
notes

fangjun

Mar 21, 2024

CONTENTS:

1	C++	3
1.1	cppcon	3
1.2	Meeting C++	3
1.3	getline	3
1.4	ratio	4
1.5	chrono	5
2	Sphinx	11
2.1	Setup	11
2.2	How to include code from a file	12
2.3	Link	12
2.3.1	hello	13
3	git	15
3.1	Branch	15
3.2	Commands	15
3.2.1	rev-parse	15
4	docker	17
4.1	Installation	17
4.1.1	macos	17
4.1.2	Change directory to save images	17
4.1.3	Run an image	17
4.1.4	copy data between container and host	18
5	LaTeX	19
5.1	TikZ	19
5.1.1	Basics	19
6	Kaldi	21
6.1	Decoding	21
7	bash	23
7.1	sort	23
7.2	echo	23
7.3	ffmpeg	23
7.3.1	Convert format	24
7.3.2	References	24
7.4	wget	24
7.4.1	cookies	24
7.5	Hello	24

7.6	pkg-config	24
8	CUDA	25
8.1	Installation	25
8.1.1	CUDA 10.1.243	25
8.1.2	CUDA 10.2.89	25
8.1.3	CUDA 11.0.3	26
8.1.4	CUDA 11.3.1	26
8.1.5	CUDA 11.5.2	27
8.1.6	CUDA 11.6.1	27
8.1.7	CUDA 11.7.1	27
9	torch	29
9.1	torch.load and torch.save	29
9.2	torch.gather	29
9.3	DDP	30
9.3.1	Initialization	30
9.4	TorchScript	30
9.4.1	doxygen doc	30
9.4.2	Hello	31
9.4.3	Load in C++	37
9.4.4	ArrayRef	39
9.4.5	ScalarType	41
9.4.6	TypeMeta	42
9.4.7	torch::Device	43
9.4.8	TensorOptions	45
9.4.9	Tensor Creation	46
9.4.10	Tensor	48
9.4.11	intrusive_ptr	60
9.4.12	optional	60
9.4.13	PackedSequence	60
9.4.14	ivalue	61
9.4.15	method	65
9.4.16	type	65
9.4.17	trace	67
9.4.18	Node	71
9.4.19	symbol	73
9.4.20	graph	75
9.4.21	passes	77
9.4.22	ops	81
9.5	Logical operations	82
9.6	Note	82
9.7	Quantization	82
9.7.1	Internals	82
9.7.2	torch.quantize_per_tensor	84
9.7.3	quantize_per_tensor_dynamic	85
9.7.4	torch.quantize_per_channel	87
9.7.5	Observer	88
9.7.6	Hello	89
9.7.7	References	90
9.8	android	90
9.8.1	References	90
9.9	onnx	91
9.9.1	Install	91

9.9.2	Hello	91
9.9.3	Multiple models	93
9.9.4	References	95
9.10	nn.LSTM	96
10	Python	99
10.1	Install from Source	99
10.2	asyncio	99
10.2.1	asyncio.Future	99
10.2.2	iterator	99
10.2.3	yield	99
10.2.4	Hello World	99
10.2.5	References	100
10.2.6	TODOs	101
10.3	argv	101
10.4	TODO	101
10.5	time	101
10.6	Numbers	101
10.6.1	binary representation	101
10.7	str	102
10.7.1	format	102
10.8	enum	103
10.8.1	Hello	103
10.9	socket	105
10.9.1	AddressFamily	105
10.9.2	SocketKind	106
10.9.3	struct sockaddr_in	107
10.9.4	AddressInfo	108
10.9.5	inet_pton	108
10.9.6	inet_ntop	113
10.9.7	Echo Server and Client	117
10.9.8	TODOs	120
10.10	numpy	120
10.10.1	Basics	120
10.11	pip	122
11	java	123
11.1	Install	123
11.1.1	formatter	123
11.1.2	JDK	123
11.2	Hello world	124
11.2.1	TODOs	128
11.3	Reference	128
12	javascript	129
12.1	Hello world	129
12.1.1	variables	130
12.1.2	array	131
12.1.3	class	131
12.1.4	template strings	132
12.1.5	functions	132
12.2	Numbers in javascript	133
12.3	Boolean in javascript	135
12.4	Strings in javascript	135

12.5	array in javascript	137
12.6	typed array in javascript	139
12.7	Regular expressions in javascript	141
12.8	node	141
12.9	TODOs	142
12.10	npm	142
12.10.1	TODO	142
12.11	module	143
12.11.1	CommonJS modules	143
12.11.2	ESmodules	143
12.11.3	webpack	145
13	HTML	147
13.1	Hello world	147
13.1.1	comments	147
13.1.2	images	147
13.1.3	ordered lists	147
13.1.4	unordered lists	148
13.1.5	links	148
13.1.6	div	148
13.2	References	149
14	css	151
14.1	Hello world	151
14.1.1	comment	151
14.1.2	Selector	151
14.2	References	152
15	pybind11	153
15.1	GIL	153
16	Protocol Buffers	155
16.1	Installation	155
16.1.1	C++	155
16.1.2	Install with cmake	164
16.2	Hello	165
16.2.1	hello.proto	165
16.2.2	makefile	165
16.2.3	hello.pb.h	166
16.2.4	hello.pb.cc	187
17	gRPC	207
17.1	Install	207
18	lwn.net	209
18.1	TODOs	209
19	Linker and Loader	211
19.1	References	211
19.2	Questions	212
20	espnet	213
20.1	aishell	213
20.1.1	AM training	213

21	cmake	215
21.1	Tutorials	215
21.2	Install	215
22	huggingface	217
22.1	spaces	217
22.1.1	Install client API	217
22.1.2	gradio	217
23	EECS E6870 Speech Recognition	219
23.1	Notes	219
24	ncnn	221
24.1	Hello	221
24.1.1	test-net	222
24.1.2	test-param-dict	225
24.1.3	test-data-reader	226
25	LLVM	229
25.1	Installation	229
25.2	ninja	230
25.3	Documentation	230
25.4	Intermediate representation	231
25.5	Install GCC	231
25.5.1	Using conda	232
26	Android	233
26.1	kotlin	233
26.1.1	Install	233
26.1.2	Basic types	235
26.1.3	Collections	240
26.1.4	jni	241
26.1.5	Hello jni	242
26.1.6	Binding kaldi-native-fbank for kotlin	244
26.1.7	References	255
26.1.8	Control flows	255
26.1.9	Functions	258
26.1.10	Package	260
26.1.11	for	260
26.1.12	enum	260
26.2	Basics	260
26.2.1	TextView	261
26.2.2	Button	262
26.2.3	EditText	262
26.2.4	LinearLayout	262
26.3	Installation	262
26.3.1	Install NDK on Linux (not recommended)	262
26.3.2	Install SDK on Linux	263
26.3.3	emulator	264
26.4	cmake	265
26.5	hello	265
26.5.1	x86	266
26.6	Android.mk	267
26.6.1	hello	267
26.7	adb	268

26.7.1	install on macos	268
26.7.2	install on windows	268
26.7.3	install on Linux	268
26.7.4	run binaries	268
26.7.5	paste text to android emulator	268
26.8	tts	269
26.9	jetpack	269
26.9.1	material design	269
26.9.2	state	269
26.9.3	bom	270
26.9.4	Arrangement	270
26.9.5	Alignment	270
26.9.6	Surface	270
26.9.7	Preview	271
26.9.8	Column	271
26.9.9	LazyColumn	271
26.9.10	Row	271
26.9.11	LazyRow	272
26.9.12	Spacer	273
26.9.13	Text	273
26.9.14	Button	274
26.9.15	IconButton	274
26.9.16	Checkbox	274
26.9.17	Modifier	274
26.9.18	Box	275
26.9.19	Scaffold	275
26.9.20	TextField	275
26.9.21	remember	276
26.9.22	Example 1	276
26.9.23	Example 2	277
26.9.24	Example 3	278
26.9.25	Example 4	279
26.9.26	Example 5	280
26.9.27	Example 6	280
26.9.28	Example 7	280
26.10	Samples	281
27	qemu	283
27.1	Install	283
27.1.1	qemu-arm	283
27.1.2	qemu-aarch64	284
28	sox	287
28.1	Basics	287
28.1.1	Extract part of a wave	287
28.1.2	Fix broken waves	287
28.1.3	pad	287
28.1.4	raw	287
28.1.5	Generate	287
28.1.6	remix	288
28.2	base64	288
29	MNN	289
29.1	Install	289

29.1.1	Python	289
29.2	Hello	289
30	SIMD	293
30.1	References	293
30.2	Headers	293
30.3	SSE	294
30.4	avx	294
31	asio	297
31.1	Install	297
31.2	Hello	297
32	websocketpp	307
32.1	Install	307
32.2	hello	307
33	Operating systems	315
33.1	macos	315
33.1.1	Port scan	315
33.2	lldb	315
34	encoding	317
34.1	C++ String	317
34.2	kotlin String	318
34.3	Basics	319
34.4	TODOs	319
35	ios	321
35.1	Call cpp	321
35.2	Swift	322
35.2.1	Basics	322
35.2.2	Package	333
35.3	Objective C	336
35.3.1	Basics	336
36	Embedded systems	339
36.1	m3axpi	339
36.1.1	How to setup date	339
36.1.2	Python proxy server	339
36.2	Raspberry PI	339
36.2.1	How to connect it to Windows with direct ethernet connection	339
36.2.2	Default IP of Raspberry pi	340
36.2.3	Headless setup	340
36.3	ALSA	340
36.3.1	USB microphone for raspberry pi	341
36.3.2	alsa-utils	341
36.3.3	Cross compiling alsa-lib	342
36.3.4	Introduction to Sound Programming with ALSA	342
36.4	nano-pi	343
36.5	star visionfive 2	344
36.6	Lichee Pi 4A	347
36.6.1	openKylin	347
37	ssh	353

37.1	Basics	353
37.1.1	Remote forward	353
37.1.2	Local forward	353
38	onnx	355
38.1	Basics	355
38.2	Python notes	355
38.2.1	TensorProto	355
38.2.2	TensorShapeProto	357
38.3	ValueInfoProto	358
38.4	onnxruntime	358
38.4.1	Install	358
38.4.2	Basics	358
39	csharp	361
39.1	Basics	361
39.1.1	Use the commandline	361
39.1.2	Visual Studio for Mac	362
39.1.3	global usings	362
39.1.4	nullable types	363
39.1.5	Dictionary	363
39.1.6	foreach	363
39.1.7	my first page	364
39.1.8	EF	364
39.2	nuget	366
39.2.1	References	366
39.3	dotnet	368
39.4	Blazor	368
39.5	Hello	369
39.5.1	unsafe	369
39.5.2	Span	371
39.5.3	File	372
39.5.4	TODO	372
40	Flask	375
40.1	Install	375
40.2	Hello	375
40.3	MySQL	376
40.3.1	Installation on macOS	376
40.4	Node	377
40.4.1	Installation on macos	377
41	ARM	379
41.1	Hello ARMv8	379
41.2	Registers	379
41.3	Intrinsics	379
42	VirtualBox	381
42.1	Add new disk	381
43	Go	383
43.1	Install	383
43.1.1	macos	383
43.2	Cross compile	383
43.2.1	Build for windows	384

43.2.2	Build for Linux	384
43.2.3	Build for macOS	384
43.2.4	Build for Raspberry Pi (arm64)	384
43.2.5	Build for Raspberry Pi (arm32)	385
43.3	Basics	385
43.3.1	Environment variables	385
43.3.2	Hello world	385
43.3.3	verbose build	385
43.3.4	exported	385
43.4	hello	386
44	Whisper	389
44.1	Basics	389
44.1.1	transforms	389
44.1.2	optimum	389
45	Windows	391
45.1	Scripts	391
45.1.1	Activate Visual studio	391
45.1.2	View DLL dependencies	391
45.1.3	Remove folder	391
45.2	Visual studio	391
45.2.1	Property files	391
46	qt	393
46.1	Install	393
46.2	Hello	393
47	webassembly	395
47.1	hello	395
47.2	emsdk	395
47.2.1	Install	395
47.2.2	hello.cc	395
47.2.3	emscripten.h	396
48	spleeter	397
48.1	Basics	397
49	django	399
49.1	Basics	399
50	React	401
50.1	Basics	401
51	tts	403
51.1	vits	403
51.2	unidecode	403
51.3	phonemizer	403
51.4	books	404
51.5	papers	404
51.6	pypinyin	404
51.7	todo	404
52	rust	407

53	ELF	409
53.1	Basics	409
53.1.1	What is this GLIBCXX error?	409
54	ROS2	411
54.1	Docker	411
55	OpenFst	413
55.1	Basics	413
55.1.1	compat.h	414

Download this website in a single [pdf file](#).

1.1 cppcon

- Opening Keynote Meeting C++ 2019 - Howard Hinnant - Design Rationale for the chrono Library
https://www.youtube.com/watch?v=adSAN282YIw&ab_channel=MeetingCpp

1.2 Meeting C++

- <https://meetingcpp.com/mcpp/slides/>

1.3 getline

Listing 1: ./code/getline/main.cc

```
1  #include <cassert>
2  #include <iostream>
3  #include <sstream>
4  #include <string>
5
6  int main() {
7      std::string s = "1,2,3";
8      std::stringstream ss(s);
9      std::getline(ss, s, ',');
10     assert(s == "1");
11
12     std::getline(ss, s, ',');
13     assert(s == "2");
14
15     std::getline(ss, s, ',');
16     assert(s == "3");
17     return 0;
18 }
```

1.4 ratio

- <https://en.cppreference.com/w/cpp/numeric/ratio/ratio>

Listing 2: ./code/ratio/main.cc

```
1  #include <assert.h>
2  #include <ratio>
3
4  int main() {
5      {
6          std::ratio<1, 3> r;
7          assert(r.num == 1);
8          assert(r.den == 3);
9      }
10
11     // some predefined constants
12     {
13         std::pico r;
14         assert(r.num == 1);
15         assert(r.den == 1e12);
16     }
17
18     {
19         std::nano r;
20         assert(r.num == 1);
21         assert(r.den == 1e9);
22     }
23
24     {
25         std::micro r;
26         assert(r.num == 1);
27         assert(r.den == 1e6);
28     }
29
30     {
31         std::milli r;
32         assert(r.num == 1);
33         assert(r.den == 1e3);
34     }
35
36     {
37         std::kilo r;
38         assert(r.num == 1e3);
39         assert(r.den == 1);
40     }
41
42     {
43         std::mega r;
44         assert(r.num == 1e6);
45         assert(r.den == 1);
46     }
47
```

(continues on next page)

(continued from previous page)

```

48 {
49     std::giga r;
50     assert(r.num == 1e9);
51     assert(r.den == 1);
52 }
53 return 0;
54 }

```

1.5 chrono

- https://www.youtube.com/watch?v=P32hvk8b13M&ab_channel=CppCon
- <https://github.com/CppCon/CppCon2016/tree/master/Tutorials/A%20chrono%20Tutorial>

Everything is inside the namespace `std::chrono`.

Differences between:

- `std::time_t`
- `std::tm`
- `std::time()`

`std::time()` to get the number of elapsed seconds since the epoch time 1970.1.1 00:00:00. The returned type is `time_t`. Note its precision is in seconds.

`std::localtime` converts `time_t` to a struct `tm` in local time.

`std::mktime()` converts a struct `tm` into `time_t` in local time.

`std::asctime()` converts a struct `tm` into a `char*` in local time.

`std::ctime()` converts a `time_t` into a `char*` in local time.

`std::strftime` is more powerful.

`std::this_thread::sleep_until` takes a timepoint as argument, while `std::this_thread::sleep_for` takes a duration as argument.

After using namespace `std::chrono_literals`; or using `std::chrono::operator""s`, we can use `auto s = 1s`.

