u0011Y\u0006\u0006\u0004\u0003>\t\u0015$q\r\u0005\u0007\u0005g\u0001\u0006\u0019A?\t\r\t]\u0002\u000b1\u0001~)
\u0011\u0011)Ca\u001b\t\u0013\u0005e\u0017+!AA\u0002\tu\u0012\u0001E:i_^tu\u000e^5gS\u000e\fG/[8o)\ri(\u0011\u000f\u0005\
\u0007\u0005g\u001a\u0006\u0019\u00015\u0002\u00199|G/\u001b4jG\u0006$\u0018n\u001c8\u0002\t5\f\u0017N\u001c\u000b\u0005\u
0005s\u0012y\bE\u0002_\u0005wJ1A!
`\u0005\u0011)f.\u001b;\t\u000f\t\u0005E\u000b1\u0001\u0003\u0004\u0006!\u0011M]4t!\u0011q&QQ?\n\u0007\t\u001duLA\u0003Be
J\f\u0017\u0010"
)
public final class ScalaFeatures {
    public static void main(final String[] args) {
        ScalaFeatures$.MODULE$.main(args);
    }

    public static String showNotification(final ScalaFeatures.Notification notification) {
        return ScalaFeatures$.MODULE$.showNotification(notification);
    }

    public static class Email extends ScalaFeatures.Notification implements Product, Serializable {
        private final String sender;
        private final String title;
        private final String body;

        public Iterator productElementNames() {
            return Product.productElementNames$(this);
        }

        public String sender() {
            return this.sender;
        }

        public String title() {
            return this.title;
        }

        public String body() {
            return this.body;
        }

        public ScalaFeatures.Email copy(final String sender, final String title, final String body) {
            return new ScalaFeatures.Email(sender, title, body);
        }
    }

```

```

}

public String copy$default$1() {
    return this.sender();
}

public String copy$default$2() {
    return this.title();
}

public String copy$default$3() {
    return this.body();
}

public String productPrefix() {
    return "Email";
}

public int productArity() {
    return 3;
}

public Object productElement(final int x$1) {
    Object var10000;
    switch(x$1) {
        case 0:
            var10000 = this.sender();
            break;
        case 1:
            var10000 = this.title();
            break;
        case 2:
            var10000 = this.body();
            break;
        default:
            var10000 = Statics.ioobe(x$1);
    }

    return var10000;
}

```

```

public Iterator productIterator() {
    return scala.runtime.ScalaRunTime..MODULE$.typedProductIterator(this);
}

public boolean canEqual(final Object x$1) {
    return x$1 instanceof ScalaFeatures.Email;
}

public String productElementName(final int x$1) {
    String var10000;
    switch(x$1) {
        case 0:
            var10000 = "sender";
            break;
        case 1:
            var10000 = "title";
            break;
        case 2:
            var10000 = "body";
            break;
        default:
            var10000 = (String)Statics.ioobe(x$1);
    }

    return var10000;
}

public int hashCode() {
    return scala.runtime.ScalaRunTime..MODULE$._hashCode(this);
}

public String toString() {
    return scala.runtime.ScalaRunTime..MODULE$._toString(this);
}

public boolean equals(final Object x$1) {
    boolean var8;
    if (this != x$1) {
        label72: {
            boolean var2;
            if (x$1 instanceof ScalaFeatures.Email) {

```

```

    var2 = true;
} else {
    var2 = false;
}

if (var2) {
    label154: {
        label63: {
            ScalaFeatures.Email var4 = (ScalaFeatures.Email)x$1;
            String var10000 = this.sender();
            String var5 = var4.sender();
            if (var10000 == null) {
                if (var5 != null) {
                    break label63;
                }
            } else if (!var10000.equals(var5)) {
                break label63;
            }

            var10000 = this.title();
            String var6 = var4.title();
            if (var10000 == null) {
                if (var6 != null) {
                    break label63;
                }
            } else if (!var10000.equals(var6)) {
                break label63;
            }

            var10000 = this.body();
            String var7 = var4.body();
            if (var10000 == null) {
                if (var7 != null) {
                    break label63;
                }
            } else if (!var10000.equals(var7)) {
                break label63;
            }

            if (var4.canEqual(this)) {
                var8 = true;
            }
        }
    }
}

```

```

        break label154;
    }
}

var8 = false;
}

if (var8) {
    break label72;
}

var8 = false;
return var8;
}
}

var8 = true;
return var8;
}

public Email(final String sender, final String title, final String body) {
    this.sender = sender;
    this.title = title;
    this.body = body;
    Product.$init$(this);
}

}

public static class Email$ extends AbstractFunction3 implements Serializable {
    public static final ScalaFeatures.Email$ MODULE$ = new ScalaFeatures.Email$();

    public final String toString() {
        return "Email";
    }

    public ScalaFeatures.Email apply(final String sender, final String title, final String body) {
        return new ScalaFeatures.Email(sender, title, body);
    }

    public Option unapply(final ScalaFeatures.Email x$0) {

```

```

        return (Option)(x$0 == null ? scala.None..MODULE$ : new Some(new Tuple3(x$0.sender(), x$0.title(),
x$0.body())));
    }

    private Object writeReplace() {
        return new ModuleSerializationProxy(ScalaFeatures.Email$.class);
    }

