МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное автономное образовательное учреждение

высшего образования

**«Национальный исследовательский**

**Нижегородский государственный университет им. Н.И. Лобачевского»**

**Институт информационных технологий, математики и механики**

Направление подготовки «Фундаментальная информатика и информационные технологии»

**ОТЧЕТ**

по лабораторной работе

**Практическое задание для лекции №8**

**Выполнил:** студент группы

381906-2 Углинский Б.С

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Подпись

**Проверил:** Карчков Д.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Подпись

Нижний Новгород

2022

Оглавление

[1. Цели работы. 3](#_Toc121059398)

[2. Постановка задачи. 4](#_Toc121059399)

[3. Решение задачи. 5](#_Toc121059400)

[4. Руководство пользователя. 6](#_Toc121059401)

[Приложение. 7](#_Toc121059402)

# 1. Цели работы.

Целью данной лабораторной работы является освоение использования C++ классов в Android, а также возможности написания собственных компонент на языке C++.

.

# 2. Постановка задачи.

Для выполнения лабораторной работы были поставлены следующие задачи:

1. Создать страницу с текстовым полем и двумя кнопками. Текстовое поле должно отображать текущее значения счётчика, кнопки используются для увеличения значения счётчика на единицу и сброса значения до нуля. Использовать при создании класс-счётчик, написанный на C++.
2. Сделать класс, содержащий список из строк. Этот класс должен уметь добавлять строки в список и удалять последнюю добавленную строку. Все строки хранятся в маленьких буквах.
3. Сделать страницу, на которой будут располагаться поле для ввода текста записи и две кнопки. Одна кнопка добавляет слово в список, другая удаляет последнее добавленное слово из списка. Внизу должны быть перечислены все слова списка через запятую с маленькой буквы. Первое слово всегда с большой.

# 3. Решение задачи.

1. Для того, чтобы воспользоваться С++ кодом, при разработке на Java под Android, необходимо проделать следующие шаги:
   1. Нажмите на поле для создания нового проекта.
   2. В шаблонах создания проекта выберите Native C++.
   3. Выберите стандарт C++.
   4. Нажмите ПКМ на папке cpp и выберите в выпадающем меню C\C++ Source File.
   5. Дайте название файлу и выберите расширения файла (.cpp или .c).
   6. Проведите конфигурацию CMakeLists.txt (добавьте название своего файла), чтобы изменения вступили в силу.
   7. В основной активности не забудьте подключить скомпилированную библиотеку следующим образом:

static {

System.loadLibrary(“название вашей библиотеки”)

}.

* 1. Чтобы использовать функции из C++ необходимо объявить их в классе активности следующим образом: public native <прототип функции>.
  2. Теперь можно использовать объявленные функции.

1. Реализация счётчика достаточно проста: Делаем две кнопки, которые по нажатию будут вызывать нативный код из с++ на увеличения счётчика и на его сброс. Сам C++ класс состоит из одного целочисленного поля и 3 методов: getCounter(), increaseCounter() и resetCounter().
2. Для создания списка из строк, достаточно подключить в C++ файле библиотеки <vector>, <algorithm> и <string>. После чего написать логику поведения класса на добавления новой строки и удаление последней строки

# 4. Руководство пользователя.

Приложения состоит из двух страниц.

На первой расположен счётчик с двумя кнопками (“Увеличить” и ”Сбросить”)

  

На второй странице вверху экрана находится строка для ввода новой записи. Под ней две кнопки. Одна добавляет новое слово в список из строк, другая удаляет из списка последнее добавленное слово. В самом низу расположены все строки в списке.