Listing 3: `./code/chrono/main.cc`

```

1 // Requires c++14
2 // g++ -std=c++14 main.cc
3 #include <assert.h>
4 #include <chrono>
5 #include <ctime>
6 #include <iostream>
7 #include <string>
8 #include <thread>
9 #include <type_traits>
10 #include <vector>
11
12 void test_duration() {

```

(continues on next page)

(continued from previous page)

```

13 std::chrono::seconds s(1);
14 static_assert(
15     std::is_same<decltype(s),
16         std::chrono::duration<long long, std::ratio<1>>>::value,
17     "");
18 static_assert(sizeof(s) == sizeof(long long), "");
19
20 assert(s.count() == 1);
21
22 std::chrono::milliseconds ms(1);
23 static_assert(
24     std::is_same<decltype(ms), std::chrono::duration<
25         long long, std::ratio<1, 1000>>>::value,
26     "");
27
28 static_assert(sizeof(ms) == sizeof(long long), "");
29
30 assert(ms.count() == 1);
31 ms = s; // implicit cast
32 assert(ms.count() == 1000);
33
34 s = std::chrono::duration_cast<std::chrono::seconds>(ms);
35 assert(s.count() == 1);
36
37 using namespace std::chrono_literals;
38 auto s2 = 10s;
39 static_assert(std::is_same<decltype(s), decltype(s2)>::value, "");
40 assert(s2.count() == 10);
41
42 ms = s2;
43 assert(ms.count() == 1e4);
44 ms = ms + s;
45 assert(ms.count() == 11000);
46
47 // s = 3; // error
48 // std::cout << s << "\n"; // error
49
50 s = 1s;
51 ms = 1ms;
52 ms = s + ms;
53 assert(ms.count() == 1001);
54 s = std::chrono::duration_cast<std::chrono::seconds>(ms);
55 assert(s.count() == 1);
56
57 ms = 1100ms;
58 s = std::chrono::duration_cast<std::chrono::seconds>(ms); // truncated
59 assert(s.count() == 1);
60
61 std::chrono::minutes m = 1min;
62 s = m;
63 assert(s.count() == 60);
64

```

(continues on next page)

(continued from previous page)

```

65  std::chrono::hours h = 1h;
66  s = h;
67  assert(s.count() == 60 * 60);
68
69  std::chrono::seconds s3; // s3.count() is uninitialized
70  // assert(s3.count() == 0); // does not hold
71 }
72
73 void test_time_1() {
74     // the resolution of std::time() is in seconds
75     std::time_t t = std::time(nullptr);
76     t = 1669039051;
77     // 16690369051
78     // 16690369052
79     // 16690369053
80     // t is an integer, representing the elapsed seconds since 1970.1.1 00:00 UTC
81
82     // the returned value points to a static allocated memory. Not thread safe.
83     // it is overwritten in the next call
84     std::tm *tm = std::localtime(&t);
85
86     // 2022-11-21 21:57:31
87     printf("%d-%d-%d %d:%d:%d\n", tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday,
88           tm->tm_hour, tm->tm_min, tm->tm_sec);
89     // tm_year, years since 1900
90     // tm_mon: months since january (0-11)
91     std::vector<std::string> days = {"sunday", "saturday", "monday", "tuesday",
92                                     "wednesday", "thursday", "friday"};
93     // tm_wday: days since sunday (0-6)
94     printf("weekday: %s, %d\n", days[tm->tm_wday].c_str(), tm->tm_wday);
95     // weekday: saturday, 1
96
97     // tm_yday: days since january 1 (0-365)
98     printf("day of the year: %d\n", tm->tm_yday);
99     // day of the year: 324
100
101     // std::asctime() is not threadsafe
102     //
103     // Mon Nov 21 21:57:31 2022
104     printf("%s\n", std::asctime(tm));
105
106     // Mon Nov 21 13:57:31 2022
107     std::tm *gm = std::gmtime(&t); // UTC
108     printf("%s\n", std::asctime(gm));
109
110     // Mon Nov 21 21:57:31 2022
111     printf("%s\n", std::ctime(&t));
112     // std::ctime(&t) == std::asctime(std::localtime(&t))
113
114     // tm is in localtime
115     //
116     // https://en.cppreference.com/w/cpp/chrono/c/strftime

```

(continues on next page)

(continued from previous page)

```
117 {
118     char s[100];
119     std::strftime(s, sizeof(s), "%Y", tm);
120     assert(s == std::string("2022"));
121 }
122
123 {
124     char s[100];
125     std::strftime(s, sizeof(s), "%y", tm);
126     assert(s == std::string("22")); // the last 2 digits of the year
127 }
128
129 {
130     char s[100];
131     std::strftime(s, sizeof(s), "%C", tm);
132     assert(s == std::string("20")); // the first 2 digits of the year
133 }
134
135 {
136     char s[100];
137     std::strftime(s, sizeof(s), "%b", tm);
138     assert(s == std::string("Nov"));
139 }
140
141 {
142     char s[100];
143     std::strftime(s, sizeof(s), "%B", tm);
144     assert(s == std::string("November"));
145 }
146
147 {
148     char s[100];
149     std::strftime(s, sizeof(s), "%m", tm);
150     assert(s == std::string("11")); // month
151 }
152
153 {
154     char s[100];
155     std::strftime(s, sizeof(s), "%U", tm);
156     // week of the year. Sunday is the first day of the week
157     assert(s == std::string("47"));
158 }
159
160 {
161     char s[100];
162     std::strftime(s, sizeof(s), "%W", tm);
163     // week of the year. Monday is the first day of the week
164     assert(s == std::string("47"));
165 }
166
167 {
168     char s[100];
```

(continues on next page)

(continued from previous page)

```

169     std::strftime(s, sizeof(s), "%j", tm);
170     // day of the year: 1-366
171     assert(s == std::string("325"));
172 }
173
174 {
175     char s[100];
176     std::strftime(s, sizeof(s), "%d", tm);
177     // day of the month: 1-31
178     assert(s == std::string("21"));
179 }
180 }
181
182 void test_time_2() {
183     struct std::tm tm {};
184     tm.tm_year = 1970 - 1900;
185     tm.tm_mon = 0;
186     tm.tm_mday = 1;
187     tm.tm_sec = 1;
188     tm.tm_isdst = 0;
189     // Cannot use std::mktime() here since it treats tm as localtime
190     // std::time_t t = std::mktime(&tm);
191
192     std::time_t t = timegm(&tm);
193     assert(t == 1); // epoch time is 1970, january 1, 00:00:00
194 }
195
196 void test_timepoint() {
197     {
198         // conversion between timepoint and time_t
199         std::time_t t = std::time(nullptr);
200
201         std::chrono::system_clock::time_point tp =
202             std::chrono::system_clock::from_time_t(t);
203
204         auto duration = tp.time_since_epoch();
205         auto s = std::chrono::duration_cast<std::chrono::seconds>(duration);
206         assert(t == s.count());
207     }
208
209     {
210         std::chrono::system_clock::time_point now =
211             std::chrono::system_clock::now();
212         std::time_t t = std::chrono::system_clock::to_time_t(now);
213         std::time_t t2 = std::time(nullptr);
214         assert(t == t2);
215     }
216 }
217
218 void test_sleep() {
219     std::time_t t = std::time(nullptr);
220     // note: std::ctime contains "\n"

```

(continues on next page)

(continued from previous page)

```
221     std::cout << "start to sleep at " << std::ctime(&t);
222     auto now = std::chrono::steady_clock::now();
223     using std::chrono::operator""s;
224     std::this_thread::sleep_until(now + 1s);
225     t = std::time(nullptr);
226     std::cout << "finish sleeping at " << std::ctime(&t);
227 }
228
229 void test_sleep_2() {
230     auto start = std::chrono::steady_clock::now();
231     using std::chrono::operator""ms;
232     // use sleep_until with a timepoint
233     std::this_thread::sleep_until(start + 500ms);
234     auto end = std::chrono::steady_clock::now();
235     std::cout << "finish sleeping at "
236               << std::chrono::duration<double, std::milli>(end - start).count()
237               << " ms\n";
238 }
239
240 void test_sleep_3() {
241     auto start = std::chrono::steady_clock::now();
242     using std::chrono::operator""ms;
243     // use sleep_for with a duration
244     std::this_thread::sleep_for(10ms);
245     auto end = std::chrono::steady_clock::now();
246     std::cout << "finish sleeping at "
247               << std::chrono::duration<double, std::milli>(end - start).count()
248               << " ms\n";
249 }
250
251 int main() {
252     test_duration();
253     test_time_1();
254     test_time_2();
255     test_timepoint();
256     test_sleep();
257     test_sleep_2();
258     test_sleep_3();
259     return 0;
260 }
```

This page describes how this website is setup.

2.1 Setup

1. Install the dependencies in `./docs/requirements.txt`.

```
sphinx
sphinx-autodoc-typehints
sphinx_rtd_theme
sphinxcontrib-bibtex
```

2. Use `sphinx-quickstart` to generate the skeleton. When it prompts:

```
Separate source and build directories(y/n)
```

Answer yes.

3. Edit `docs/source/conf.py` and add the following lines to it:

```
import sphinx_rtd_theme
extensions = [
    'sphinx.ext.autodoc',
    'sphinx.ext.autosummary',
    'sphinx.ext.githubpages',
    'sphinx.ext.mathjax',
    'sphinx.ext.napoleon',
    'sphinx.ext.todo',
    'sphinx.ext.viewcode',
    'sphinxcontrib.bibtex',
]

html_theme = 'sphinx_rtd_theme'

master_doc = 'index'
pygments_style = 'sphinx'
html_theme_path = [sphinx_rtd_theme.get_html_theme_path()]
smartquotes = False
html_show_sourcelink = True

html_context = {
```

(continues on next page)

(continued from previous page)

```
'display_github': True,
'github_user': 'csu-fangjun',
'github_repo': 'notes',
'github_version': 'master',
'conf_py_path': '/docs/source/',
}

html_theme_options = {
    'logo_only': False,
    'display_version': True,
    'prev_next_buttons_location': 'bottom',
    'style_external_links': True,
}
latex_engine = 'xelatex'
```

4. To generate the notes in pdf format, use `make latex`, which generates lots of `tex` files in `./build/latex`. Switch to `build/latex` and run `make`. Assume that you have installed the software to compile `tex` files. It will generate `notes.pdf`.

2.2 How to include code from a file

See <https://www.sphinx-doc.org/en/master/usage/restructuredtext/directives.html#directive-literalinclude>.

1. Show line number: `:linenos:`. By default, line number counts from 0. To add an offset, e.g., 10, to the line number, use `:lineno-start: 10`. Note: It still includes all the contents of the file.
2. To emphasize a line, specified lines, or specified line ranges, use: `:emphasize-lines: 10`, `:emphasize-lines: 10,12,14`, and `:emphasize-lines: 12,15-18`. Note: `emphasize` means to change the background color.
3. Set the language, e.g., `:language: python`.
4. Set the caption, e.g., `:caption: hello world`.
5. To include a function from the python file, use `:pyobject: my_func`.
6. To include specified lines, use `:lines:1,3,5-10,15-`. Note that if using this option, line number counts from 0. Use `:lineno-start: xx` to change the offset for display.

2.3 Link

See <https://sublime-and-sphinx-guide.readthedocs.io/en/latest/references.html> and <https://www.sphinx-doc.org/en/master/usage/restructuredtext/basics.html#hyperlinks>

2.3.1 hello

Here is a link to *hello*.

```
.. _Link to hello:
```

```
hello
```

```
-----
```

```
Here is a link to :ref:`Link to hello`.
```


This page describes commonly used git commands.

3.1 Branch

After renaming `main` to `master`, do the following locally:

```
git branch -m main master
git fetch origin
git branch -u origin/master master
git remote set-head origin -a
```

3.2 Commands

3.2.1 rev-parse

It is quite common to get the root directory of the repository with the command:

```
git rev-parse --show-toplevel
```

For instance, the above command executed in this repository prints something like as follows:

```
/xxx/notes
```

The following shows its usage in a Python script:

```
#!/usr/bin/env python3

import subprocess

d = (
    subprocess.check_output(["git", "rev-parse", "--show-toplevel"])
    .decode("ascii")
    .strip() # remove the trailing \n
)
print(d) # /path/to/notes
```

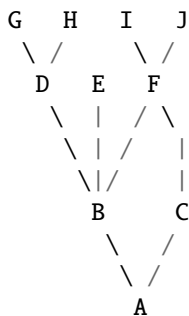
It can also be used in bash script:

```
root_dir=$(git rev-parse --show-toplevel)
echo "root_dir ${root_dir}"
```

help git-rev-parse outputs helpful information for git rev-parse. In particular, it explains the differences among HEAD~, HEAD~n, HEAD^, and HEAD^n. The following shows the help information about it:

```
<rev>^[<n>], e.g. HEAD^, v1.5.1^0
A suffix ^ to a revision parameter means the first parent of that commit object. ^
↳ <n> means the <n>th parent
(i.e. <rev>^ is equivalent to <rev>^1). As a special rule, <rev>^0 means the
↳ commit itself and is used when
<rev> is the object name of a tag object that refers to a commit object.

<rev>~[<n>], e.g. HEAD~, master~3
A suffix ~ to a revision parameter means the first parent of that commit object. A
↳ suffix ~<n> to a revision
parameter means the commit object that is the <n>th generation ancestor of the
↳ named commit object, following
only the first parents. I.e. <rev>~3 is equivalent to <rev>^^^ which is equivalent
↳ to <rev>^1^1^1. See below
for an illustration of the usage of this form.
```



```
A =      = A^0
B = A^    = A^1      = A~1
C =      = A^2
D = A^^   = A^1^1    = A~2
E = B^2   = A^^2
F = B^3   = A^^3
G = A^^^  = A^1^1^1  = A~3
H = D^2   = B^^2     = A^^^2 = A~2^2
I = F^    = B^3^     = A^^3^
J = F^2   = B^3^2    = A^^3^2
```

4.1 Installation

4.1.1 macos

Refer to <https://docs.docker.com/desktop/mac/install/>.

4.1.2 Change directory to save images

```
docker info -f '{{ .DockerRootDir }}'
mkdir -p /tmp/new-docker-root
sudo vi /etc/docker/daemon.json

cat /etc/docker/daemon.json
{
  "data-root": "/tmp/new-docker-root"
}

sudo systemctl daemon-reload
sudo systemctl restart docker
docker info -f '{{ .DockerRootDir}}'
```

4.1.3 Run an image

```
sudo docker run -it pytorch/manylinux-builder:cuda10.2 /bin/bash
```

Map a host directory to docker container:

4.1.4 copy data between container and host

```
sudo docker cp container:src_path dst_path  
sudo docker cp src_path container:dst_path
```

Use `docker ps` to view the container ID.

5.1 TikZ

5.1.1 Basics

This page describes commonly used git commands.