    // $FF: synthetic method
    // $FF: bridge method
    public Object apply(final Object v1, final Object v2, final Object v3) {
        return this.apply((String)v1, (String)v2, (String)v3);
    }
}

public abstract static class Notification {

public static class SMS extends ScalaFeatures.Notification implements Product, Serializable {
    private final String caller;
    private final String message;

    public Iterator productElementNames() {
        return Product.productElementNames$(this);
    }

    public String caller() {
        return this.caller;
    }

    public String message() {
        return this.message;
    }

    public ScalaFeatures.SMS copy(final String caller, final String message) {
        return new ScalaFeatures.SMS(caller, message);
    }

    public String copy$default$1() {
        return this.caller();
    }
}

```

```
public String copy$default$2() {
    return this.message();
}

public String productPrefix() {
    return "SMS";
}

public int productArity() {
    return 2;
}

public Object productElement(final int x$1) {
    Object var10000;
    switch(x$1) {
        case 0:
            var10000 = this.caller();
            break;
        case 1:
            var10000 = this.message();
            break;
        default:
            var10000 = Statics.ioobe(x$1);
    }

    return var10000;
}

public Iterator productIterator() {
    return scala.runtime.ScalaRunTime..MODULE$.typedProductIterator(this);
}

public boolean canEqual(final Object x$1) {
    return x$1 instanceof ScalaFeatures.SMS;
}

public String productElementName(final int x$1) {
    String var10000;
    switch(x$1) {
        case 0:
```

```

        var10000 = "caller";
        break;
    case 1:
        var10000 = "message";
        break;
    default:
        var10000 = (String)Statics.ioobe(x$1);
    }

    return var10000;
}

public int hashCode() {
    return scala.runtime.ScalaRunTime..MODULE$._hashCode(this);
}

public String toString() {
    return scala.runtime.ScalaRunTime..MODULE$._toString(this);
}

public boolean equals(final Object x$1) {
    boolean var7;
    if (this != x$1) {
        label63: {
            boolean var2;
            if (x$1 instanceof ScalaFeatures.SMS) {
                var2 = true;
            } else {
                var2 = false;
            }

            if (var2) {
                label45: {
                    label54: {
                        ScalaFeatures.SMS var4 = (ScalaFeatures.SMS)x$1;
                        String var10000 = this.caller();
                        String var5 = var4.caller();
                        if (var10000 == null) {
                            if (var5 != null) {
                                break label54;
                            }
                        }
                    }
                }
            }
        }
    }
}

```



```

    } else if (!var10000.equals(var5)) {
        break label54;
    }

    var10000 = this.message();
    String var6 = var4.message();
    if (var10000 == null) {
        if (var6 != null) {
            break label54;
        }
    } else if (!var10000.equals(var6)) {
        break label54;
    }

    if (var4.canEqual(this)) {
        var7 = true;
        break label45;
    }
}

var7 = false;
}

if (var7) {
    break label63;
}

var7 = false;
return var7;
}

var7 = true;
return var7;
}

public SMS(final String caller, final String message) {
    this.caller = caller;
    this.message = message;
    Product.$init$(this);
}

```

```

    }
}

public static class SMS$ extends AbstractFunction2 implements Serializable {
    public static final ScalaFeatures.SMS$ MODULE$ = new ScalaFeatures.SMS$();

    public final String toString() {
        return "SMS";
    }

    public ScalaFeatures.SMS apply(final String caller, final String message) {
        return new ScalaFeatures.SMS(caller, message);
    }

    public Option unapply(final ScalaFeatures.SMS x$0) {
        return (Option)(x$0 == null ? scala.None..MODULE$ : new Some(new Tuple2(x$0.caller(), x$0.message())));
    }

    private Object writeReplace() {
        return new ModuleSerializationProxy(ScalaFeatures.SMS$.class);
    }

    // $FF: synthetic method
    // $FF: bridge method
    public Object apply(final Object v1, final Object v2) {
        return this.apply((String)v1, (String)v2);
    }
}

public static class VoiceRecording extends ScalaFeatures.Notification implements Product, Serializable {
    private final String contactName;
    private final String link;

    public Iterator productElementNames() {
        return Product.productElementNames$(this);
    }

    public String contactName() {
        return this.contactName;
    }
}

```

```
public String link() {
    return this.link;
}

public ScalaFeatures.VoiceRecording copy(final String contactName, final String link) {
    return new ScalaFeatures.VoiceRecording(contactName, link);
}

public String copy$default$1() {
    return this.contactName();
}

public String copy$default$2() {
    return this.link();
}

public String productPrefix() {
    return "VoiceRecording";
}

public int productArity() {
    return 2;
}

public Object productElement(final int x$1) {
    Object var10000;
    switch(x$1) {
        case 0:
            var10000 = this.contactName();
            break;
        case 1:
            var10000 = this.link();
            break;
        default:
            var10000 = Statics.ioobe(x$1);
    }