 

# Приложение.

* MainActivity.java

package com.example.lab8;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 static {  
 System.*loadLibrary*("a08\_logic\_integration");  
 }  
  
 public native static long createCounter();  
  
 public native static int getCounter(long nativePointer);  
  
 public native static long increaseCounter(long nativePointer);  
  
 public native static long resetCounter(long nativePointer);  
  
 @Override  
 protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 JCounter counter = new JCounter();  
 TextView tCounter = findViewById(R.id.Counter);  
 tCounter.setText(Integer.*toString*(counter.JGetCounter()));  
 Button bIncrease = findViewById(R.id.Increase);  
 Button bReset = findViewById(R.id.Reset);  
  
 bIncrease.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 counter.JIncreaseCounter();  
 tCounter.setText(Integer.*toString*(counter.JGetCounter()));  
 }  
 });  
  
 bReset.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 counter.JResetCounter();  
 tCounter.setText(Integer.*toString*(counter.JGetCounter()));  
 }  
 });  
 }  
  
 public void NextActivity(View v) {  
 startActivity(new Intent(this, StringListActivity.class));  
 }  
  
 public class JCounter {  
 private long nativePointer;  
  
 public JCounter() {  
 nativePointer = *createCounter*();  
 }  
  
 public int JGetCounter() {  
 return *getCounter*(nativePointer);  
 }  
  
 public void JIncreaseCounter() {  
 nativePointer = *increaseCounter*(nativePointer);  
 }  
  
 public void JResetCounter() {  
 nativePointer = *resetCounter*(nativePointer);  
 }  
 }  
}

* activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/Counter"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="0"  
 android:textColor="@color/black"  
 android:textSize="30sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.218" />  
  
  
 <Button  
 android:id="@+id/Increase"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:width="320dp"  
 android:height="60dp"  
 android:text="Увеличить"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.494"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.454" />  
  
 <Button  
 android:id="@+id/Reset"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:width="320dp"  
 android:height="60dp"  
 android:text="Сбросить"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.494"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Increase"  
 app:layout\_constraintVertical\_bias="0.045" />  
  
 <Button  
 android:id="@+id/NextPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="28dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="NextActivity"  
 android:text="Next"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Lab8"  
 android:textColor="@color/black"  
 android:textSize="40dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.023" />  
  
 <TextView  
 android:id="@+id/task\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="28dp"  
 android:text="1"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

* StringListActivity.java

package com.example.lab8;  
  
import android.content.Context;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ListView;  
import android.widget.TextView;  
  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
import java.util.ArrayList;  
  
public class StringListActivity extends AppCompatActivity {  
  
 public JStringList list = new JStringList();  
  
 public native static long createStringList();  
  
 public native static long pushStringList(String element, long nativePointer);  
  
 public native static long popStringList(long nativePointer);  
  
 public native static String getStringList(int position, long nativePointer);  
  
 public native static String getAllStringList(long nativePointer);  
  
 public native static boolean getIsAdded(long nativePointer);  
  
 @Override  
 protected void onCreate(@Nullable Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_string\_list);  
  
 ListView listView = findViewById(R.id.listView);  
 ArrayList Notes = new ArrayList<String>();  
  
 Notes.add("");  
 list.JPush("Яблоко");  
 ListAdapter adapter = new ListAdapter(this, R.layout.list\_items, Notes);  
 listView.setAdapter(adapter);  
 EditText Item = findViewById(R.id.TypeNote);  
 Item.bringToFront();  
 TextView listOneString = findViewById(R.id.ListOneString);  
 listOneString.setText(list.JGetALL());  
  
 Button bAddNewItem = findViewById(R.id.AddItem);  
 bAddNewItem.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String str = String.valueOf(Item.getText());  
 list.JPush(str);  
 // bAddNewItem.setText(String.valueOf(list.JGetIsAdded()));  
 if (list.JGetIsAdded()) {  
 Notes.add(str);  
 listOneString.setText(list.JGetALL());  
 adapter.notifyDataSetChanged();  
 }  
 }  
 });  
 Button bDeleteLastItem = findViewById(R.id.DeleteLastItem);  
 bDeleteLastItem.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 if (Notes.size() != 0) {  
 Notes.remove(Notes.size() - 1);  
 list.JPop();  
 listOneString.setText(list.JGetALL());  
 adapter.notifyDataSetChanged();  
 }  
 }  
 });  
 }  
  
 public void PrevActivity(View v) {  
 startActivity(new Intent(this, MainActivity.class));  
 }  
  
 public class JStringList {  
 private long nativePointer;  
  
 public JStringList() {  
 nativePointer = createStringList();  
 }  
  
 public void JPush(String element) {  
 nativePointer = pushStringList(element, nativePointer);  
 }  
  