6.1 Decoding

```
CompactLattice compact_lat;  
decoder.GetLattice(true, &compact_lat);  
  
CompactLattice compact_best_path;  
CompactLatticeShortestPath(compact_lat, &compact_best_path);  
  
Lattice best_path;  
ConvertLattice(compact_best_path, best_path);  
  
std::vector<int32_t> tokens;  
std::vector<int32_t> words;  
LatticeWeight weight;  
GetLinearSymbolSequence(best_path, &tokens, &words, &weight);
```

- `decoder/simple-decoder.{h,cc}`

BASH

7.1 sort

Sort files in the folder `t`. The filename has the pattern `xxx.n.txt`, where `n` is some numerical value. Also, exclude `xxx.100.txt`.

```
find ./t -name "xxx*.txt" ! -name "xxx.100.txt" -print0 | sort -z -t. -k2 -n | xargs -r0  
  
# -z line delimiter is NUL, not new line  
# -t field separator  
# -k sort via a key  
# -n numeric sort
```

Convert 32-bit floating pcm to wave:

```
# sample rate: 22050  
# c: channel  
sox -t raw -r 22050 -b 32 -e floating-point -c 1 ./t.pcm ./t.wav
```

7.2 echo

Generate a binary file:

```
echo -n -e '\x30\x31\x32' > a.bin  
hexdump a.bin
```

7.3 ffmeg

```
ffprobe xxx.opus
```

7.3.1 Convert format

- opus to wav

```
ffmpeg -i input.opus output.wav  
  
ffmpeg -i input.opus -acodec pcm_s16le -ac 1 -ar 16000 output.wav
```

- Extract part of a file

```
# extract 30 seconds starting at offset 1 minute  
ffmpeg -i input.opus -ss 60 -t 30 output.wav  
# or use HH:MM:SS format  
ffmpeg -i input.opus -ss 0:01:00 -t 0:00:30 output.wav  
  
ffmpeg -ss 00:00:16 -to 00:00:30 -i ./abc.mov -c copy part.mp4
```

7.3.2 References

See <https://gist.github.com/whizkydee/804d7e290f46c73f55a84db8a8936d74>

7.4 wget

7.4.1 cookies

1. Install the chrome extension: `editthiscookie`
2. Copy the cookie of the selected website to clipboard
3. Save it to a text file, e.g., `a.txt`
4. Load the cookie with `wget --load-cookie a.txt`

7.5 Hello

See <http://mywiki.woolledge.org/BashGuide>

7.6 pkg-config

```
# Show the pre-defined search paths  
pkg-config --variable pc_path pkg-config  
  
echo $PKG_CONFIG_PATH
```

8.1 Installation

8.1.1 CUDA 10.1.243

```
./cuda_10.1.243_418.87.00_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/  
↪software/cuda-10.1.243 --no-opengl-libs --no-drm --no-man-page  
  
# Install cuDNN  
cd /ceph-data4/fangjun/software/cuda-10.1.243  
tar xvf /ceph-sh0/fangjun/cudnn/cudnn-10.1-linux-x64-v8.0.4.30.tgz --strip-components=1
```

```
#!/usr/bin/env bash  
  
export CUDA_HOME=/ceph-data4/fangjun/software/cuda-10.1.243  
export PATH=$CUDA_HOME/bin:$PATH  
export LD_LIBRARY_PATH=$CUDA_HOME/lib64:$LD_LIBRARY_PATH  
  
# See /ceph-fj/fangjun/py38/lib/python3.8/site-packages/torch/share/cmake/Caffe2/Modules_  
↪CUDA_fix/upstream/FindCUDA.cmake  
export CUDA_TOOLKIT_ROOT_DIR=$CUDA_HOME  
export CUDA_TOOLKIT_ROOT=$CUDA_HOME  
export CUDA_BIN_PATH=$CUDA_HOME  
export CUDA_PATH=$CUDA_HOME  
export CUDA_INC_PATH=$CUDA_HOME/targets/x86_64-linux
```

8.1.2 CUDA 10.2.89

```
./cuda_10.2.89_440.33.01_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/  
↪software/cuda-10.2.89 --no-opengl-libs --no-drm --no-man-page  
# Install cuDNN  
wget https://huggingface.co/csukuangfj/cudnn/resolve/main/cudnn-10.2-linux-x64-v8.0.2.39.  
↪tgz  
tar xvf cudnn-10.2-linux-x64-v8.0.2.39.tgz --strip-components=1 -C /ceph-data4/fangjun/  
↪software/cuda-10.2.89
```

```
#!/usr/bin/env bash

export CUDA_HOME=/ceph-sh1/fangjun/software/cuda-10.2.89
export PATH=$CUDA_HOME/bin:$PATH
export LD_LIBRARY_PATH=$CUDA_HOME/lib64:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$CUDA_HOME/lib:$LD_LIBRARY_PATH

# See /ceph-fj/fangjun/py38/lib/python3.8/site-packages/torch/share/cmake/Caffe2/Modules_
↳CUDA_fix/upstream/FindCUDA.cmake
export CUDA_TOOLKIT_ROOT_DIR=$CUDA_HOME
export CUDA_TOOLKIT_ROOT=$CUDA_HOME
export CUDA_BIN_PATH=$CUDA_HOME
export CUDA_PATH=$CUDA_HOME
export CUDA_INC_PATH=$CUDA_HOME/targets/x86_64-linux
export CFLAGS=-I$CUDA_HOME/targets/x86_64-linux/include
```

8.1.3 CUDA 11.0.3

```
./cuda_11.0.3_450.51.06_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/
↳software/cuda-11.0.3 --no-opengl-libs --no-drm --no-man-page

# Install cuDNN
cd /ceph-data4/fangjun/software/cuda-11.0.3
tar xvf /ceph-sh0/fangjun/cudnn/cudnn-11.0-linux-x64-v8.0.4.30.tgz --strip-components=1
```

8.1.4 CUDA 11.3.1

```
./cuda_11.3.1_465.19.01_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/
↳software/cuda-11.3.1 --no-opengl-libs --no-drm --no-man-page
cd /ceph-data4/fangjun/software/cuda-11.3.1
tar xvf /ceph-sh0/fangjun/cudnn/cudnn-11.3-linux-x64-v8.2.1.32.tgz --strip-components=1
```

8.1.5 CUDA 11.5.2

```
./cuda_11.5.2_495.29.05_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/  
↳ software/cuda-11.5.2 --no-opengl-libs --no-drm --no-man-page  
cd /ceph-data4/fangjun/software/cuda-11.5.2  
tar xvf /ceph-sh0/fangjun/cudnn/cudnn-linux-x86_64-8.3.2.44_cuda11.5-archive.tar.xz --  
↳ strip-components=1
```

8.1.6 CUDA 11.6.1

```
./cuda_11.6.1_510.47.03_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/  
↳ software/cuda-11.6.1 --no-opengl-libs --no-drm --no-man-page  
cd /ceph-data4/fangjun/software/cuda-11.6.1  
tar xvf /ceph-sh0/fangjun/cudnn/cudnn-11.3-linux-x64-v8.2.1.32.tgz --strip-components=1
```

8.1.7 CUDA 11.7.1

```
wget https://developer.download.nvidia.com/compute/cuda/11.7.1/local_installers/cuda_11.  
↳ 7.1_515.65.01_linux.run  
chmod +x cuda_11.7.1_515.65.01_linux.run  
./cuda_11.7.1_515.65.01_linux.run --silent --toolkit --installpath=/ceph-data4/fangjun/  
↳ software/cuda-11.7.1 --no-opengl-libs --no-drm --no-man-page  
cd /ceph-data4/fangjun/software/cuda-11.7.1
```


9.1 torch.load and torch.save

Listing 1: ./code/load-and-save.py

```
1  #!/usr/bin/env python3
2
3  import torch
4  import tempfile
5
6
7  def main():
8      a = torch.arange(3)
9      with tempfile.NamedTemporaryFile() as f:
10         torch.save(a, f)
11         f.seek(0)
12         b = torch.load(f)
13         assert torch.all(torch.eq(a, b)), (a, b)
14
15
16 if __name__ == "__main__":
17     main()
```

9.2 torch.gather

Listing 2: ./code/gather.py

```
1  #!/usr/bin/env python3
2
3  import torch
4
5
6  def main():
7      left_context = 0
8      N = 1
9      T = 1
10     H = 5 # time1
11     W = 2 * H - 1 + left_context # 2time1 - 1 + left_context
```

(continues on next page)

(continued from previous page)

```
12 a = torch.randn(N, T, H, W)
13 a = torch.arange(N * T * H * W).reshape(N, T, H, W).contiguous()
14
15 if True:
16     rows = torch.arange(start=H - 1, end=-1, step=-1).unsqueeze(-1)
17     cols = torch.arange(H + left_context)
18     indexes = rows + cols
19
20     indexes = torch.tile(indexes, (N * T, 1))
21 else:
22     rows = torch.arange(start=H - 1, end=-1, step=-1)
23     cols = torch.arange(H + left_context)
24     rows = torch.cat([rows] * (N * T)).unsqueeze(-1)
25     indexes = rows + cols
26
27 print(indexes.shape)
28
29 ta = a.reshape(-1, W)
30
31 b = torch.gather(ta, dim=1, index=indexes)
32 b = b.reshape(N, T, H, -1)
33
34 c = a.as_strided(
35     (N, T, H, H + left_context),
36     (T * H * W, H * W, W - 1, 1),
37     storage_offset=H - 1,
38 )
39 assert torch.equal(b, c), (b, c)
40
41
42 if __name__ == "__main__":
43     torch.manual_seed(20220727)
44     main()
```

9.3 DDP

9.3.1 Initialization

9.4 TorchScript

9.4.1 doxygen doc

See

9.4.2 Hello

See https://pytorch.org/tutorials/beginner/Intro_to_TorchScript_tutorial.html.

`torch.jit.script` as a decorator

Listing 3: `./code/1-ex.py`

```

1 @torch.jit.script
2 def adder(x: int):
3     return x + 1
4
5
6 def test_adder():
7     assert isinstance(adder, torch.jit.ScriptFunction)
8     print(adder.graph)
9     print("-" * 10)
10    print(adder.code)
11    adder.save("adder.pt")
12
13    my_adder = torch.jit.load("adder.pt")
14
15    assert isinstance(my_adder, torch.jit._script.RecursiveScriptModule)
16    assert isinstance(my_adder, torch.jit.ScriptModule)
17    assert not isinstance(my_adder, torch.jit.ScriptFunction)
18    print(my_adder(torch.tensor([3])))
19
20
21 """
22 graph(%x.1 : int):
23     %2 : int = prim::Constant[value=1]() # ./1-ex.py:8:15
24     %3 : int = aten::add(%x.1, %2) # ./1-ex.py:8:11
25     return (%3)
26
27 -----
28 def adder(x: int) -> int:
29     return torch.add(x, 1)
30
31 4
32 """

```

`torch.Value` has the following attributes (output of `dir()`):

```

['__class__', '__delattr__', '__dir__', '__doc__', '__eq__', '__format__',
 '__ge__', '__getattr__', '__gt__', '__hash__', '__init__',
 '__init_subclass__', '__le__', '__lt__', '__module__', '__ne__', '__new__',
 '__reduce__', '__reduce_ex__', '__repr__', '__setattr__', '__sizeof__',
 '__str__', '__subclasshook__', 'copyMetadata', 'debugName', 'inferTypeFrom',
 'isCompleteTensor', 'node', 'offset', 'replaceAllUsesAfterNodeWith',
 'replaceAllUsesWith', 'requiresGrad', 'requires_grad', 'setDebugName',
 'setType', 'setTypeAs', 'toIValue', 'type', 'unique', 'uses']

```

Listing 4: ./code/1-ex.py

```

1 def print_graph():
2     assert isinstance(adder.graph, torch._C.Graph)
3     assert isinstance(adder.graph, torch.Graph)
4
5     # It should have only 1 input
6     assert len(list(adder.graph.inputs())) == 1
7
8     x = next(adder.graph.inputs())
9     assert isinstance(x, torch.Value)
10    assert isinstance(x.debugName(), str)
11    assert x.debugName() == "x.1"
12    print(type(x.uses()[0]))
13    print(dir(x.uses()[0]))
14    print(x.uses()[0].user)
15    assert isinstance(x.uses()[0].user, torch.Node)
16
17    x.setDebugName("x.2")
18    assert next(adder.graph.inputs()).debugName() == "x.2"
19    assert isinstance(x.type(), torch.IntType)
20
21    print(x.node())
22    assert isinstance(x.node(), torch.Node)
23    print(dir(x.node()))
24    n = x.node()
25    assert isinstance(n.kind(), str)
26    assert n.kind() == "prim::Param", n.kind()
27    print(n.kind())
28    # a node as input and output
29    assert list(n.inputs()) == []
30
31    # n has only one output, i.e., x
32    assert len(list(n.outputs())) == 1
33    x2 = next(n.outputs()) # its type is torch.Value
34    assert x2 is x
35    assert len(list(n.blocks())) == 0

```

torch.Node has the following attributes (output from dir()):

```

['_class__', '__delattr__', '__dir__', '__doc__', '__eq__', '__format__',
'_ge__', '__getattr__', '__getitem__', '__gt__', '__hash__', '__init__',
'_init_subclass__', '__le__', '__lt__', '__module__', '__ne__', '__new__',
'_reduce__', '__reduce_ex__', '__repr__', '__setattr__', '__sizeof__',
'_str__', '__subclasshook__', 'addBlock', 'addInput', 'addOutput',
'attributeNames', 'blocks', 'c', 'c_', 'cconv', 'copyAttributes', 'copyMetadata',
'destroy', 'eraseOutput', 'f', 'f_', 'findAllNodes', 'findNode', 'fs',
'fs_', 'g', 'g_', 'getModuleHierarchy', 'gs', 'gs_', 'hasAttribute',
'hasAttributes', 'hasMultipleOutputs', 'hasUses', 'i', 'i_', 'input',
'inputs', 'inputsAt', 'inputsSize', 'insertAfter', 'insertBefore', 'is',
'isAfter', 'isBefore', 'isNondeterministic', 'is_', 'ival', 'ival_', 'kind',
'kindOf', 'matches', 'moveAfter', 'moveBefore', 'mustBeNone', 'namedInput',
'output', 'outputs', 'outputsAt', 'outputsSize', 'owningBlock', 'prev',

```

(continues on next page)

(continued from previous page)

```
'pyname', 'pyobj', 'removeAllInputs', 'removeAttribute', 'removeInput',
'replaceAllUsesWith', 'replaceInput', 'replaceInputWith', 's', 's_', 'scalar_args',
'schema', 'scopeName', 'sourceRange', 'ss', 'ss_', 't', 't_', 'ts', 'ts_',
'ty_', 'tys_', 'z', 'z_', 'zs', 'zs_']
```

torch.jit.script as a function

Listing 5: ./code/2-ex.py

```
1 def adder(x: int):
2     return x + 2
3
4
5 def test_adder():
6     adder_func = torch.jit.script(adder)
7     assert isinstance(adder_func, torch.jit.ScriptFunction)
8     print(adder_func.graph)
9     print(adder_func(3))
10
11
12 """
13 graph(%x.1 : int):
14     %2 : int = prim::Constant[value=2]() # ./2-ex.py:6:15
15     %3 : int = aten::add(%x.1, %2) # ./2-ex.py:6:11
16     return (%3)
17
18 5
19 """
```

torchscript a module

Listing 6: ./code/3-ex.py

```
1 class MyModel(torch.nn.Module):
2     def __init__(self):
3         super().__init__()
4         self.p = torch.nn.Parameter(torch.tensor([2.0]))
5
6     def forward(self, x: torch.Tensor):
7         return self.p * x
8
9
10 def test_my_model():
11     model = MyModel()
12     scripted_model = torch.jit.script(model)
13     print(scripted_model.graph)
14     print("-" * 10)
15     print(scripted_model.code)
16     print(scripted_model(torch.tensor([10])))
```

(continues on next page)

(continued from previous page)

```

17
18
19 """
20 graph(%self : __torch__.MyModel,
21       %x.1 : Tensor):
22     %p : Tensor = prim::GetAttr[name="p"](%self)
23     %4 : Tensor = aten::mul(%p, %x.1) # ./3-ex.py:12:15
24     return (%4)
25
26 -----
27 def forward(self,
28             x: Tensor) -> Tensor:
29     p = self.p
30     return torch.mul(p, x)
31 """

```

trace a module

Listing 7: ./code/trace/ex0.py

```

1  #!/usr/bin/env python3
2
3  import torch
4
5  import torch.nn as nn
6  from typing import List
7
8
9  class Foo(nn.Module):
10     def __init__(self):
11         super().__init__()
12         self.relu = nn.ReLU()
13
14     def forward(self, x):
15         return self.relu(x)
16
17
18  def test_foo():
19     f = Foo()
20     m = torch.jit.trace(f, torch.rand(2, 3))
21
22     print(m(torch.rand(2)))
23     print(m(torch.rand(2, 3, 4)))
24     # Note: The input shape is dynamic, not fixed.
25
26
27  def simple(x: List[torch.Tensor], y: torch.Tensor):
28     x = x[0].item()
29     if x > 2:
30         return y + x + 1
31     elif x < 1:

```

(continues on next page)

(continued from previous page)

```

32     return y
33 else:
34     return y + x
35
36
37 def test_simple():
38     f0 = torch.jit.trace(simple, ([torch.tensor([0])], torch.rand(2, 3)))
39     # print(dir(f0))
40     """
41     ['__call__', '__class__', '__delattr__', '__dict__', '__dir__', '__doc__',
42     '__eq__', '__format__', '__ge__', '__getattribute__', '__gt__', '__hash__',
43     '__init__', '__init_subclass__', '__le__', '__lt__', '__module__', '__ne__',
44     '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__setattr__',
45     '__sizeof__', '__str__', '__subclasshook__', '_debug_flush_compilation_cache',
46     'code', 'get_debug_state', 'graph', 'graph_for', 'inlined_graph', 'name',
47     'qualified_name', 'save', 'save_to_buffer', 'schema']
48     """
49     # print(f0.schema) # simple(Tensor[] x, Tensor y) -> (Tensor)
50     # print(f0.code)
51     """
52     def simple(x: List[Tensor],
53               y: Tensor) -> Tensor:
54         return y
55     """
56     # print(f0.graph)
57     """
58     graph(%x : Tensor[],
59           %y : Float(2, 3, strides=[3, 1], requires_grad=0, device=cpu)):
60         return (%y)
61     """
62     # print(f0.inlined_graph) # same as the above one
63     # print(f0.name) # simple
64     print(f0.qualified_name) # __torch__.simple
65
66
67 def main():
68     # test_foo()
69     test_simple()
70
71
72 if __name__ == "__main__":
73     main()

```

Export and ignore methods

1. Use `@torch.jit.export` decorator to export a method.
2. Use `torch.jit.export` function call to export a method.
3. Use `@torch.jit.ignore` decorator to ignore a method.
4. Use `torch.jit.ignore` function call to ignore a method.
5. Use `@torch.jit.unused` or `torch.jit.unused` to ignore a method.