    return var10000;
}

public Iterator productIterator() {
```

```

        return scala.runtime.ScalaRunTime..MODULE$.typedProductIterator(this);
    }

    public boolean canEqual(final Object x$1) {
        return x$1 instanceof ScalaFeatures.VoiceRecording;
    }

    public String productElementName(final int x$1) {
        String var10000;
        switch(x$1) {
            case 0:
                var10000 = "contactName";
                break;
            case 1:
                var10000 = "link";
                break;
            default:
                var10000 = (String)Statics.ioobe(x$1);
        }

        return var10000;
    }

    public int hashCode() {
        return scala.runtime.ScalaRunTime..MODULE$._hashCode(this);
    }

    public String toString() {
        return scala.runtime.ScalaRunTime..MODULE$._toString(this);
    }

    public boolean equals(final Object x$1) {
        boolean var7;
        if (this != x$1) {
            label63: {
                boolean var2;
                if (x$1 instanceof ScalaFeatures.VoiceRecording) {
                    var2 = true;
                } else {
                    var2 = false;
                }
            }
        }
    }

```

```

if (var2) {
  label45: {
    label54: {
      ScalaFeatures.VoiceRecording var4 = (ScalaFeatures.VoiceRecording)x$1;
      String var10000 = this.contactName();
      String var5 = var4.contactName();
      if (var10000 == null) {
        if (var5 != null) {
          break label54;
        }
      } else if (!var10000.equals(var5)) {
        break label54;
      }

      var10000 = this.link();
      String var6 = var4.link();
      if (var10000 == null) {
        if (var6 != null) {
          break label54;
        }
      } else if (!var10000.equals(var6)) {
        break label54;
      }

      if (var4.canEqual(this)) {
        var7 = true;
        break label45;
      }
    }

    var7 = false;
  }

  if (var7) {
    break label63;
  }
}

var7 = false;
return var7;

```

```

    }
}

var7 = true;
return var7;
}

public VoiceRecording(final String contactName, final String link) {
    this.contactName = contactName;
    this.link = link;
    Product.$init$(this);
}
}

public static class VoiceRecording$ extends AbstractFunction2 implements Serializable {
    public static final ScalaFeatures.VoiceRecording$ MODULE$ = new ScalaFeatures.VoiceRecording$();

    public final String toString() {
        return "VoiceRecording";
    }

    public ScalaFeatures.VoiceRecording apply(final String contactName, final String link) {
        return new ScalaFeatures.VoiceRecording(contactName, link);
    }

    public Option unapply(final ScalaFeatures.VoiceRecording x$0) {
        return (Option)(x$0 == null ? scala.None..MODULE$ : new Some(new Tuple2(x$0.contactName(), x$0.link())));
    }

    private Object writeReplace() {
        return new ModuleSerializationProxy(ScalaFeatures.VoiceRecording$.class);
    }

    // $FF: synthetic method
    // $FF: bridge method
    public Object apply(final Object v1, final Object v2) {
        return this.apply((String)v1, (String)v2);
    }
}
}

```

**3. Написать алгоритм обхода графа (DFS и BFS) на языке Java, собрать в пакет и опубликовать (хоть в Maven, хоть в Gradle, не имеет значения). Использовать в другом проекте на Java/Scala этот пакет. Повторить это с C#/F#. В отчёте написать про алгоритм работы пакетных менеджеров, особенности их работы в C# и Java мирах.**

**Чтобы собрать пакет C# в Visual Studio, нужно:**

1. Щёлкнуть правой кнопкой мыши по проекту и выбрать Pack, это создаст NuGet-пакет в одной из папок с проектом
2. Зайти в эту папку в командной строке и выполнить команду:

```
nuget add CsDfsAndBfs.1.0.0.nupkg -source D:\\MyNuGetPackages
```

где CsDfsAndBfs.1.0.0.nupkg – название пакета (при создании можно указать версию в разделе Package в Properties проекта), D:\\MyNuGetPackages – папка, в которую будет опубликован NuGet-пакет

**Чтобы подключить пакет в F# проект в Visual Studio, нужно:**

1. Щёлкнуть правой кнопкой мыши по проекту и выбрать Manage NuGet Packages...
2. В настройках Package source добавить папку, в которую был опубликован проект, и выбрать её
3. Установить как обычный NuGet-пакет

**Вызов функций из пакета в коде на F#:**

```
let a = CsDfsAndBfs.Entities.Graph<int>(10)
a.AddEdge(0, 1)
a.AddEdge(1, 2)
```

```
a.ChangeVertex(0, 10)
a.ChangeVertex(1, 20)
a.ChangeVertex(2, 50)
let twiceOp = CsDfsAndBfs.Tools.TwiceOperation()
let newGraph = CsDfsAndBfs.Services.Algorithms.Bfs(a, 0, twiceOp)
let outputOp = CsDfsAndBfs.Tools.OutputOperation()
CsDfsAndBfs.Services.Algorithms.Dfs(newGraph, 0, outputOp)
```

#### Чтобы собрать пакет Java, нужно:

1. Создать проект на Maven, указать имя организации, артефакта и версии в pom.xml:

```
<groupId>dfs-and-bfs1</groupId>
<artifactId>graphs</artifactId>
<version>0.0.1-SNAPSHOT</version>
```

2. Зайти в папку с проектом в командной строке и выполнить команду:

```
mvn install
```

Это опубликует jar-файл в локальную папку .m2\repository в папке пользователя

#### Чтобы подключить пакет Java, нужно:

1. Создать проект на Maven, добавить зависимость с теми же именем организации, артефакта и версии, что и у собранного пакета:

```
<dependency>
<groupId>dfs-and-bfs1</groupId>
<artifactId>graphs</artifactId>
<version>0.0.1-SNAPSHOT</version>
</dependency>
```

2. Использовать как обычный пакет, при компиляции Maven подгрузит всё необходимое



**4. Изучить инструменты для оценки производительности в C# и Java. Написать несколько алгоритмов сортировок (и взять стандартную) и запустить бенчмарки (в бенчмарках помимо времени выполнения проверить аллокации памяти). В отчёт написать про инструменты для бенчмаркинга, их особенности, анализ результатов проверок.**

Для бенчмарков было выбрано 5 сортировок: Stooge Sort ( $O(n^{\log_{1.5} 3})$ ), Insertion sort ( $O(n^2)$  в худшем случае), Merge Sort, Quick Sort и сортировка по умолчанию (Default Sort) ( $O(n \log n)$ )

#### Java:

Для бенчмарков был использован Java Microbenchmark Harness

**Код класса с методами для бенчмаркинга:**

```
@BenchmarkMode({Mode.ALL})
@OutputTimeUnit(TimeUnit.MICROSECONDS)
@State(Scope.Thread)
public class SortingBenchmark {
    private ArrayList<Integer> _array;

    public SortingBenchmark()
    {
        _array = new ArrayList<Integer>();
    }
    @Benchmark
    public ArrayList<Integer> StoogeSortBenchmark()
    {
        GenerateRandomSequence();
        return SortingAlgorithms.StoogeSort(_array, 0, _array.size() - 1, new LessComparator());
    }
}
```

```

@Benchmark
public ArrayList<Integer> InsertionSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.InsertionSort(_array, 0, _array.size() - 1, new LessComparator());
}
@Benchmark
public ArrayList<Integer> MergeSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.MergeSort(_array, 0, _array.size() - 1, new LessComparator());
}
@Benchmark
public ArrayList<Integer> QuickSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.QuickSort(_array, 0, _array.size() - 1, new LessComparator());
}
@Benchmark
public ArrayList<Integer> DefaultSortBenchmark()
{
    GenerateRandomSequence();
    Collections.sort(_array);
    return _array;
}
private void GenerateRandomSequence()
{
    _array.clear();
    for (int i = 0; i < 1000; ++i)
    {
        _array.add(new Random().nextInt());
    }
}
}

```

Методы, для которых будут запускаться бенчмарки помечаются аннотацией @Benchmark  
 Аннотация @BenchmarkMode используется для выбора режима бенчмарков

### Код для запуска (в функции main):