 public void JPop() {  
 nativePointer = popStringList(nativePointer);  
 }  
  
 public String JGet(int position) {  
 return getStringList(position, nativePointer);  
 }  
  
 public String JGetALL() {  
 return getAllStringList(nativePointer);  
 }  
  
 public boolean JGetIsAdded() {  
 return getIsAdded(nativePointer);  
 }  
 }  
  
 public class ListAdapter extends ArrayAdapter {  
  
 private int resourceLayout;  
 private Context mContext;  
  
 public ListAdapter(Context context, int resource, ArrayList<String> tmp) {  
 super(context, resource, tmp);  
 this.resourceLayout = resource;  
 this.mContext = context;  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
  
 View v = convertView;  
  
 if (v == null) {  
 LayoutInflater vi;  
 vi = LayoutInflater.from(mContext);  
 v = vi.inflate(resourceLayout, null);  
 }  
 TextView item = v.findViewById(R.id.Item);  
 item.setText(list.JGet(position).toLowerCase());  
  
 return v;  
 }  
 }  
}

* activity\_string\_list.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".StringListActivity">  
  
 <TextView  
 android:id="@+id/task\_txt2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="32dp"  
 android:text="2"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.506"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
 <Button  
 android:id="@+id/previousPage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginBottom="16dp"  
 android:onClick="PrevActivity"  
 android:text="prev"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
 <EditText  
 android:id="@+id/TypeNote"  
 android:layout\_width="330dp"  
 android:layout\_height="49dp"  
 android:hint="Введите слово"  
 android:inputType="textPersonName"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.493"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.093" />  
  
  
 <Button  
 android:id="@+id/AddItem"  
 android:layout\_width="160dp"  
 android:layout\_height="100dp"  
 android:width="180dp"  
 android:height="60dp"  
 android:text="Добавить новое слово"  
 android:textSize="15sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.087"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.209" />  
  
 <Button  
 android:id="@+id/DeleteLastItem"  
 android:layout\_width="170dp"  
 android:layout\_height="100dp"  
 android:width="250dp"  
 android:height="60dp"  
 android:text="Удалить последнее слово"  
 android:textSize="15sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.372"  
 app:layout\_constraintStart\_toEndOf="@+id/AddItem"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.209" />  
  
 <ListView  
 android:id="@+id/listView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="320dp"  
 android:paddingTop="10dp"  
 android:paddingBottom="10dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/AddItem"  
 app:layout\_constraintVertical\_bias="0.044"  
 tools:layout\_editor\_absoluteX="0dp"></ListView>  
  
 <TextView  
 android:id="@+id/ListOneString"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="TextView"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.066"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/listView"  
 app:layout\_constraintVertical\_bias="0.131" />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

* list\_item.xml

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/Rectangle"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:paddingTop="1dip"  
 android:paddingBottom="1dip"  
 android:paddingStart="8dip"  
 android:paddingEnd="8dip"  
 android:backgroundTint="@null"  
 android:background="@color/white">  
  
 <TextView android:id="@+id/Item"  
 android:layout\_width="0sp"  
 android:layout\_weight="0.5"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:paddingStart="?android:attr/listPreferredItemPaddingStart"  
 android:textColor="@color/black"  
 android:textSize="20sp"  
 android:gravity="center"  
 android:paddingBottom="10sp"  
 android:paddingTop="10sp"/>  
</LinearLayout>

* classes.cpp

#include <jni.h>  
  
// Write C++ code here.  
//  
// Do not forget to dynamically load the C++ library into your application.  
//  
// For instance,  
//  
// In MainActivity.java:  
// static {  
// System.loadLibrary("a08\_logic\_integration");  
// }  
//  
// Or, in MainActivity.kt:  
// companion object {  
// init {  
// System.loadLibrary("a08\_logic\_integration")  
// }  
// }  
  
#include <iostream>  
#include <vector>  
#include <algorithm>  
#include <string>  
  
class Counter {  
private:  
 int counter;  
public:  
 Counter() {  
 counter = 58;  
 };  
  