See [Load in C++](#) to load the saved file.

Listing 8: `./code/4-ex.py`

```

1 class MyModel(torch.nn.Module):
2     def __init__(self):
3         super().__init__()
4         self.p = torch.nn.Parameter(torch.tensor([2.0]))
5
6     def foobar(self, x: torch.Tensor):
7         return x + 3
8
9     def foo(self, x: torch.Tensor):
10        return self.foobar(x)
11
12    def bar(self, x: torch.Tensor):
13        return self.p - x
14
15    @torch.jit.export
16    def baz(self, x: torch.Tensor):
17        return self.p + x + 2
18
19    def forward(self, x: torch.Tensor):
20        return self.p * x
21
22
23 def test_my_model():
24     MyModel.foo = torch.jit.export(MyModel.foo) # manually export
25
26     # Note: forward is exported by default. We ignore it here manually
27     MyModel.forward = torch.jit.ignore(MyModel.forward)
28
29     model = MyModel()
30     scripted_model = torch.jit.script(model)
31     assert hasattr(scripted_model, "foo")
32     assert hasattr(scripted_model, "baz")
33     assert hasattr(scripted_model, "foobar") # because it is called by `foo`
34     assert not hasattr(scripted_model, "bar")
35
36     scripted_model.save("foo.pt")
37
38     m = torch.jit.load("foo.pt")
39     print(m.foo(torch.tensor([1])))
40     print(m.baz(torch.tensor([1])))

```

(continues on next page)

(continued from previous page)

```

41
42
43 """
44 graph(%self : __torch__.MyModel,
45       %x.1 : Tensor):
46     %p : Tensor = prim::GetAttr[name="p"](%self)
47     %4 : Tensor = aten::mul(%p, %x.1) # ./3-ex.py:12:15
48     return (%4)
49
50 -----
51 def forward(self,
52             x: Tensor) -> Tensor:
53     p = self.p
54     return torch.mul(p, x)
55 """

```

9.4.3 Load in C++

See https://pytorch.org/tutorials/advanced/cpp_export.html.

Load the saved `foo.pt` in C++ from *Export and ignore methods*.

Listing 9: `./code/load-in-cpp/Makefile`

```

1  USE_CXX11_ABI := $(shell python3 -c 'import torch; print(int(torch.compiled_with_cxx11_
   ↳abi()))')
2  TORCH_INSTALL_DIR := $(shell python3 -c 'import os; import torch; print(os.path.
   ↳dirname(torch.__file__))')
3
4  $(info USE_CXX11_ABI $(USE_CXX11_ABI))
5  $(info TORCH_INSTALL_DIR $(TORCH_INSTALL_DIR))
6
7  CXXFLAGS := -I$(TORCH_INSTALL_DIR)/include
8  CXXFLAGS += -I$(TORCH_INSTALL_DIR)/include/torch/csrc/api/include
9  CXXFLAGS += -I$(TORCH_INSTALL_DIR)/include/TH
10 CXXFLAGS += -I$(TORCH_INSTALL_DIR)/include/THC
11 CXXFLAGS += -std=c++14
12 CXXFLAGS += -D_GLIBCXX_USE_CXX11_ABI=$(USE_CXX11_ABI)
13
14 CXXFLAGS += -Wno-unknown-pragmas # disable omp warnings
15
16 LDFLAGS := -L$(TORCH_INSTALL_DIR)/lib
17 LDFLAGS += -lc10 -ltorch -ltorch_cpu
18 # LDFLAGS += -lc10 -ltorch
19 LDFLAGS += -Wl,-rpath,$(TORCH_INSTALL_DIR)/lib
20
21 HAS_CUDA := $(shell python3 -c 'import torch; print("yes" if torch.cuda.is_available()
   ↳else "no")')
22 HAS_CUDA := yes
23 $(info has cuda $(HAS_CUDA))
24

```

(continues on next page)

(continued from previous page)

```

25 ifeq ($(HAS_CUDA),yes)
26 CUDA_HOME := $(shell which nvcc | xargs dirname | xargs dirname)
27 CXXFLAGS += -I$(CUDA_HOME)/include
28 LDFLAGS += -L$(CUDA_HOME)/lib64
29 LDFLAGS += -lcudart -lc10_cuda -ltorch_cuda
30 LDFLAGS += -Wl,-rpath,$(CUDA_HOME)/lib64
31 endif
32
33 .PHONY: clean
34
35 main: main.o
36     $(CXX) -o $@ $< $(LDFLAGS)
37
38 main.o: main.cc
39     $(CXX) $(CXXFLAGS) -c -o $@ $<
40
41 clean:
42     $(RM) main.o main

```

Note: `torch::jit::script::Module` is deprecated, use `torch::jit::Module` instead.

Listing 10: `./code/load-in-cpp/main.cc`

```

1  #include "torch/script.h"
2
3  int main() {
4      // see torch/csrc/jit/module.h
5      torch::jit::Module m = torch::jit::load("../foo.pt");
6      std::cout << "is training: " << m.is_training() << "\n";
7      m.eval();
8      std::cout << "after m.eval(): is training: " << m.is_training() << "\n";
9      torch::Tensor x = torch::tensor({1, 2, 3}, torch::kFloat);
10     torch::Tensor y = m.run_method("baz", x).toTensor();
11     std::cout << y << "\n";
12
13     return 0;
14 }

```

The output of make is:

```

USE_CXX11_ABI 0
TORCH_INSTALL_DIR /ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch
has cuda yes
g++ -I/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/include \
    -I/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/include/torch/
↪ csrc/api/include \
    -I/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/include/TH \
    -I/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/include/THC \
    -std=c++14 \
    -D_GLIBCXX_USE_CXX11_ABI=0 \

```

(continues on next page)

(continued from previous page)

```

-Wno-unknown-pragmas \
-I/ceph-sh1/fangjun/software/cuda-10.2.89/include \
-c -o main.o main.cc
g++ -o main main.o \
-L/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/lib \
-lc10 -ltorch -ltorch_cpu \
-Wl,-rpath,/ceph-fj/fangjun/software/py38/lib/python3.8/site-packages/torch/lib \
-L/ceph-sh1/fangjun/software/cuda-10.2.89/lib64 \
-lcudart -lc10_cuda -ltorch_cuda \
-Wl,-rpath,/ceph-sh1/fangjun/software/cuda-10.2.89/lib64

```

The output of ./main is:

```

is training: 1
after m.eval(): is training: 0
5
6
7
[ CPUFloatType{3} ]

```

9.4.4 ArrayRef

See [c10/Utils/ArrayRef.h](#).

Caution: `IntArrayRef` is an alias to `ArrayRef<int64_t>`.

`ArrayRef<T>` contains only two members: A const data pointer and a size. It is trivially copyable and assignable.

It has similar methods like `std::vector`. It also has two methods to get the front and back: `front()` and `back()`; both return a const reference.

Its method `vec()` converts itself to a `std::vector` by **copying** the underlying data.

Constructors

Data members

Listing 11: `./code/array_ref/main.cc` (Check size)

```

1 struct Foo {
2     const int32_t *p;
3     size_t len;
4 };
5
6 static void TestSize() {
7     // Note: The data pointer in ArrayRef is const!
8     static_assert(sizeof(torch::ArrayRef<int32_t>) == sizeof(Foo), "");
9 }

```

Default constructed

Listing 12: ./code/array_ref/main.cc (Default constructor)

```
1 static void TestDefaultConstructor() {
2     torch::ArrayRef<int32_t> a;
3     TORCH_CHECK(a.data() == nullptr);
4     TORCH_CHECK(a.size() == 0);
5     TORCH_CHECK(a.empty() == true);
6
7     TORCH_CHECK(a.begin() == nullptr);
8     TORCH_CHECK(a.end() == nullptr);
9 }
```

From a single element

Listing 13: ./code/array_ref/main.cc (From a single element)

```
1 static void TestFromSingleElement() {
2     int32_t a = 10;
3     torch::ArrayRef<int32_t> b(a);
4     TORCH_CHECK(b[0] == a);
5     TORCH_CHECK(b.data() == &a);
6     TORCH_CHECK(b.size() == 1);
7 }
```

From an initializer list

Listing 14: ./code/array_ref/main.cc (From an initializer list)

```
1 static void TestFromInitializerList() {
2     torch::ArrayRef<int32_t> a = {1, 2, 3};
3     TORCH_CHECK(a.size() == 3);
4     TORCH_CHECK(a[0] == 1);
5     TORCH_CHECK(a[1] == 2);
6     TORCH_CHECK(a[2] == 3);
7 }
```

Other types of constructors

- From two pointers: begin and end
- From a pointer and a length
- From a *std::vector*
- From a container that has `data()` and `size()` methods
- From a C array
- From a `std::array`

9.4.5 ScalarType

See `c10/core/ScalarType.h` and <https://github.com/pytorch/pytorch/blob/master/torch/csrc/api/include/torch/types.h>.

ScalarType is an enum class, i.e., `enum class ScalarType : int8_t { ... }`.

Members

It has the following members:

Listing 15: `./code/scalar-type/members.cc`

```

1  #define AT_FORALL_SCALAR_TYPES_WITH_COMPLEX_EXCEPT_COMPLEX_HALF(_) \
2      _(uint8_t, Byte) \
3      _(int8_t, Char) \
4      _(int16_t, Short) \
5      _(int, Int) \
6      _(int64_t, Long) \
7      _(at::Half, Half) \
8      _(float, Float) \
9      _(double, Double) \
10     _(c10::complex<float>, ComplexFloat) \
11     _(c10::complex<double>, ComplexDouble) \
12     _(bool, Bool) \
13     _(at::BFloat16, BFloat16)

```

Some aliases

Listing 16: `./code/scalar-type/main.cc` (alias)

```

1  static void TestAlias() {
2      static_assert(c10::ScalarType::Int == c10::kInt, "");
3      static_assert(c10::ScalarType::Byte == c10::kByte, "");
4  }

```

Listing 17: `./code/scalar-type/alias.cc`

```

1  // See torch/csrc/api/include/torch/types.h
2  using Dtype = at::ScalarType;
3
4  /// Fixed width dtypes.
5  constexpr auto kUInt8 = at::kByte;
6  constexpr auto kInt8 = at::kChar;
7  constexpr auto kInt16 = at::kShort;
8  constexpr auto kInt32 = at::kInt;
9  constexpr auto kInt64 = at::kLong;
10 constexpr auto kFloat16 = at::kHalf;
11 constexpr auto kFloat32 = at::kFloat;
12 constexpr auto kFloat64 = at::kDouble;
13
14 /// Rust-style short dtypes.
15 constexpr auto kU8 = kUInt8;

```

(continues on next page)

(continued from previous page)

```

16 constexpr auto kI8 = kInt8;
17 constexpr auto kI16 = kInt16;
18 constexpr auto kI32 = kInt32;
19 constexpr auto kI64 = kInt64;
20 constexpr auto kF16 = kFloat16;
21 constexpr auto kF32 = kFloat32;
22 constexpr auto kF64 = kFloat64;

```

ScalarType to CPP type

Listing 18: ./code/scalar-type/main.cc

```

1 static void TestScalarTypeToCppType() {
2     static_assert(
3         std::is_same<
4             int32_t, //
5             c10::impl::ScalarTypeToCppType<c10::ScalarType::Int>::value,
6         "");
7 }

```

CPP type to ScalarType

Listing 19: ./code/scalar-type/main.cc

```

1 static void TestCppTypeToScalarType() {
2     static_assert(
3         c10::CppTypeToScalarType<float>::value == c10::ScalarType::Float, "");
4 }

```

Note: It is `c10::impl::ScalarTypeToCppType`, but it is `c10::CppTypeToScalarType`.

9.4.6 TypeMeta

See

- <https://github.com/pytorch/pytorch/blob/master/c10/util/typeid.h>
- <https://github.com/pytorch/pytorch/blob/master/c10/core/ScalarTypeToTypeMeta.h>

`struct TypeMeta` contains only a single `int16_t` data member:

Listing 20: ./code/type-meta/main.cc (Check size)

```

1 static void TestSize() {
2     static_assert(sizeof(caffe2::TypeMeta) == sizeof(int16_t), "");
3 }

```

Constructors

Listing 21: ./code/type-meta/main.cc (Make)

```

1 static void TestConstructor() {
2     caffe2::TypeMeta t = caffe2::TypeMeta::Make<int32_t>();
3     TORCH_CHECK(t.Match<int32_t>());
4
5     TORCH_CHECK(t.isScalarType());
6
7     TORCH_CHECK(t.isScalarType(torch::kInt));
8     TORCH_CHECK(t.isScalarType(torch::kFloat) == false);
9
10    TORCH_CHECK(t.name() == "int");
11 }

```

Operations with ScalarType

Listing 22: ./code/type-meta/main.cc (Operations with ScalarType)

```

1 static void TestFromScalarType() {
2     caffe2::TypeMeta t = caffe2::TypeMeta::fromScalarType(torch::kDouble);
3
4     TORCH_CHECK(t.isScalarType(torch::kDouble));
5     TORCH_CHECK(t.name() == "double");
6
7     TORCH_CHECK(t.toScalarType() == torch::kDouble);
8     TORCH_CHECK(t == torch::kDouble);
9     TORCH_CHECK(t != torch::kFloat);
10    TORCH_CHECK(torch::kInt != t);

```

9.4.7 torch::Device

See

- <https://github.com/pytorch/pytorch/blob/master/c10/core/DeviceType.h>
- <https://github.com/pytorch/pytorch/blob/master/c10/core/Device.h>

DeviceType

`torch::DeviceType` is defined as enum class `Device: int8_t {...}`. The most commonly used types are `torch::DeviceType::CPU` and `torch::DeviceType::CUDA`, which are aliased to `torch::kCPU` and `torch::kCUDA`.

Listing 23: ./code/device/main.cc

```

1 void TestDeviceType() {
2     torch::DeviceType d = torch::kCPU;
3     std::ostringstream os;
4     os << d;

```

(continues on next page)

(continued from previous page)

```
5 TORCH_CHECK(os.str() == "cpu");
6
7 TORCH_CHECK(DeviceTypeName(d /*,lower_case=false*/ ) == "CPU");
8 TORCH_CHECK(DeviceTypeName(d, /*lower_case*/ true) == "cpu");
```

Device

A `torch::Device` class has two members: a `torch::DeviceType` and an `int8_t index`.