```
Options options = new OptionsBuilder()
    .include(SortingBenchmark.class.getSimpleName())
    .forks(1)
    .warmupIterations(0)
    .measurementIterations(7)
    .build();
new Runner(options).run();
```

forks – количество потоков

warmupIterations – количество итераций бенчмарка для «прогрева» (несколько запусков для проведения оптимизаций, например, кэширования)

measurementIterations – количество итераций для замеров

### Результаты бенчмаркинга:

Benchmark	Mode	Cnt	Score	Error	Units
SortingBenchmark.DefaultSortBenchmark	thrpt	7	0,005 ±	0,001	ops/us
SortingBenchmark.InsertionSortBenchmark	thrpt	7	0,001 ±	0,001	ops/us
SortingBenchmark.MergeSortBenchmark	thrpt	7	0,005 ±	0,001	ops/us
SortingBenchmark.QuickSortBenchmark	thrpt	7	0,007 ±	0,001	ops/us
SortingBenchmark.StoogeSortBenchmark	thrpt	7	$\approx 10^{-5}$		ops/us
SortingBenchmark.DefaultSortBenchmark	avgt	7	213,676 ±	7,230	us/op
SortingBenchmark.InsertionSortBenchmark	avgt	7	1994,341 ±	1446,027	us/op
SortingBenchmark.MergeSortBenchmark	avgt	7	211,149 ±	8,974	us/op
SortingBenchmark.QuickSortBenchmark	avgt	7	138,089 ±	6,078	us/op
SortingBenchmark.StoogeSortBenchmark	avgt	7	259779,742 ±	8131,998	us/op
SortingBenchmark.DefaultSortBenchmark	sample	332358	210,715 ±	0,535	us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.00	sample		178,688		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.50	sample		195,584		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.90	sample		208,128		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.95	sample		255,488		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.99	sample		559,104		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.999	sample		859,792		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p0.9999	sample		2326,024		us/op
SortingBenchmark.DefaultSortBenchmark:DefaultSortBenchmark.p1.00	sample		32538,624		us/op
SortingBenchmark.InsertionSortBenchmark	sample	38160	1834,677 ±	13,273	us/op

SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.00	sample		1185,792		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.50	sample		1339,392		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.90	sample		2695,168		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.95	sample		2867,200		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.99	sample		4210,688		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.999	sample		7428,825		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p0.9999	sample		9073,530		us/op
SortingBenchmark.InsertionSortBenchmark:InsertionSortBenchmark.p1.00	sample		17924,096		us/op
SortingBenchmark.MergeSortBenchmark	sample	318856	219,504 ±	0,477	us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.00	sample		180,224		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.50	sample		199,680		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.90	sample		227,840		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.95	sample		335,949		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.99	sample		571,392		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.999	sample		889,125		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p0.9999	sample		1871,346		us/op
SortingBenchmark.MergeSortBenchmark:MergeSortBenchmark.p1.00	sample		6889,472		us/op
SortingBenchmark.QuickSortBenchmark	sample	489506	142,923 ±	0,280	us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.00	sample		116,352		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.50	sample		130,304		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.90	sample		141,824		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.95	sample		268,800		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.99	sample		374,784		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.999	sample		657,913		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p0.9999	sample		1409,702		us/op
SortingBenchmark.QuickSortBenchmark:QuickSortBenchmark.p1.00	sample		5169,152		us/op
SortingBenchmark.StoogeSortBenchmark	sample	276	256109,939 ±	3516,741	us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.00	sample		236453,888		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.50	sample		250085,376		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.90	sample		274884,198		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.95	sample		284321,382		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.99	sample		337840,701		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.999	sample		390594,560		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p0.9999	sample		390594,560		us/op
SortingBenchmark.StoogeSortBenchmark:StoogeSortBenchmark.p1.00	sample		390594,560		us/op
SortingBenchmark.DefaultSortBenchmark	ss	7	1067,529 ±	1044,765	us/op
SortingBenchmark.InsertionSortBenchmark	ss	7	4096,900 ±	12228,222	us/op
SortingBenchmark.MergeSortBenchmark	ss	7	1434,100 ±	3141,102	us/op

SortingBenchmark.QuickSortBenchmark	ss	7	1205,086 ± 2946,050	us/op
SortingBenchmark.StoogeSortBenchmark	ss	7	256572,786 ± 26322,704	us/op

Режимы бенчмаркинга:

- throughput – количество раз, которое метод успевает исполниться за определённое время
- average time – среднее время исполнения метода
- sample time – непрерывный вызов методов со случайным выбором времени вызова с регуляцией частоты дискретизации
- single shot time – время одной операции с холодным стартом

По результатам бенчмаркинга (размер массива – 1000 элементов):

- throughput – самое большое количество операций у быстрой сортировки, на втором месте – стандартная и слиянием, на 3-ем – вставками, самое маленькое количество – у марионеточной
- average time – наименьшее среднее время у быстрой сортировки, второе – у сортировки слиянием, 3-е – у стандартной, 4-е – у сортировки вставками, наибольшее – у марионеточной
- sample time – наименьшее время и наибольшее количество запусков – у быстрой сортировки, на 2-м месте – стандартная, на 3-ем – слиянием, на 4-ом – вставками, наибольшее время и наименьшее количество запусков – у марионеточной
- single shot time – наименьшее время у стандартной сортировки, второе – у быстрой, 3-е – у сортировки слиянием, 4-е – у сортировки вставками, наибольшее – у марионеточной

Для данного размера массива это отражает реальность, быстрая, стандартная и сортировка слиянием конкурируют за первое место, т.к. размер массива не очень большой и количество запусков бенчмарка не очень велико, а также отсутствуют warmups. Сортировка вставками работает медленнее в несколько раз (возможно, достаточно часто выпадал массив с относительно небольшим количеством инверсий). Самой медленной является марионеточная сортировка.

**C#:**

Для бенчмарков был использован BenchmarkDotNet

**Код класса с методами для бенчмаркинга:**

```
[SimpleJob(RunStrategy.Throughput, launchCount: 10)]
[MemoryDiagnoser]
public class SortingBenchmark
{
```

```

private int[] _array;
private IComparer<int> _comparator;
public SortingBenchmark()
{
    _array = new int[1000];
    _comparator = new LessComparator();
}
[Benchmark(Description = "StoogeSortBenchmark")]
public int[] StoogeSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.StoogeSort(_array, 0, _array.Length - 1, _comparator);
}
[Benchmark(Description = "InsertionSortBenchmark")]
public int[] InsertionSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.InsertionSort(_array, 0, _array.Length - 1, _comparator);
}
[Benchmark(Description = "MergeSortBenchmark")]
public int[] MergeSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.MergeSort(_array, 0, _array.Length - 1, _comparator);
}
[Benchmark(Description = "QuickSortBenchmark")]
public int[] QuickSortBenchmark()
{
    GenerateRandomSequence();
    return SortingAlgorithms.QuickSort(_array, 0, _array.Length - 1, _comparator);
}
[Benchmark(Description = "DefaultSortBenchmark")]
public int[] DefaultSortBenchmark()
{
    GenerateRandomSequence();
    Array.Sort(_array);
    return _array;
}
private void GenerateRandomSequence()
{
    for (int i = 0; i < _array.Length; ++i)
    {

```