 Counter(int local\_counter) {  
 counter = local\_counter;  
 };  
  
 int getCounter() {  
 return counter;  
 }  
  
 void increaseCounter() {  
 counter++;  
 }  
  
 void resetCounter() {  
 counter = 0;  
 }  
};  
  
  
class StringList {  
private:  
 std::vector<std::string> list;  
 std::string result;  
 bool is\_added;  
  
public:  
 StringList() = default;  
  
 bool push(const std::string &element) {  
 std::string toup;  
 std::string tolow;  
 std::string toad = "";  
 if (element.size() != 0) {  
 toup = element.substr(0, 1);  
 for (auto &c: toup) toad += toupper(c);  
 tolow = element.substr(1, element.size() - 1);  
 for (auto &c: tolow) toad += tolower(c);  
 }  
   
 {  
 list.emplace\_back(toad);  
  
 if (list.size() == 1) {  
 if (list[0].size() != 0) {  
 toup = list[0].substr(0, 1);  
 for (auto &c: toup) result += toupper(c);  
 tolow = list[0].substr(1, list[0].size() - 1);  
 for (auto &c: tolow) result += tolower(c);  
 }  
 } else {  
 tolow = element.substr(0, element.size());  
 result += ", ";  
 for (auto &c: tolow) result += tolower(c);  
 }  
 is\_added = true;  
 return true;  
 }  
 is\_added = false;  
 return false;  
 }  
  
 void pop() {  
 if (list.size() == 1) {  
 result = "";  
 } else {  
 result = result.substr(0, result.size() - list[list.size() - 1].size() - 2);  
 }  
 list.pop\_back();  
 }  
  
 const char \*get(int position) {  
 return list[position].c\_str();  
 }  
  
 const char \*getAll() {  
 return result.c\_str();  
 }  
  
 const bool getIsAdded() {  
 return is\_added;  
 }  
};  
  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_MainActivity\_createCounter(JNIEnv \*env, jclass clazz) {  
 return (jlong) (new Counter());  
}  
extern "C"  
JNIEXPORT jint JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_MainActivity\_getCounter(JNIEnv \*env, jclass clazz,  
 jlong nativePointer) {  
 return ((Counter \*) nativePointer)->getCounter();  
}  
  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_MainActivity\_increaseCounter(JNIEnv \*env, jclass clazz,  
 jlong nativePointer) {  
 Counter \*pointer = ((Counter \*) nativePointer);  
 pointer->increaseCounter();  
 return reinterpret\_cast<jlong>(pointer);  
}  
  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_MainActivity\_resetCounter(JNIEnv \*env, jclass clazz,  
 jlong nativePointer) {  
 Counter \*pointer = ((Counter \*) nativePointer);  
 pointer->resetCounter();  
 return reinterpret\_cast<jlong>(pointer);  
}  
  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_createStringList(JNIEnv \*env,  
 jclass clazz) {  
 return (jlong) (new StringList());  
}  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_pushStringList(JNIEnv \*env,  
 jclass clazz,  
 jstring element,  
 jlong nativePointer) {  
 StringList \*pointer = ((StringList \*) nativePointer);  
 pointer->push(env->GetStringUTFChars(element, nullptr));  
 return reinterpret\_cast<jlong> (pointer);  
}  
extern "C"  
JNIEXPORT jlong JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_popStringList(JNIEnv \*env, jclass clazz,  
 jlong nativePointer) {  
 StringList \*pointer = ((StringList \*) nativePointer);  
 pointer->pop();  
 return reinterpret\_cast<jlong>(pointer);  
}  
extern "C"  
JNIEXPORT jstring JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_getStringList(JNIEnv \*env, jclass clazz,  
 jint position,  
 jlong nativePointer) {  
 return env->NewStringUTF((((StringList \*) nativePointer)->get(position)));  
}  
extern "C"  
JNIEXPORT jstring JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_getAllStringList(JNIEnv \*env,  
 jclass clazz,  
 jlong nativePointer) {  
 return env->NewStringUTF((((StringList \*) nativePointer)->getAll()));  
}  
extern "C"  
JNIEXPORT jboolean JNICALL  
Java\_com\_example\_a08\_1logic\_1integration\_StringListActivity\_getIsAdded(JNIEnv \*env, jclass clazz,  
 jlong nativePointer) {  
 return ((StringList \*) nativePointer)->getIsAdded();  
}