Listing 24: `./code/device/main.cc` (Constructors)

```
1 void TestDeviceConstructorCPU() {
2     torch::Device d(torch::kCPU);
3     TORCH_CHECK(d.is_cpu() == true);
4     TORCH_CHECK(d.is_cuda() == false);
5     TORCH_CHECK(d.type() == torch::kCPU);
6     TORCH_CHECK(d.has_index() == false);
7     TORCH_CHECK(d.index() == -1);
8     TORCH_CHECK(d.str() == "cpu");
9 }
10
11 void TestDeviceConstructorCUDA() {
12     torch::Device d(torch::kCUDA, 3);
13     TORCH_CHECK(d.is_cpu() == false);
14     TORCH_CHECK(d.is_cuda() == true);
15     TORCH_CHECK(d.type() == torch::kCUDA);
16     TORCH_CHECK(d.has_index() == true);
17     TORCH_CHECK(d.index() == 3);
18     TORCH_CHECK(d.str() == "cuda:3");
19
20     d.set_index(2);
21     TORCH_CHECK(d.index() == 2);
22     TORCH_CHECK(d.str() == "cuda:2");
23
24     d = torch::Device("cpu");
25     TORCH_CHECK(d.is_cpu() == true);
26
27     d = torch::Device("CPU");
28     TORCH_CHECK(d.is_cpu() == true);
29
30     d = torch::Device("cuda:1");
31     TORCH_CHECK(d.is_cuda() == true);
32     TORCH_CHECK(d.index() == 1);
33
34     d = torch::Device("CUDA:1");
35     TORCH_CHECK(d.is_cuda() == true);
36     TORCH_CHECK(d.index() == 1);
37 }
```


9.4.8 TensorOptions

See <https://github.com/pytorch/pytorch/blob/master/c10/core/TensorOptions.h>

Constructors (not recommended)

Listing 25: ./code/tensor-options/main.cc (Not recommended constructors)

```

1 void TestConstructor() {
2     // not recommended
3     torch::TensorOptions opt1(torch::kCPU);
4     torch::TensorOptions opt2(torch::Device(torch::kCPU));
5     torch::TensorOptions opt3(torch::Device({torch::kCUDA, 1}));
6     torch::TensorOptions opt4("cpu");
7     // torch::TensorOptions opt5("CPU") // error;
8     torch::TensorOptions opt6("cuda:1");
9     // torch::TensorOptions opt7("CUDA:1"); // error
10
11     // not recommended, from a scalar type (implicit)
12     torch::TensorOptions opt8(torch::kInt32);
13 }

```

Constructors (Recommended)

Listing 26: ./code/tensor-options/main.cc (Recommended constructors)

```

1 void TestConstructor2() {
2     // recommended
3     torch::TensorOptions opt1 = torch::dtype(torch::kFloat);
4     torch::TensorOptions opt2 = torch::dtype(caffe2::TypeMeta::Make<float>());
5     torch::TensorOptions opt3 = torch::device(torch::kCPU);
6     torch::TensorOptions opt4 = torch::device({torch::kCUDA, 1});
7     // Note: torch::device() returns a TensorOptions
8     // while torch::Device() is the constructor of a class
9
10    torch::TensorOptions opt5 = torch::requires_grad(true);
11    std::cout << opt5 << "\n";
12    // TensorOptions(dtype=float (default), device=cpu (default), layout=Strided
13    // (default), requires_grad=true, pinned_memory=false (default),
14    // memory_format=(nullopt))
15
16    torch::TensorOptions opt6 = torch::dtype<float>();
17    std::cout << torch::toString(opt6) << "\n";
18    // TensorOptions(dtype=float, device=cpu (default), layout=Strided (default),
19    // requires_grad=false (default), pinned_memory=false (default),
20    // memory_format=(nullopt))
21
22    std::cout << "default:" << torch::TensorOptions() << "\n";
23    // default:TensorOptions(dtype=float (default), device=cpu (default),
24    // layout=Strided (default), requires_grad=false (default),

```

(continues on next page)

(continued from previous page)

```

25 // pinned_memory=false (default), memory_format=(nullopt))
26 }

```

Methods

Listing 27: ./code/tensor-options/main.cc (Methods)

```

1 void TestMethods() {
2     torch::TensorOptions opts = torch::dtype<float>();
3     TORCH_CHECK(opts.device() == torch::Device(torch::kCPU));
4     // It has not device_type()!
5     TORCH_CHECK(opts.device() == torch::kCPU);
6     TORCH_CHECK(opts.device().type() == torch::kCPU);
7     TORCH_CHECK(opts.requires_grad() == false);
8
9     torch::TensorOptions opts2 =
10         opts.device("cuda:2").dtype(torch::kInt).requires_grad(false);
11
12     TORCH_CHECK(opts2.dtype() == caffe2::TypeMeta::Make<int32_t>());
13     TORCH_CHECK(opts2.dtype() == torch::kInt32);
14     TORCH_CHECK(opts2.requires_grad() == false);
15 }

```

9.4.9 Tensor Creation

See

TensorDataContainer

Note: data is **copied** to the returned tensor!

See

- <https://github.com/pytorch/pytorch/blob/master/torch/csrc/api/include/torch/detail/TensorDataContainer.h>
- https://github.com/pytorch/pytorch/blob/master/tools/autograd/templates/variable_factories.h
- <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/Utils.cpp>

Support the following data types:

- From a `std::vector<T>`
- From a scalar
- From an initializer list
- From an `ArrayRef<T>`.

From std::vector

Listing 28: ./code/tensor-creation/main.cc

```

1 static void FromStdVecotr() {
2     torch::Tensor t1 = torch::tensor(std::vector<int32_t>{1, 2, 3});
3     TORCH_CHECK(t1.scalar_type() == torch::kLong);
4     t1 = t1.to(torch::kInt);
5     const int32_t *p1 = t1.data_ptr<int32_t>();
6     TORCH_CHECK(p1[0] == 1);
7     TORCH_CHECK(p1[1] == 2);
8     TORCH_CHECK(p1[2] == 3);
9
10    torch::Tensor t2 = torch::tensor(std::vector<float>{1, 2, 3});
11    TORCH_CHECK(t2.scalar_type() == torch::kFloat);
12
13    torch::Tensor t3 =
14        torch::tensor(std::vector<double>{1, 2, 3}, torch::kDouble);
15    TORCH_CHECK(t3.scalar_type() == torch::kDouble);
16
17    torch::Tensor t4 =
18        torch::tensor(std::vector<double>{1, 2, 3},
19                      torch::dtype(torch::kDouble).device("cuda:0"));
20    TORCH_CHECK(t4.is_cuda());
21 }

```

From scalar

Listing 29: ./code/tensor-creation/main.cc

```

1 static void FromScalar() {
2     torch::Tensor t = torch::tensor(3);
3     TORCH_CHECK(t.item<int64_t>() == 3);
4
5     torch::Tensor t2 = torch::tensor(0.5);
6     TORCH_CHECK(t2.scalar_type() == torch::kFloat);
7 }

```

From initializer list

Listing 30: ./code/tensor-creation/main.cc

```

1 static void FromInitializerList() {
2     torch::Tensor t1 = torch::tensor({1, 2, 3});
3     torch::Tensor t2 = torch::tensor(std::vector<int32_t>{1, 2, 3});
4     TORCH_CHECK(torch::allclose(t1, t2));
5
6     torch::Tensor t3 = torch::tensor({{1, 2, 3}, {4, 5, 6}});
7     TORCH_CHECK(t3.dim() == 2);
8 }

```

(continues on next page)

(continued from previous page)

```

9  torch::Tensor t4 = torch::tensor({1, 2, 3});
10 torch::Tensor t5 = torch::tensor({4, 5, 6});
11 TORCH_CHECK(torch::allclose(t3[0], t4));
12 TORCH_CHECK(torch::allclose(t3[1], t5));
13 }

```

From ArrayRef

Listing 31: ./code/tensor-creation/main.cc

```

1  static void FromArrayRef() {
2      int32_t i[] = {1, 2, 3};
3      torch::ArrayRef<int32_t> a(i);
4      torch::Tensor t = torch::tensor(a);
5      // Data is copied to t
6
7      TORCH_CHECK(t[0].item<int64_t>(), 1);
8      TORCH_CHECK(t[1].item<int64_t>(), 2);
9      TORCH_CHECK(t[2].item<int64_t>(), 3);
10 }

```

9.4.10 Tensor

See

- <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/TensorBase.h>
- <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/templates/TensorBody.h>
- <https://github.com/pytorch/pytorch/blob/master/c10/core/TensorImpl.h>

Common methods

Listing 32: ./code/tensor/main.cc (Not recommended constructors)

```

1  static void TestCommonMethods() {
2      torch::Tensor t = torch::rand({2, 3, 4});
3
4      TORCH_CHECK(t.dim() == 3);           // 3-d tensor
5      TORCH_CHECK(t.ndimension() == t.dim()); // same
6      TORCH_CHECK(t.numel() == 2 * 3 * 4);
7      TORCH_CHECK(t.is_contiguous() == true);
8      TORCH_CHECK(t.contiguous().is_contiguous() == true);
9
10     t.fill_(10); // fill all entries to 0
11     t.zero_();   // zero out all entries
12
13     t = t.to(torch::kInt);
14     TORCH_CHECK(t.is_floating_point() == false);
15     TORCH_CHECK(t.is_signed() == true);

```

(continues on next page)

(continued from previous page)

```

16 TORCH_CHECK(t.size(0) == 2);
17 TORCH_CHECK(t.size(1) == 3);
18 TORCH_CHECK(t.size(2) == 4);
19 TORCH_CHECK(t.sizes() == torch::ArrayRef<int64_t>({2, 3, 4}));
20
21 t = t.contiguous();
22 TORCH_CHECK(t.stride(0) == 3 * 4);
23 TORCH_CHECK(t.stride(1) == 4);
24 TORCH_CHECK(t.stride(2) == 1);
25 TORCH_CHECK(t.strides() == torch::ArrayRef<int64_t>({12, 4, 1}));
26
27 TORCH_CHECK(t.defined() == true);
28 {
29     torch::Tensor a;
30     TORCH_CHECK(a.defined() == false);
31     a = t;
32     TORCH_CHECK(a.defined() == true);
33     a.reset();
34     TORCH_CHECK(a.defined() == false);
35 }
36
37 t = t.to(torch::kShort);
38 TORCH_CHECK(t.itemsize() == sizeof(int16_t));
39 TORCH_CHECK(t.nbytes() == t.numel() * t.itemsize());
40 TORCH_CHECK(t.itemsize() == t.element_size()); // same
41
42 TORCH_CHECK(t.scalar_type() == torch::kShort);
43 TORCH_CHECK(t.dtype() == caffe2::TypeMeta::Make<int16_t>());
44 TORCH_CHECK(t.dtype().toScalarType() == torch::kShort);
45
46 TORCH_CHECK(t.device() == torch::Device("cpu"));
47 TORCH_CHECK(t.device() == torch::Device(torch::kCPU));
48
49 // Note: t.device() return an instance of torch::Device
50 // t.get_device() returns the device index.
51 TORCH_CHECK(t.get_device() == t.device().index());
52
53 TORCH_CHECK(t.is_cpu() == true);
54 TORCH_CHECK(t.is_cuda() == false);
55
56 t = t.to(torch::kInt);
57 int32_t *p = t.data_ptr<int32_t>();
58 p[0] = 100;
59
60 torch::TensorAccessor<int32_t, 3> acc = t.accessor<int32_t, 3>();
61 TORCH_CHECK(acc[0][0][0] == p[0]);
62 p[12] = -2;
63 TORCH_CHECK(acc[1][0][0] == -2);
64
65 acc[1][1][2] = 3;
66 TORCH_CHECK(*(p + 12 + 4 + 2) == 3);
67

```

(continues on next page)

(continued from previous page)

```

68
69 t = t.to(torch::kFloat);
70 t.set_requires_grad(true);
71 TORCH_CHECK(t.requires_grad() == true);
72
73 t.set_requires_grad(false);
74 TORCH_CHECK(t.requires_grad() == false);
75
76 t = t.cuda();
77 TORCH_CHECK(t.device().type() == torch::kCUDA);
78 t = t.cpu();
79
80 torch::TensorOptions opts = t.options();
81 TORCH_CHECK(opts.device() == t.device());
82 }

```

slice

Listing 33: torch::slice

```

1 static void TestSlice() {
2     auto t = torch::tensor({1, 2, 3, 4, 5}, torch::kInt);
3     torch::TensorAccessor<int32_t, 1> acc = t.accessor<int32_t, 1>();
4
5     // t2 = t[1:3]
6     torch::Tensor t2 = t.slice(/*dim*/ 0, /*start*/ 1,
7                               /*end, exclusive*/ 3); // memory is shared
8     torch::TensorAccessor<int32_t, 1> acc2 = t2.accessor<int32_t, 1>();
9     TORCH_CHECK(acc2[0] == 2);
10    TORCH_CHECK(acc2[1] == 3);
11
12    acc2[0] = 10; // also changes t since the memory is shared
13    TORCH_CHECK(acc[1] == 10);
14 }

```

topk

Listing 34: torch::topk

```

1 // https://pytorch.org/docs/stable/generated/torch.topk.html
2 static void TestTopK() {
3     auto t = torch::tensor({1, 0, 3, -1}, torch::kInt).to(torch::kFloat);
4     torch::Tensor values, indexes;
5     std::tie(values, indexes) =
6         t.topk(/*k*/ 2, /*dim*/ 0, /*largest*/ true, /*sorted*/ true);
7     auto values_acc = values.accessor<float, 1>();
8     auto indexes_acc = indexes.accessor<int64_t, 1>(); // Note: it is int64_t
9
10    TORCH_CHECK(values.numel() == 2); // k in topk is 2

```

(continues on next page)

(continued from previous page)

```
11  TORCH_CHECK(values_acc[0] == 3); // the largest value is 3, at t[2]
12  TORCH_CHECK(values_acc[1] == 1); // the second largest value is 1, at t[0]
13  //
14  TORCH_CHECK(indexes_acc[0] == 2); // the largest value is t[2]
15  TORCH_CHECK(indexes_acc[1] == 0); // the second largest value is t[0]
16  }
```

floor_divide

Listing 35: torch::floor_divide

```
1  static void TestFloorDivide() {
2      auto t = torch::tensor({1, 0, 3, 5, 9}, torch::kInt);
3      auto p = torch::floor_divide(t, 2);
4      auto acc = p.accessor<int32_t, 1>();
5      TORCH_CHECK(acc[0] == 1 / 2);
6      TORCH_CHECK(acc[1] == 0 / 2);
7      TORCH_CHECK(acc[2] == 3 / 2);
8      TORCH_CHECK(acc[3] == 5 / 2);
9      TORCH_CHECK(acc[4] == 9 / 2);
10 }
```

div

Listing 36: torch::div

```

1 // https://pytorch.org/docs/stable/generated/torch.div.html
2 static void TestDiv() {
3     auto t = torch::tensor({1, 0, 3, 5, 9}, torch::kInt);
4     // the rounding mode is supported in torch >= 1.8.0
5     auto p = torch::div(t, 2, /*rounding_mode*/ "trunc");
6     auto acc = p.accessor<int32_t, 1>();
7     TORCH_CHECK(acc[0] == 1 / 2);
8     TORCH_CHECK(acc[1] == 0 / 2);
9     TORCH_CHECK(acc[2] == 3 / 2);
10    TORCH_CHECK(acc[3] == 5 / 2);
11    TORCH_CHECK(acc[4] == 9 / 2);
12 }

```

remainder

Listing 37: torch::remainder

```

1 static void TestRemainder() {
2     auto t = torch::tensor({1, 3, 8}, torch::kInt);
3     auto p = torch::remainder(t, 3);
4     auto acc = p.accessor<int32_t, 1>();
5     TORCH_CHECK(acc[0] == 1);
6     TORCH_CHECK(acc[1] == 0);
7     TORCH_CHECK(acc[2] == 2);
8 }

```

empty

Listing 38: torch::empty

```

1 static void TestEmpty() {
2     auto t = torch::empty({3}, torch::kInt);
3     TORCH_CHECK(t.scalar_type() == torch::kInt);
4     TORCH_CHECK(t.numel() == 3);
5 }

```

stack

Listing 39: torch::stack

```

1 static void TestStack() {
2     auto t = torch::empty({6, 5}, torch::kInt);
3     auto a = torch::stack({t, t}, /*dim*/ 1);
4     TORCH_CHECK(a.sizes() == torch::ArrayRef<int64_t>({6, 2, 5}));
5
6     a = torch::stack({t, t}, /*dim*/ 0);

```

(continues on next page)

(continued from previous page)

```

7  TORCH_CHECK(a.sizes() == torch::ArrayRef<int64_t>({2, 6, 5}));
8
9  a = torch::stack({t, t}, /*dim*/ 2);
10 TORCH_CHECK(a.sizes() == torch::ArrayRef<int64_t>({6, 5, 2}));
11 }

```

unbind

Listing 40: torch::unbind

```

1  static void TestUnbind() {
2      auto t = torch::empty({4, 6, 5}, torch::kInt);
3      std::vector<torch::Tensor> v = torch::unbind(t, /*dim*/ 1);
4      TORCH_CHECK(v.size() == t.size(1));
5      for (int32_t i = 0; i != v.size(); ++i) {
6          TORCH_CHECK(v[i].sizes() == torch::ArrayRef<int64_t>({4, 5}));
7      }
8  }

```

full

Listing 41: torch::full

```

1  static void TestFull() {
2      auto t = torch::full({2, 3}, 10, torch::kInt);
3      const int32_t *p = t.data_ptr<int32_t>();
4      for (int32_t i = 0; i != t.numel(); ++i) {
5          TORCH_CHECK(p[i] == 10);
6      }
7  }

```

split

Listing 42: torch::split

```

1  static void TestSplit() {
2      auto t = torch::arange(6).reshape({2, 3});
3      std::vector<torch::Tensor> s = t.split(1);
4      TORCH_CHECK(s.size() == 2);
5      TORCH_CHECK(s[0].sizes() == torch::ArrayRef<int64_t>({1, 3}));
6      TORCH_CHECK(s[1].sizes() == torch::ArrayRef<int64_t>({1, 3}));
7
8      s = t.split(1, /*dim*/ 1);
9      TORCH_CHECK(s.size() == 3);
10     TORCH_CHECK(s[0].sizes() == torch::ArrayRef<int64_t>({2, 1}));
11     TORCH_CHECK(s[1].sizes() == torch::ArrayRef<int64_t>({2, 1}));