```

        _array[i] = new Random().Next();
    }
}
}

```

SimpleJob – атрибут для указания параметров бенчмаркинга:

- RunStrategy – режим запуска
- launchCount – количество запусков

MemoryDiagnoser – атрибут для замеров используемой памяти помимо времени

Benchmark – атрибут, помечающий методы, для которых необходимо запустить бенчмарки:

- Description – имя метода, которое будет отображаться в таблице с результатами бенчмаркинга

**Код для запуска (в функции Main):**

```
BenchmarkRunner.Run<SortingBenchmark>();
```

При такой настройке количество итераций для warmup'a рассчитывается автоматически

**Результаты бенчмаркинга:**

Method	Mean	Error	StdDev	Median	Gen 0	Allocated
StoogeSortBenchmark	260.078 ms	1.0335 ms	3.6778 ms	259.649 ms	-	276 KB
InsertionSortBenchmark	2.282 ms	0.0051 ms	0.0183 ms	2.283 ms	132.8125	273 KB
MergeSortBenchmark	1.773 ms	0.0023 ms	0.0080 ms	1.772 ms	164.0625	336 KB
QuickSortBenchmark	1.727 ms	0.0039 ms	0.0141 ms	1.722 ms	132.8125	273 KB
DefaultSortBenchmark	1.697 ms	0.0028 ms	0.0100 ms	1.694 ms	132.8125	273 KB

По времени: наименьшее время у стандартной сортировки, на 2-ом месте – быстрая, на 3-ем – сортировка слиянием, на 4-ом – сортировка вставками, наибольшее – у марионеточной

По памяти: наименьшее использование памяти у стандартной, быстрой сортировок и сортировки вставками, на 2-ом месте – марионеточная, наибольшее – у сортировки слиянием

Для данного размера массива это отражает реальность, стандартная, быстрая сортировки и сортировка слиянием делят первое место, являясь примерно одинаковыми по времени, сортировка вставками выполняется чуть дольше (достаточно хорошее время может быть

связано с тем, что в тестовых примерах могли сгенерироваться массивы с относительно небольшим количеством инверсий), наиболее медленной является марионеточная сортировка. Расход памяти у марионеточной сортировки обусловлен глубиной рекурсии, а у сортировки слиянием – копиями массива.

***5. Используя инструменты dotTrace, dotMemory, всё-что-угодно-хоть-windbg, проанализировать работу написанного кода для бекапов. Необходимо написать сценарий, когда в цикле будет выполняться много запусков, будут создаваться и удаляться точки. Проверить два сценария: с реальной работой с файловой системой и без неё. В отчёте необходимо проанализировать полученные результаты, сделать вывод о написанном коде.***

Был написан сценарий, в котором последовательно создаётся 100 точек, причём, если точек становится больше 5, то старые удаляются.

**DotTrace:**

**С файловой системой:**



▲ 29,8 % Main • 70 975 ms • BackupsExtra.**Program**.Main()

▲ 29,8 % ProcessObjects • 70 970 ms • BackupsExtra.Entities.**BackupJobDeletingRestoringDecorator**.ProcessOb

▲ 28,3 % ReduceRestorePoints • 67 405 ms • BackupsExtra.Entities.**RestorePointsDeleter**.ReduceRestorePoint

▶ 28,3 % RemoveRestorePoint • 67 402 ms • Backups.Entities.**FileSystemRepository**.RemoveRestorePoint(uint)

▶ 0,00 % coreclr.dll • 2,5 ms

▶ 0,00 % [Unknown] • 0,2 ms

▶ 0,00 % Find • 0,1 ms • System.Collections.Generic.**List`1**.Find(Predicate)

▶ 0,00 % MoveNext • 0,09 ms • System.Collections.Generic.**List`1+Enumerator**.MoveNext()

▶ 0,00 % GetEnumerator • 0,09 ms • System.Collections.Generic.**List`1**.GetEnumerator()

▶ 0,00 % get\_Repository • 0,07 ms • BackupsExtra.Entities.**BackupJobDeletingRestoringDecorator**.get\_Repo

▶ 0,00 % ReduceRestorePoints • 0,05 ms • BackupsExtra.Entities.**RestorePointsDeleter**.ReduceRestorePoints(IBa

▲ 1,48 % ProcessObjects • 3 538 ms • Backups.Entities.**BackupJob**.ProcessObjects()

▶ 1,48 % SaveRestorePoint • 3 517 ms • Backups.Entities.**FileSystemRepository**.SaveRestorePoint(RestorePoint)

▶ 0,01 % RestorePoint..ctor • 13 ms • Backups.Entities.**RestorePoint**..ctor(uint, List)

▢ 0,00 % coreclr.dll • 2,8 ms

▶ 0,00 % OrganizeJobObjects • 2,5 ms • Backups.Entities.**SingleStorageAlgorithm**.OrganizeJobObjects(List)

▶ 0,00 % RestorePointJobObjectsInfo..ctor • 1,2 ms • Backups.Entities.**RestorePointJobObjectsInfo**..ctor(uint, D

▢ 0,00 % [Unknown] • 1,0 ms

▶ 0,00 % get\_Now • 0,7 ms • System.**DateTime**.get\_Now()

▢ 0,01 % [Unknown] • 16 ms

▶ 0,00 % SelectUnsuitableRestorePointsIds • 9,3 ms • BackupsExtra.Entities.**CountRestorePointsDeletingAlgorit**

▢ 0,00 % coreclr.dll • 1,4 ms

▶ 0,00 % get\_CreatedRestorePoints • 0,08 ms • BackupsExtra.Entities.**BackupJobDeletingRestoringDecorator**.g

▢ 0,00 % coreclr.dll • 2,0 ms

▶ 0,00 % FileObject..ctor • 1,7 ms • Backups.Entities.**FileObject**..ctor(String, String, String)

▶ 0,00 % AddJobObject • 0,6 ms • BackupsExtra.Entities.**BackupJobDeletingRestoringDecorator**.AddJobObject(I

0,00 % [Unknown] • 0,3 ms

▶ 0,00 % FileSystemRepository..ctor • 0,2 ms • Backups.Entities.**FileSystemRepository**..ctor(String, IArchiver, IStora

▶ 0,02 % ProcessStartupHooks • 38 ms • System.**StartupHookProvider**.ProcessStartupHooks()

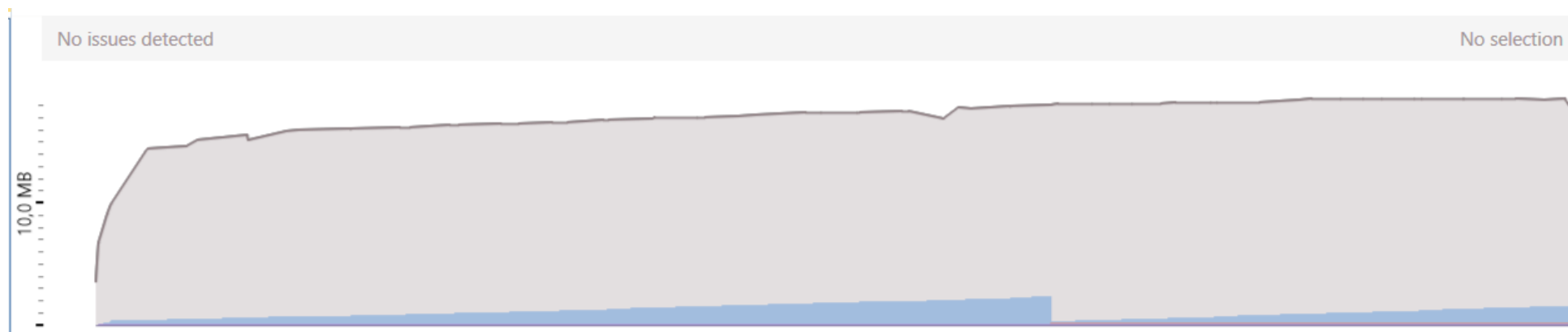
▶ 0,01 % Setup • 31 ms • System.**AppContext**.Setup(char\*\*, char\*\*, int)

▶ 0,00 % Finalize • 1,6 ms • System.**Gen2GcCallback**.Finalize()

▶ 0,00 % SetCommandLineArgs • 0,2 ms • System.**Environment**.SetCommandLineArgs(String[])

▶ 0,00 % Finalize • 0,1 ms • System.Runtime.InteropServices.**SafeHandle**.Finalize()