```

(continues on next page)

(continued from previous page)

```

12  TORCH_CHECK(s[2].sizes() == torch::ArrayRef<int64_t>({2, 1}));
13  }

```

zeros

Listing 43: torch::zeros

```

1  static void TestZeros() {
2      auto t = torch::zeros({2, 3}, torch::kFloat);
3      std::cout << t << "\n";

```

cat

Listing 44: torch::cat

```

1  static void TestCat() {
2      auto t = torch::arange(24).reshape({2, 3, 4});
3      std::vector<torch::Tensor> v(5, t);
4      auto p = torch::cat(v, /*dim*/ 1);
5      TORCH_CHECK(p.sizes() == torch::ArrayRef<int64_t>({2, 3 * 5, 4}));
6  }

```

division

Listing 45: test division

```

1  static void TestDivision() {
2      auto t = torch::arange(4).to(torch::kInt);
3      auto b = t / 2;
4      TORCH_CHECK(b.scalar_type() == torch::kFloat);
5
6      const float *p = b.data_ptr<float>();
7      TORCH_CHECK(p[0] == 0 / 2.);
8      TORCH_CHECK(p[1] == 1 / 2.);
9      TORCH_CHECK(p[2] == 2 / 2.);
10     TORCH_CHECK(p[3] == 3 / 2.);
11
12     auto c = b.to(torch::kInt);
13
14     const int32_t *q = c.data_ptr<int32_t>();
15     TORCH_CHECK(q[0] == 0 / 2);
16     TORCH_CHECK(q[1] == 1 / 2);
17     TORCH_CHECK(q[2] == 2 / 2);
18     TORCH_CHECK(q[3] == 3 / 2);
19 }

```

default constructed

Listing 46: test default constructed

```
1 void TestDefaultConstructed() {  
2     torch::Tensor t;  
3     TORCH_CHECK(t.size(0) == 0);  
}
```

copy

Listing 47: test rowwise copy

```
1 void TestCopy() {  
2     auto t0 = torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});  
3     auto t = torch::empty({4, 3}, torch::kFloat);  
4  
5     t.slice(/*dim*/ 0, 0, 1) = t0.slice(0, 0, 1);  
6     t.slice(/*dim*/ 0, 1, 2) = t0.slice(0, 1, 2);  
7     t.slice(/*dim*/ 0, 2, 3) = t0.slice(0, 0, 1) + 10;  
8     t.slice(/*dim*/ 0, 3, 4) = t0.slice(0, 1, 2) + 10;  
9  
10    std::cout << t << "\n";  
11 }
```

default addmm

Listing 48: test default constructed

```

1 void TestAddmm() {
2     std::cout << "---TestAddmm---\n";
3     // 1 2 3
4     // 4 5 6
5     torch::Tensor m =
6         torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
7
8     torch::Tensor v = torch::tensor({1, 1, -1}, torch::kFloat).unsqueeze(1);
9
10    // 10 20 30
11    torch::Tensor a = torch::tensor({10, 20}, torch::kFloat).unsqueeze(1);
12    a.addmm_(m, v);
13    std::cout << a << "\n";
14    std::cout << a.squeeze(1) << "\n";
15 }

```

elementwise operation

Listing 49: test elementwise operation

```

1 void TestElementwiseOp() {
2     std::cout << "---TestElementwiseOp---\n";
3     torch::Tensor a = torch::tensor({1, 2, 3, 40}, torch::kFloat).reshape({2, 2});
4     torch::Tensor b =
5         torch::tensor({10, 20, 30, 4}, torch::kFloat).reshape({2, 2});
6     torch::Tensor c = a * b;
7     torch::Tensor d = a / b;
8     torch::Tensor e = 1.0 / a;
9     std::cout << c << "\n"; // [[10, 40], [90, 160]]
10    std::cout << d << "\n"; // [[0.1, 0.1], [0.1, 10]]
11    std::cout << e << "\n"; // [[1.0, 0.5], [0.3333, 0.0250]], float32
12 }

```

torch.roll

Listing 50: torch.roll

```

1 void TestRoll() {
2     // 1 2 3
3     // 4 5 6
4     torch::Tensor a =
5         torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
6     torch::Tensor b = a.roll(1 /*shift right 1 column*/, 1 /*dim*/);
7     // Now b is
8     // 3 2 1
9     // 6 4 5
10
11    // -----

```

(continues on next page)

(continued from previous page)

```

12 // 1 2 3 4
13 // 5 6 7 8
14 //
15 // 9 10 11 12
16 // 13 14 15 16
17 a = torch::tensor({1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16},
18                  torch::kInt)
19      .reshape({2, 2, 4});
20 b = a.roll(1 /*shift right 1 column*/, 2 /*dim*/);
21 // now b is
22 // 4 1 2 3
23 // 8 5 6 7
24 //
25 // 12 9 10 11
26 // 16 13 14 15
27 std::cout << b;
28 }

```

torch.mean

Listing 51: torch.mean

```

1 std::vector<torch::Tensor> v(5, t);
2 auto p = torch::cat(v, /*dim*/ 1);
3 TORCH_CHECK(p.sizes() == torch::ArrayRef<int64_t>({2, 3 * 5, 4}));
4 }
5
6 static void TestDivision() {
7     auto t = torch::arange(4).to(torch::kInt);
8     auto b = t / 2;
9     TORCH_CHECK(b.scalar_type() == torch::kFloat);
10
11     const float *p = b.data_ptr<float>();
12     TORCH_CHECK(p[0] == 0 / 2.);
13     TORCH_CHECK(p[1] == 1 / 2.);
14     TORCH_CHECK(p[2] == 2 / 2.);
15     TORCH_CHECK(p[3] == 3 / 2.);
16
17     auto c = b.to(torch::kInt);
18
19     const int32_t *q = c.data_ptr<int32_t>();
20     TORCH_CHECK(q[0] == 0 / 2);
21     TORCH_CHECK(q[1] == 1 / 2);
22     TORCH_CHECK(q[2] == 2 / 2);
23     TORCH_CHECK(q[3] == 3 / 2);
24 }
25
26 void TestDefaultConstructed() {
27     torch::Tensor t;
28     TORCH_CHECK(t.size(0) == 0);

```

(continues on next page)

(continued from previous page)

```

29 }
30
31 void TestCopy() {
32     auto t0 = torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
33     auto t = torch::empty({4, 3}, torch::kFloat);
34
35     t.slice(/*dim*/ 0, 0, 1) = t0.slice(0, 0, 1);
36     t.slice(/*dim*/ 0, 1, 2) = t0.slice(0, 1, 2);
37     t.slice(/*dim*/ 0, 2, 3) = t0.slice(0, 0, 1) + 10;
38     t.slice(/*dim*/ 0, 3, 4) = t0.slice(0, 1, 2) + 10;
39
40     std::cout << t << "\n";
41 }
42
43 void TestAddmm() {
44     std::cout << "----TestAddmm---\n";
45     // 1 2 3
46     // 4 5 6
47     torch::Tensor m =
48         torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
49
50     torch::Tensor v = torch::tensor({1, 1, -1}, torch::kFloat).unsqueeze(1);
51
52     // 10 20 30
53     torch::Tensor a = torch::tensor({10, 20}, torch::kFloat).unsqueeze(1);
54     a.addmm_(m, v);
55     std::cout << a << "\n";
56     std::cout << a.squeeze(1) << "\n";
57 }
58
59 void TestElementwiseOp() {
60     std::cout << "----TestElementwiseOp---\n";
61     torch::Tensor a = torch::tensor({1, 2, 3, 40}, torch::kFloat).reshape({2, 2});
62     torch::Tensor b =
63         torch::tensor({10, 20, 30, 4}, torch::kFloat).reshape({2, 2});
64     torch::Tensor c = a * b;
65     torch::Tensor d = a / b;
66     torch::Tensor e = 1.0 / a;
67     std::cout << c << "\n"; // [[10, 40], [90, 160]]
68     std::cout << d << "\n"; // [[0.1, 0.1], [0.1, 10]]
69     std::cout << e << "\n"; // [[1.0, 0.5], [0.3333, 0.0250]], float32
70 }
71
72 void TestRoll() {
73     // 1 2 3
74     // 4 5 6
75     torch::Tensor a =
76         torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
77     torch::Tensor b = a.roll(1 /*shift right 1 column*/, 1 /*dim*/);
78     // Now b is
79     // 3 2 1
80     // 6 4 5

```

(continues on next page)

(continued from previous page)

```

81
82 // -----
83 // 1 2 3 4
84 // 5 6 7 8
85 //
86 // 9 10 11 12
87 // 13 14 15 16
88 a = torch::tensor({1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16},
89                   torch::kInt)
90     .reshape({2, 2, 4});
91 b = a.roll(1 /*shift right 1 column*/, 2 /*dim*/);
92 // now b is
93 // 4 1 2 3
94 // 8 5 6 7
95 //
96 // 12 9 10 11
97 // 16 13 14 15
98 std::cout << b;
99 }
100
101 void TestMean() {
102 // 1 2 3
103 // 4 5 6
104 torch::Tensor a =
105     torch::tensor({1, 2, 3, 4, 5, 6}, torch::kFloat).reshape({2, 3});
106 torch::Tensor b = a.mean(1 /*dim*/, true /*keep_dim*/);
107 std::cout << b;
108 // Now b is:
109 // 2
110 // 5
111 //-----
112 b = a.mean(1 /*dim*/, false /*keep_dim*/);
113 std::cout << b << "\n";
114 }

```

torch.index

Listing 52: torch.mean

```

1 void TestSlice2() {
2     auto t = torch::full({2, 3}, -1);
3     std::cout << t << "\n";
4
5     // set the last column to 0
6     t.index({torch::indexing::Slice(), -1}) = 0;
7     std::cout << t << "\n";
8 }

```

9.4.11 intrusive_ptr

9.4.12 optional

9.4.13 PackedSequence

See

- <https://github.com/pytorch/pytorch/blob/master/torch/csrc/api/include/torch/nn/utils/rnn.h>
- <https://github.com/pytorch/pytorch/blob/master/torch/nn/utils/rnn.py>

pack_padded_sequence

Listing 53: ./code/packed-sequence/main.cc

```

1 static void TestPadPackedSequence() {
2     torch::Tensor t = torch::tensor({
3         {{10, 20, 30}, {0, 0, 0}, {0, 0, 0}},
4         {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}},
5         {{-1, 2, 3}, {-4, 5, 6}, {0, 0, 0}},
6     });
7     torch::Tensor lengths = torch::tensor({1, 3, 2});
8     torch::nn::utils::rnn::PackedSequence packed_seq =
9         torch::nn::utils::rnn::pack_padded_sequence(
10             t, lengths, /*batch_first*/ true, /*enforce_sorted*/ false);
11     std::cout << "data: " << packed_seq.data() << "\n";
12     std::cout << "batch_sizes: " << packed_seq.batch_sizes() << "\n";
13     std::cout << "sorted_indices: " << packed_seq.sorted_indices() << "\n";
14     std::cout << "unsorted_indices: " << packed_seq.unsorted_indices() << "\n";
15 }
16 /*
17 data:   1   2   3
18  -1   2   3
19  10  20  30
20   4   5   6
21  -4   5   6
22   7   8   9
23 [ CPULongType{6,3} ]
24 batch_sizes: 3

```

(continues on next page)

(continued from previous page)

```

25  2
26  1
27  [ CPULongType{3} ]
28  sorted_indices: 1
29  2
30  0
31  [ CPULongType{3} ]
32  unsorted_indices: 2
33  0
34  1
35  [ CPULongType{3} ]
36  */

```

The output is

Listing 54: ./code/packed-sequence/main.cc

```

1  }
2  /*
3  data:  1  2  3
4  -1  2  3
5  10 20 30
6  4  5  6
7  -4  5  6
8  7  8  9
9  [ CPULongType{6,3} ]
10 batch_sizes: 3
11  2
12  1
13 [ CPULongType{3} ]
14 sorted_indices: 1
15  2

```

9.4.14 ivalue

Listing 55: ./code/ivalue/main.cc

```

1  #include "torch/script.h"
2
3  static void TestVectorOfTensor() {
4      torch::jit::Module m("m");
5      m.define(R"(
6          def forward(self, x, y):
7              return [x, y]
8      )");
9      auto x = torch::tensor({1, 2, 3});
10     auto y = torch::tensor({4, 5, 6});
11     auto i = m.run_method("forward", x, y);
12
13     assert(i.tagKind() == "GenericList");
14

```

(continues on next page)

(continued from previous page)

```

15 torch::ArrayRef<torch::IValue> tensor_list = i.toListRef();
16 TORCH_CHECK(torch::allclose(x, tensor_list[0].toTensor()));
17 TORCH_CHECK(torch::allclose(y, tensor_list[1].toTensor()));
18
19 torch::List<torch::IValue> k = i.toList();
20
21 torch::List<torch::Tensor> o =
22     c10::impl::toTypedList<torch::Tensor>(std::move(k));
23
24 TORCH_CHECK(torch::allclose(o[0], x));
25 TORCH_CHECK(torch::allclose(o[1], y));
26
27 std::vector<torch::Tensor> p = o.vec();
28 TORCH_CHECK(torch::allclose(p[0], x));
29 TORCH_CHECK(torch::allclose(p[1], y));
30 }
31
32 static void TestVectorOfTensor2() {
33     torch::jit::Module m("m");
34     m.define(R"(
35         def forward(self, x):
36             return [[x], [x,x]]
37     )");
38     auto x = torch::tensor({1, 2, 3});
39     auto i = m.run_method("forward", x);
40     TORCH_CHECK(i.tagKind() == "GenericList");
41
42     torch::List<torch::IValue> list = i.toList();
43     torch::Tensor a = list.get(0).toListRef()[0].toTensor();
44     TORCH_CHECK(torch::allclose(a, x));
45
46     std::vector<torch::Tensor> b =
47         c10::impl::toTypedList<torch::Tensor>(list.get(1).toList()).vec();
48     TORCH_CHECK(torch::allclose(b[0], x));
49     TORCH_CHECK(torch::allclose(b[1], x));
50 }
51
52 static void TestVectorOfTensor3() {
53     torch::jit::Module m("m");
54     m.define(R"(
55         def forward(self, x: List[torch.Tensor]):
56             return x[0] + x[1]
57     )");
58
59     std::vector<torch::Tensor> v;
60     v.push_back(torch::tensor({1, 2}));
61     v.push_back(torch::tensor({3, 4}));
62     c10::List<torch::Tensor> ilist(v);
63
64     c10::impl::GenericList generic_list = c10::impl::toList(ilist);
65
66     c10::List<torch::Tensor> l2 =

```

(continues on next page)

(continued from previous page)

```

67     c10::impl::toTypedList<torch::Tensor>(generic_list);
68
69     TORCH_CHECK(torch::allclose(l2[0], v[0]));
70     TORCH_CHECK(torch::allclose(l2[1], v[1]));
71
72     auto r = m.run_method("forward", generic_list);
73     TORCH_CHECK(torch::allclose(r.toTensor(), v[0] + v[1]));
74
75     // Note: We can pass a vector directly
76     r = m.run_method("forward", v);
77     TORCH_CHECK(torch::allclose(r.toTensor(), v[0] + v[1]));
78
79     r = m.run_method("forward", ilist); // also OK
80     TORCH_CHECK(torch::allclose(r.toTensor(), v[0] + v[1]));
81 }
82
83 static void TestVectorOfTensor4() {
84     torch::jit::Module m("m");
85     m.define(R"(
86         def forward(self, x: Tuple[List[torch.Tensor]]):
87             return x[0][0] + x[0][1]
88     )");
89
90     std::vector<torch::Tensor> v;
91     v.push_back(torch::tensor({1, 2}));
92     v.push_back(torch::tensor({3, 4}));
93     auto t = torch::ivar::Tuple::create(v);
94
95     auto r = m.run_method("forward", t);
96     TORCH_CHECK(torch::allclose(r.toTensor(), v[0] + v[1]));
97 }
98
99 static void TestVectorOfTensor5() {
100     torch::jit::Module m("m");
101     m.define(R"(
102         def forward(self, x: Tuple[List[List[torch.Tensor]], List[torch.Tensor]]):
103             return x[0][0][0] + x[0][0][1] + x[1][0] + x[1][1]
104     )");
105
106     std::vector<torch::Tensor> v;
107     v.push_back(torch::tensor({1, 2}));
108     v.push_back(torch::tensor({3, 4}));
109
110     std::vector<std::vector<torch::Tensor>> vv;
111     vv.push_back(v);
112     vv.push_back(v);
113
114     auto t = torch::ivar::Tuple::create(vv, v);
115
116     auto r = m.run_method("forward", t);
117     TORCH_CHECK(torch::allclose(r.toTensor(), v[0] + v[1] + v[0] + v[1]));
118 }