Большую часть времени занимает удаление объектов (возможно, это связано с перегруженным состоянием корзины)

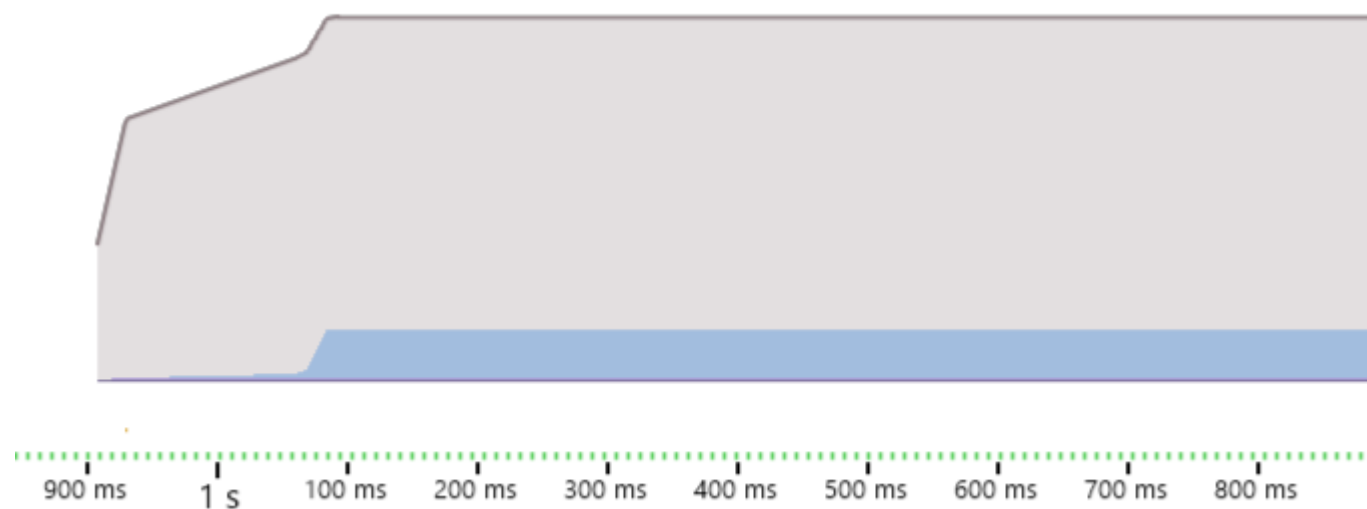


Использование памяти равномерно на протяжении работы всего сценария. Видно, как в начале увеличивается неуправляемая память, затем точки начинают удаляться. Объекты на куче также успешно удаляются сборщиком мусора.

## Без файловой системы:

▲ 7,00 %	Main	• 79 ms	• BackupsExtra. <b>Program</b> .Main()
▲ 6,39 %	ProcessObjects	• 72 ms	• BackupsExtra.Entities. <b>BackupJobDeletingRestoringDecorator</b> .ProcessObjects()
▲ 4,96 %	ProcessObjects	• 56 ms	• Backups.Entities. <b>BackupJob</b> .ProcessObjects()
▶ 3,87 %	SaveRestorePoint	• 44 ms	• Backups.Entities. <b>MemoryRepository</b> .SaveRestorePoint(RestorePoint)
▶ 0,85 %	RestorePoint..ctor	• 9,5 ms	• Backups.Entities. <b>RestorePoint</b> ..ctor(uint, List)
▶ 0,08 %	OrganizeJobObjects	• 0,9 ms	• Backups.Entities. <b>SingleStorageAlgorithm</b> .OrganizeJobObjects(List)
☹ 0,07 %	coreclr.dll	• 0,8 ms	
▶ 0,05 %	RestorePointJobObjectsInfo..ctor	• 0,6 ms	• Backups.Entities. <b>RestorePointJobObjectsInfo</b> ..ctor(uint, Da
☹ 0,02 %	[Unknown]	• 0,3 ms	
▶ 0,02 %	Add	• 0,2 ms	• System.Collections.Generic. <b>List`1</b> .Add(!0)
▶ 0,81 %	SelectUnsuitableRestorePointIds	• 9,2 ms	• BackupsExtra.Entities. <b>CountRestorePointsDeletingAlgorith</b>
▶ 0,40 %	ReduceRestorePoints	• 4,5 ms	• BackupsExtra.Entities. <b>RestorePointsDeleter</b> .ReduceRestorePoints(!Bac
☹ 0,11 %	coreclr.dll	• 1,2 ms	
☹ 0,10 %	[Unknown]	• 1,1 ms	
▶ 0,01 %	get_CreatedRestorePoints	• 0,06 ms	• BackupsExtra.Entities. <b>BackupJobDeletingRestoringDecorator</b> .ge
▶ 0,29 %	FileObject..ctor	• 3,3 ms	• Backups.Entities. <b>FileObject</b> ..ctor(String, String, String)
☹ 0,20 %	coreclr.dll	• 2,2 ms	
▶ 0,06 %	AddJobC	• C:\Program Files\dotnet\shared\Microsoft.NETCore.App\5.0.4\coreclr.dll	• orator.AddJobObject(!Jc
▶ 0,03 %	MemoryRepository..ctor	• 0,4 ms	• Backups.Entities. <b>MemoryRepository</b> ..ctor()
0,02 %	[Unknown]	• 0,3 ms	
▶ 0,01 %	BackupJoba..ctor	• 0,08 ms	• Backups.Entities. <b>BackupJoba</b> ..ctor(String, IStorageAlgorithm, IRepository)
▶ 2,62 %	ProcessStartupHooks	• 30 ms	• System. <b>StartupHookProvider</b> .ProcessStartupHooks()
▶ 2,07 %	Setup	• 23 ms	• System. <b>AppContext</b> .Setup(char**, char**, int)
▶ 0,01 %	SetCommandLineArgs	• 0,1 ms	• System. <b>Environment</b> .SetCommandLineArgs(String[])

Большую часть времени занимает сохранение точки, также часть времени занимает поиск и удаление устаревших точек.



Использование памяти равномерно на протяжении работы всего сценария. Видно, как в начале увеличивается выделенная память на куче, а затем точки начинают удаляться.