```

(continues on next page)

(continued from previous page)

```

119
120 static void TestVectorOfTensor6() {
121     // List[List[Tensor]]
122     std::vector<torch::Tensor> v;
123     v.push_back(torch::tensor({1, 2}));
124     v.push_back(torch::tensor({3, 4}));
125
126     c10::List<torch::Tensor>  ilist(v);
127     torch::IValue  ivalue(ilist);
128     TORCH_CHECK(ivalue.tagKind() == "GenericList");
129
130     c10::List<c10::List<torch::Tensor>>  ilist2(ilist);
131     ilist2.push_back(ilist);
132     ilist2.push_back(ilist);
133
134     torch::IValue  ivalue2(ilist2);
135     TORCH_CHECK(ivalue2.tagKind() == "GenericList");
136
137     c10::List<torch::IValue>  a0 = ivalue2.toList();
138     c10::List<c10::List<torch::Tensor>>  a1 =
139         c10::impl::toTypedList<c10::List<torch::Tensor>>(a0);
140
141     c10::ArrayRef<torch::IValue>  a = ivalue2.toListRef();
142
143     torch::List<torch::Tensor>  b =
144         c10::impl::toTypedList<torch::Tensor>(a[0].toList());
145     for (int32_t i = 0; i != b.size(); ++i) {
146         std::cout << b[i] << "\n";
147     }
148     std::vector<std::vector<torch::Tensor>>  v2{v};
149     torch::List<torch::List<torch::Tensor>>  c;
150     for (auto k : v2) {
151         c10::List<torch::Tensor>  dd{torch::ArrayRef<torch::Tensor>(k)};
152         c.push_back(std::move(dd));
153     }
154 }
155
156 int main() {
157     TestVectorOfTensor();
158     TestVectorOfTensor2();
159     TestVectorOfTensor3();
160     TestVectorOfTensor4();
161     TestVectorOfTensor5();
162     TestVectorOfTensor6();
163     return 0;
164 }

```

9.4.15 method

See:

- <https://github.com/pytorch/pytorch/blob/master/torch/csrc/api/include/torch/imethod.h>

Listing 56: ./code/method/main.cc

```

1  #include "torch/script.h"
2
3  static void TestHello() {
4      torch::jit::Module m("m");
5      m.define(R"(
6          def forward(self, x: torch.Tensor, y: torch.Tensor):
7              return x + y
8      )");
9
10     torch::jit::Method method = m.get_method("forward");
11     TORCH_CHECK(method.name() == "forward");
12
13     const std::vector<std::string> &names = method.getArgumentNames();
14     TORCH_CHECK(names.size() == 2);
15     TORCH_CHECK(names[0] == "x");
16     TORCH_CHECK(names[1] == "y");
17
18     std::vector<torch::IValue> args;
19     auto x = torch::tensor({1, 2});
20     auto y = torch::tensor({1, 2});
21     args.emplace_back(x);
22     args.emplace_back(y);
23     auto z = method(args).toTensor();
24
25     TORCH_CHECK(torch::equal(z, x + y));
26
27     std::shared_ptr<torch::jit::Graph> g = method.graph();
28     // see node/main.cc
29 }
30
31 int main() {
32     TestHello();
33     return 0;
34 }
```

9.4.16 type

See: - https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/jit_type_base.h -

https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/jit_type.h - https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/jit_type_base.h

`torch::Type` contains a member `torch::TypeKind`. `torch::SharedType` is a subclass of `torch::Type` and `std::enabled_shared_from_this<torch::SharedType>`.

```

// https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/jit_type_base.h#L637
using TypePtr = SingletonOrSharedTypePtr<Type>;
```

Listing 57: ./code/type/main.cc

```

1  #include "torch/script.h"
2
3  static void TestTypeKind() {
4      // https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/jit_type_base.h
5      torch::TypeKind k = torch::TypeKind::AnyType;
6      TORCH_CHECK(torch::typeKindToString(k) == std::string("AnyType"));
7
8      // NamedType is not a member of TypeKind
9  }
10
11 static void TestNumberType() {
12     // torch::NumberType::get() returns a static object!
13     // so p and q are actually the same
14     torch::NumberTypePtr p = torch::NumberType::get();
15     torch::NumberTypePtr q = torch::NumberType::get();
16
17     TORCH_CHECK(p.get() == q.get());
18
19     TORCH_CHECK(p->str() == "Scalar");
20     TORCH_CHECK(p->kind() == torch::NumberType::Kind);
21     TORCH_CHECK(p->kind() == torch::TypeKind::NumberType);
22 }
23
24 static void TestIntType() {
25     torch::IntTypePtr p = torch::IntType::get();
26     TORCH_CHECK(p->str() == "int");
27     TORCH_CHECK(p->kind() == torch::TypeKind::IntType);
28     TORCH_CHECK(p->kind() == torch::IntType::Kind);
29     TORCH_CHECK(p->isSubtypeOf(torch::NumberType::get()) == true);
30 }
31
32 static void TestFloatType() {
33     torch::FloatTypePtr p = torch::FloatType::get();
34     TORCH_CHECK(p->str() == "float");
35     TORCH_CHECK(p->kind() == torch::TypeKind::FloatType);
36     TORCH_CHECK(p->kind() == torch::FloatType::Kind);
37     TORCH_CHECK(p->isSubtypeOf(torch::NumberType::get()) == true);
38     TORCH_CHECK(p->isSubtypeOf(torch::IntType::get()) == false);
39 }
40
41 static void TestBoolType() {
42     torch::BoolTypePtr p = torch::BoolType::get();
43     TORCH_CHECK(p->str() == "bool");
44     TORCH_CHECK(p->kind() == torch::TypeKind::BoolType);
45     TORCH_CHECK(p->kind() == torch::BoolType::Kind);
46     TORCH_CHECK(p->isSubtypeOf(torch::NumberType::get()) == true);
47     TORCH_CHECK(p->isSubtypeOf(torch::IntType::get()) == false);
48 }
49
50 static void TestNamedType() {

```

(continues on next page)

(continued from previous page)

```

51 // torch::Type is an abstract class!
52 //
53 // torch::NamedType is an abstract class!
54 //
55 // torch::NamedType t(torch::TypeKind::AnyType, "foo.bar"); // error
56 // TORCH_CHECK(t.name()->qualifiedName() == "foo.bar");
57 }
58
59 static void TestAnyType() {
60     torch::AnyTypePtr p = torch::AnyType::get();
61     TORCH_CHECK(p->Kind == torch::TypeKind::AnyType);
62     TORCH_CHECK(p->kind() == torch::TypeKind::AnyType);
63     TORCH_CHECK(p->str() == "Any");
64     TORCH_CHECK(p->requires_grad() == false);
65
66     TORCH_CHECK(p == torch::AnyType::get());
67
68     // available in newer versions of PyTorch
69     // TORCH_CHECK(p->equals(torch::AnyType::get()));
70
71     TORCH_CHECK(torch::toString(p) == "Any");
72 }
73
74 int main() {
75     TestTypeKind();
76     TestNumberType();
77     TestIntType();
78     TestFloatType();
79     TestNamedType();
80     TestAnyType();
81     return 0;
82 }

```

9.4.17 trace

Listing 58: ./code/trace/ex0.py

```

1  #!/usr/bin/env python3
2
3  import torch
4
5  import torch.nn as nn
6  from typing import List
7
8
9  class Foo(nn.Module):
10     def __init__(self):
11         super().__init__()
12         self.relu = nn.ReLU()
13

```

(continues on next page)

(continued from previous page)

```

14     def forward(self, x):
15         return self.relu(x)
16
17
18 def test_foo():
19     f = Foo()
20     m = torch.jit.trace(f, torch.rand(2, 3))
21
22     print(m(torch.rand(2)))
23     print(m(torch.rand(2, 3, 4)))
24     # Note: The input shape is dynamic, not fixed.
25
26
27 def simple(x: List[torch.Tensor], y: torch.Tensor):
28     x = x[0].item()
29     if x > 2:
30         return y + x + 1
31     elif x < 1:
32         return y
33     else:
34         return y + x
35
36
37 def test_simple():
38     f0 = torch.jit.trace(simple, ([torch.tensor([0])], torch.rand(2, 3)))
39     # print(dir(f0))
40     """
41     ['_call__', '__class__', '__delattr__', '__dict__', '__dir__', '__doc__',
42     '__eq__', '__format__', '__ge__', '__getattr__', '__gt__', '__hash__',
43     '__init__', '__init_subclass__', '__le__', '__lt__', '__module__', '__ne__',
44     '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__setattr__',
45     '__sizeof__', '__str__', '__subclasshook__', '_debug_flush_compilation_cache',
46     'code', 'get_debug_state', 'graph', 'graph_for', 'inlined_graph', 'name',
47     'qualified_name', 'save', 'save_to_buffer', 'schema']
48     """
49     # print(f0.schema) # simple(Tensor[] x, Tensor y) -> (Tensor)
50     # print(f0.code)
51     """
52     def simple(x: List[Tensor],
53               y: Tensor) -> Tensor:
54         return y
55     """
56     # print(f0.graph)
57     """
58     graph(%x : Tensor[],
59           %y : Float(2, 3, strides=[3, 1], requires_grad=0, device=cpu)):
60         return (%y)
61     """
62     # print(f0.inlined_graph) # same as the above one
63     # print(f0.name) # simple
64     print(f0.qualified_name) # __torch__.simple
65

```

(continues on next page)

(continued from previous page)

```

66
67 def main():
68     # test_foo()
69     test_simple()
70
71
72 if __name__ == "__main__":
73     main()

```

Listing 59: ./code/trace/ex1.py

```

1  #!/usr/bin/env python3
2
3  import torch
4
5
6  def f(a, b):
7      c = a + b
8      d = c * c
9      e = torch.tanh(d * c)
10     return d + (e + e)
11
12
13 m = torch.jit.script(f)
14 print(m.graph)
15
16 """
17 graph(%a.1 : Tensor,
18       %b.1 : Tensor):
19     %4 : int = prim::Constant[value=1]()
20     %c.1 : Tensor = aten::add(%a.1, %b.1, %4) # ./ex1.py:7:8
21     %d.1 : Tensor = aten::mul(%c.1, %c.1) # ./ex1.py:8:8
22     %11 : Tensor = aten::mul(%d.1, %c.1) # ./ex1.py:9:19
23     %e.1 : Tensor = aten::tanh(%11) # ./ex1.py:9:8
24     %17 : Tensor = aten::add(%e.1, %e.1, %4) # ./ex1.py:10:16
25     %19 : Tensor = aten::add(%d.1, %17, %4) # ./ex1.py:10:11
26     return (%19)
27 """
28
29 """
30 Note: for aten::add(a0, a1, a2), it does a0 + a2 * a1.
31 See torch/csrc/jit/codegen/fuser/codegen.cpp
32
33 """
34 assert isinstance(m.graph, torch._C.Graph)
35
36 # Every graph has inputs and outputs
37 # m.graph.inputs() returns an iterator
38 assert len(list(m.graph.inputs())) == 2, "It has two inputs: a, b, in our case"
39 it = m.graph.inputs()
40 a = next(it)
41 b = next(it)

```

(continues on next page)

(continued from previous page)

```

42
43 assert isinstance(a, torch._C.Value)
44 assert isinstance(a.node(), torch._C.Node)
45
46 # every node has inputs and outputs
47 # a.node().inputs() is an iterator
48 assert list(a.node().inputs()) == []
49 assert a.node().kind() == "prim::Param"
50 assert a.node().inputsSize() == 0
51 assert a.node().outputsSize() == 2
52 print(next(a.node().outputs()))
53
54 oit = a.node().outputs()
55 assert next(oit) == a
56 assert next(oit) == b
57
58 assert next(a.node().outputs()) == a
59
60 assert a.node().outputsAt(0) == a
61 assert a.node().outputsAt(1) == b
62 assert a.node() == b.node()
63 assert a.node().attributeNames() == [], "this node has no attributes"
64 assert a.debugName() == "a.1"
65 assert isinstance(a.type(), torch._C.TensorType)
66 assert a.type().kind() == "TensorType"
67 assert a.unique() == 0 # TODO(fangjun): what does it mean?
68 assert isinstance(a.uses(), list)
69 assert isinstance(a.uses()[0], torch._C.Use)
70 assert isinstance(a.uses()[0].user, torch._C.Node)
71
72 c_node = a.uses()[0].user
73 assert c_node.kind() == "aten::add"
74 assert c_node.attributeNames() == []
75 assert len(list(c_node.inputs())) == 3
76 c_it = c_node.inputs()
77 assert a == next(c_it)
78 assert b == next(c_it)
79 v4 = next(c_it)
80 assert v4.debugName() == "4"
81 assert c_node.hasAttributes() is False
82 assert c_node.hasMultipleOutputs() is False
83 assert c_node.hasUses() is True
84 assert (
85     c_node.schema()
86     == "aten::add.Tensor(Tensor self, Tensor other, *, Scalar alpha=1) -> (Tensor)"
87 )
88 print(c_node.schema())
89 print(type(c_node.schema()))
90 v4_node = v4.node()
91 assert v4_node.attributeNames() == ["value"]
92 assert v4_node.hasAttributes() is True
93 assert v4_node.hasAttribute("value") is True

```

(continues on next page)

(continued from previous page)

```

94 # print(v4_node.t("value"))
95 print(dir(v4_node))

```

9.4.18 Node

Listing 60: ./code/node/main.cc

```

1  #include "torch/csrc/jit/passes/quantization/helper.h" // for removeTorchMangle
2  #include "torch/script.h"
3
4  static void TestRemoveTorchMangle() {
5      std::string s = torch::jit::removeTorchMangle("a.__torch_mangle_1.foo");
6      TORCH_CHECK(s == "a.foo");
7
8      s = torch::jit::removeTorchMangle("a.__torch_mangle_123.foo");
9      TORCH_CHECK(s == "a.foo");
10 }
11
12 static void TestSimple() {
13     torch::jit::Module m("m");
14     m.define(R"(
15         def forward(self, x: torch.Tensor, y: torch.Tensor):
16             a = x + 2
17             b = y * 3
18             return a + b
19     )");
20     std::shared_ptr<torch::jit::Graph> graph = m.get_method("forward").graph();
21     std::cout << "graph string: \n" << graph->toString() << "\n";
22     // Or we can use graph->dump();
23     torch::jit::Block *block = graph->block();
24     for (auto it = block->nodes().begin(), end = block->nodes().end();
25          it != end; ++it) {
26         torch::jit::Node *n = *it;
27         torch::jit::NodeKind k = n->kind();
28         std::cout << "node kind: " << k << " " << k.toQualString() << "\n";
29     }
30     #if 0
31     graph string:
32     graph(%self : __torch__.m,
33           %x.1 : Tensor,
34           %y.1 : Tensor):
35         %5 : int = prim::Constant[value=1]()
36         %4 : int = prim::Constant[value=2]() # <string>:3:14
37         %8 : int = prim::Constant[value=3]() # <string>:4:14
38         %a.1 : Tensor = aten::add(%x.1, %4, %5) # <string>:3:10
39         %b.1 : Tensor = aten::mul(%y.1, %8) # <string>:4:10
40         %13 : Tensor = aten::add(%a.1, %b.1, %5) # <string>:5:13
41         return (%13)
42
43     node kind: 14 prim::Constant

```

(continues on next page)

(continued from previous page)

```

44 node kind: 14 prim::Constant
45 node kind: 14 prim::Constant
46 node kind: 534 aten::add
47 node kind: 241 aten::mul
48 node kind: 534 aten::add
49 #endif
50 }
51
52 static void TestFunctionCall() {
53     torch::jit::Module m("m");
54     m.define(R"(
55         def add(self, x: torch.Tensor, y: torch.Tensor):
56             '''my add doc'''
57             return x + y + 3
58
59         def forward(self, x: torch.Tensor, y: torch.Tensor):
60             c = self.add(x, y)
61             return c
62     )");
63     std::shared_ptr<torch::jit::Graph> graph = m.get_method("forward").graph();
64     std::cout << "graph string: \n" << graph->toString() << "\n";
65     torch::jit::Block *block = graph->block();
66     for (auto it = block->nodes().begin(), end = block->nodes().end();
67          it != end;) {
68         torch::jit::Node *n = *it++;
69         torch::jit::NodeKind k = n->kind();
70         std::cout << "node kind: " << k << " " << k.toQualString() << "\n";
71     }
72     #if 0
73     graph string:
74     graph(%self.1 : __torch__.m,
75          %x.1 : Tensor,
76          %y.1 : Tensor):
77         %c.1 : Tensor = prim::CallMethod[name="add"](%self.1, %x.1, %y.1) # <string>:6:10
78         return (%c.1)
79
80 node kind: 149 prim::CallMethod
81 #endif
82     for (auto it = block->nodes().begin(), end = block->nodes().end();
83          it != end;) {
84         torch::jit::Node *n = *it++;
85         torch::jit::NodeKind k = n->kind();
86         if (k == c10::prim::CallMethod) {
87             torch::ArrayRef<torch::jit::Value *> inputs = n->inputs();
88             TORCH_CHECK(inputs.size() == 3);
89
90             torch::jit::TypePtr type = inputs[0]->type();
91
92             auto class_type = type->cast<torch::jit::ClassType>();
93             TORCH_CHECK(class_type->str() == "__torch__.m");
94             if (!class_type) {
95                 std::cout << "Not a class type: " << type->str() << "\n";

```

(continues on next page)

(continued from previous page)

```

96     continue;
97 }
98 // defined by the macro "CREATE_ACCESSOR()" in ir/ir.h
99 const std::string &function_name = n->s(c10::attr::name);
100 // const std::string &function_name = n->s(torch::jit::attr::name);
101 TORCH_CHECK(function_name == "add");
102
103 TORCH_CHECK(torch::jit::attr::name == c10::attr::name);
104
105 torch::jit::Function &function = class_type->getMethod(function_name);
106 if (!function.isGraphFunction()) {
107     std::cout << function_name << " is not a graph function"
108               << "\n";
109     continue;
110 }
111 std::string class_type_str =
112     torch::jit::removeTorchMangle(class_type->str());
113 // remove __torch__, which is 10 characters long
114 std::string no_torch_class_type_str = class_type_str.substr(10);
115 }
116 }
117 }
118
119 int main() {
120     // TestRemoveTorchMangle();
121     // TestSimple();
122     TestFunctionCall();
123     return 0;
124 }

```

9.4.19 symbol

See

- <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/symbol.h>
- https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/interned_strings.h

Listing 61: ./code/symbol/main.cc

```

1  #include "torch/script.h"
2
3  static void TestHello() {
4      // defined in
5      // https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/interned_strings.h
6      c10::Symbol s = c10::namespaces::prim;
7      TORCH_CHECK(int(s) == 0);
8
9      s = c10::namespaces::aten;
10     s = c10::namespaces::cuda;
11     s = c10::namespaces::onnx;
12     s = c10::namespaces::scope;

```

(continues on next page)

```

13  s = c10::namespaces::_caffe2;
14  s = c10::namespaces::namespaces;
15
16  s = c10::prim::Assign;
17  TORCH_CHECK(int(s) == 10);
18
19  s = c10::prim::Constant;
20  s = c10::prim::device;
21  s = c10::prim::dtype;
22
23  s = c10::aten::Bool;
24  s = c10::aten::sorted;
25
26  s = c10::onnx::Add;
27
28  // c10::_keys is an enum class.
29  int32_t max_num_symbols = int32_t(c10::_keys::num_symbols);
30
31  // max num symbols: 1535. (as of 2022-08-02)
32  std::cout << "max num symbols: " << max_num_symbols << "\n";
33
34  c10::_keys k = c10::_keys::namespaces_prim;
35  k = c10::_keys::namespaces_aten;
36  k = c10::_keys::prim_Constant;
37 }
38
39 static void TestSymbol() {
40     // Symbol contains only an integer, which is from an Enum
41     c10::Symbol s = c10::Symbol::fromQualString("prim::Constant");
42     TORCH_CHECK(s == c10::prim::Constant);
43     TORCH_CHECK(s.is_prim() == true);
44     TORCH_CHECK(s.is_aten() == false);
45     TORCH_CHECK(s.ns() == c10::namespaces::prim);
46     TORCH_CHECK(s.toUnqualString() == std::string("Constant"));
47     TORCH_CHECK(s.toQualString() == std::string("prim::Constant"));
48
49     TORCH_CHECK(s == c10::Symbol::prim("Constant"));
50
51     // Register a new symbol
52     // It will first register the namespace k2 as the symbol "namespaces::k2"
53     s = c10::Symbol::fromQualString("k2::Foo");
54
55     TORCH_CHECK(s.ns() == c10::Symbol::fromQualString("namespaces::k2"));
56     TORCH_CHECK(s.toUnqualString() == std::string("Foo"));
57     TORCH_CHECK(s.toQualString() == std::string("k2::Foo"));
58 }
59
60 int main() {
61     TestHello();
62     TestSymbol();
63     return 0;
64 }

```

9.4.20 graph

Listing 62: ./code/graph/main.cc

```

1  #include "torch/script.h"
2
3  static void TestConv2d() {
4      torch::jit::Module m("m");
5      m.define(R"(
6          def __init__(self):
7              self.conv = torch.nn.Conv2d(2, 3)
8          def forward(self, x: torch.Tensor):
9              return self.conv(x)
10     )");
11     torch::jit::Method method = m.get_method("forward");
12     std::shared_ptr<torch::jit::Graph> g = method.graph();
13     torch::ArrayRef<torch::jit::Value *> inputs = g->inputs();
14     torch::ArrayRef<torch::jit::Value *> outputs = g->outputs();
15     TORCH_CHECK(inputs.size() == 1);
16     TORCH_CHECK(outputs.size() == 1);
17
18     torch::jit::Value *in = inputs[0];
19     std::cout << in->type()->str() << "\n";
20     std::cout << in->debugName() << "\n";
21 }
22
23 int main() {
24     TestConv2d();
25     return 0;
26 }

```

Listing 63: ./code/graph/inline_calls.py

```

1  #!/usr/bin/env python3
2
3  from pathlib import Path
4
5  import torch
6  import torch.nn as nn
7
8
9  class Foo(nn.Module):
10     def __init__(self):
11         super().__init__()
12         self.linear = nn.Linear(2, 2)
13         self.linear2 = nn.Linear(2, 2)
14         self.relu = nn.ReLU()
15         self.t = torch.rand(2)
16
17     def forward(self, x: torch.Tensor):
18         y = self.linear(x + self.t)
19         y = self.linear2(y)
20         y = self.linear2(y)

```

(continues on next page)

```
21         # z = self.relu(y)
22         return nn.functional.elu(y)
23         return z
24
25
26 def generate_foo_pt():
27     f = Foo()
28     x = torch.rand(1, 2)
29     m = torch.jit.trace(f, x)
30     m.save("foo.pt")
31
32
33 def test_foo_pt():
34     m = torch.jit.load("foo.pt")
35     assert isinstance(m.forward, torch._C.ScriptMethod)
36     assert isinstance(m.forward.graph, torch._C.Graph)
37     assert isinstance(m.forward.inlined_graph, torch._C.Graph)
38
39     print(m.linear.graph)
40     return
41
42     print(m.forward.graph)
43     # print(m.forward.inlined_graph)
44     g = m.forward.graph
45     nodes = g.nodes()
46
47     n = next(nodes)
48     print(dir(n))
49     assert n.kind() == "prim::GetAttr"
50     for i in n.inputs():
51         assert isinstance(i, torch._C.Value)
52         assert i.debugName() == "self.1"
53         assert isinstance(i.type(), torch._C.ClassType)
54         t = i.type()
55         assert t.str() == "__torch__.Foo"
56
57
58 def main():
59     generate_foo_pt()
60     # test_foo_pt()
61
62
63 if __name__ == "__main__":
64     main()
```


9.4.21 passes

`_jit_pass_fuse_add_relu`

See https://github.com/pytorch/pytorch/blob/master/torch/csrc/jit/passes/fuse_relu.cpp

Listing 64: `./code/passes/fuse_add_relu.py`

```

1  #!/usr/bin/env python3
2
3  import torch
4
5
6  class Foo(torch.nn.Module):
7     def forward(self, x: torch.Tensor, y: torch.Tensor):
8         a = torch.nn.functional.relu(x + y)
9         return a + 10
10
11
12 def main():
13     f = Foo()
14     m = torch.jit.trace(f, (torch.rand(3), torch.rand(3)))
15     g = m.graph
16
17     with open("fuse_add_relu-before.txt", "w") as f:
18         print(g, file=f)
19
20     torch._C._jit_pass_fuse_add_relu(g)
21
22     with open("fuse_add_relu-after.txt", "w") as f:
23         print(g, file=f)
24
25
26 if __name__ == "__main__":
27     main()

```

Listing 65: `./code/passes/fuse_add_relu-before.txt`

```

1  graph(%self : __torch__.Foo,
2     %x : Float(3, strides=[1], requires_grad=0, device=cpu),
3     %y : Float(3, strides=[1], requires_grad=0, device=cpu)):
4     %5 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:8:0
5     %input : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%x, %y, %5) # .
6     ↪ ./fuse_add_relu.py:8:0
7     %a : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::relu(%input) # /Users/
8     ↪ fangjun/py38/lib/python3.8/site-packages/torch/nn/functional.py:1457:0
9     %8 : Long(requires_grad=0, device=cpu) = prim::Constant[value={10}]() # ./fuse_add_
10    ↪ relu.py:9:0
11    %9 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:9:0
12    %10 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%a, %8, %9) # ./
13    ↪ fuse_add_relu.py:9:0
14    return (%10)

```

Listing 66: ./code/passes/fuse_add_relu-after.txt

```

1 graph(%self : __torch__.Foo,
2     %x : Float(3, strides=[1], requires_grad=0, device=cpu),
3     %y : Float(3, strides=[1], requires_grad=0, device=cpu)):
4     %5 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:8:0
5     %11 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::_add_relu(%x, %y, %5)
6     %8 : Long(requires_grad=0, device=cpu) = prim::Constant[value={10}]() # ./fuse_add_
    ↪ relu.py:9:0
7     %9 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:9:0
8     %10 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%11, %8, %9) # ./
    ↪ fuse_add_relu.py:9:0
9     return (%10)
10

```

in_place relu_

Listing 67: ./code/passes/fuse_add_relu_.py

```

1 #!/usr/bin/env python3
2
3 import torch
4
5
6 class Foo(torch.nn.Module):
7     def forward(self, x: torch.Tensor, y: torch.Tensor):
8         a = torch.nn.functional.relu(x + y, inplace=True)
9         return a + 10
10
11
12 def main():
13     f = Foo()
14     m = torch.jit.trace(f, (torch.rand(3), torch.rand(3)))
15     g = m.graph
16
17     with open("fuse_add_relu-before.txt", "w") as f:
18         print(g, file=f)
19
20     torch._C._jit_pass_fuse_add_relu(g)
21
22     with open("fuse_add_relu-after.txt", "w") as f:
23         print(g, file=f)
24
25
26 if __name__ == "__main__":
27     main()

```

Listing 68: ./code/passes/fuse_add_relu_-before.txt

```

1 graph(%self : __torch__.Foo,
2     %x : Float(3, strides=[1], requires_grad=0, device=cpu),

```

(continues on next page)

(continued from previous page)

```

3      %y : Float(3, strides=[1], requires_grad=0, device=cpu)):
4      %5 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:8:0
5      %input : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%x, %y, %5) # .
    ↪ ./fuse_add_relu.py:8:0
6      %a : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::relu(%input) # /Users/
    ↪ fangjun/py38/lib/python3.8/site-packages/torch/nn/functional.py:1455:0
7      %8 : Long(requires_grad=0, device=cpu) = prim::Constant[value={10}]() # ./fuse_add_
    ↪ relu.py:9:0
8      %9 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:9:0
9      %10 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%a, %8, %9) # ./
    ↪ fuse_add_relu.py:9:0
10     return (%10)
11

```

Listing 69: ./code/passes/fuse_add_relu-after.txt

```

1 graph(%self : __torch__.Foo,
2      %x : Float(3, strides=[1], requires_grad=0, device=cpu),
3      %y : Float(3, strides=[1], requires_grad=0, device=cpu)):
4      %5 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:8:0
5      %11 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::_add_relu(%x, %y, %5)
6      %8 : Long(requires_grad=0, device=cpu) = prim::Constant[value={10}]() # ./fuse_add_
    ↪ relu.py:9:0
7      %9 : int = prim::Constant[value=1]() # ./fuse_add_relu.py:9:0
8      %10 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%11, %8, %9) # ./
    ↪ fuse_add_relu.py:9:0
9      return (%10)
10

```

_jit_pass_fuse_linear

See https://github.com/pytorch/pytorch/blob/master/torch/csrc/jit/passes/fuse_linear.cpp

Listing 70: ./code/passes/fuse_linear.py

```

1  #!/usr/bin/env python3
2
3  import torch
4
5
6  class Foo(torch.nn.Module):
7      def forward(self, x: torch.Tensor, w: torch.Tensor, b: torch.Tensor):
8          return torch.matmul(x, w.t()) + b
9
10
11 def main():
12     f = Foo()
13     m = torch.jit.trace(f, (torch.rand(3), torch.rand(3, 3), torch.rand(3)))
14     g = m.graph
15
16     with open("fuse_linear-before.txt", "w") as f:

```

(continues on next page)

(continued from previous page)

```

17     print(g, file=f)
18
19     torch._C._jit_pass_fuse_linear(g)
20
21     with open("fuse_linear-after.txt", "w") as f:
22         print(g, file=f)
23
24
25 if __name__ == "__main__":
26     main()

```

Listing 71: ./code/passes/fuse_linear-before.txt

```

1 graph(%self : __torch__.Foo,
2       %x : Float(3, strides=[1], requires_grad=0, device=cpu),
3       %w : Float(3, 3, strides=[3, 1], requires_grad=0, device=cpu),
4       %b : Float(3, strides=[1], requires_grad=0, device=cpu)):
5   %6 : Float(3, 3, strides=[1, 3], requires_grad=0, device=cpu) = aten::t(%w) # ./fuse_
↪ linear.py:8:0
6   %7 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::matmul(%x, %6) # ./
↪ fuse_linear.py:8:0
7   %8 : int = prim::Constant[value=1]() # ./fuse_linear.py:8:0
8   %9 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%7, %b, %8) # ./
↪ fuse_linear.py:8:0
9   return (%9)
10

```

Listing 72: ./code/passes/fuse_linear-after.txt

```

1 graph(%self : __torch__.Foo,
2       %x : Float(3, strides=[1], requires_grad=0, device=cpu),
3       %w : Float(3, 3, strides=[3, 1], requires_grad=0, device=cpu),
4       %b : Float(3, strides=[1], requires_grad=0, device=cpu)):
5   %11 : Tensor? = prim::Constant()
6   %13 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::linear(%x, %w, %11) #
↪ ./fuse_linear.py:8:0
7   %8 : int = prim::Constant[value=1]() # ./fuse_linear.py:8:0
8   %9 : Float(3, strides=[1], requires_grad=0, device=cpu) = aten::add(%13, %b, %8) # ./
↪ fuse_linear.py:8:0
9   return (%9)
10

```

9.4.22 ops

aten::add

```
// add_res = a + b * alpha
%add_res = aten::add(%a, %b, %alpha)

// add_res = a + b * alpha
%add_res = aten::add(%a, %b, %alpha, %out)
```

aten::add_

```
// add_res = a * b * alpha
%add_res = aten::add_(%a, %b, %alpha)
```

aten::_add_relu

```
// res = relu(a + b * alpha)
%res = aten::_add_relu(%a, %b, %alpha)

// res = relu(a + b * alpha)
%res = aten::_add_relu(%a, %b, %alpha, %out)
```

aten::add_relu_

```
// res = relu(a + b * alpha)
%res = aten::add_relu_(%a, %b, %alpha)
```

aten::_add_relu_

```
// res = relu(a + b * alpha)
%res = aten::_add_relu_(%a, %b, %alpha)
```

aten::relu

```
%res = aten::relu_(%add_res)
```

9.5 Logical operations

Listing 73: ./code/logical-op.py

```
1  #!/usr/bin/env python3
2
3  import torch
4
5  a = torch.tensor([float("inf")])
6  b = torch.tensor([float("nan")])
7  assert torch.isinf(a).item() is True
8  assert torch.isnan(a).item() is False
9
10 assert torch.isinf(b).item() is False
11 assert torch.isnan(b).item() is True
12
13 assert torch.logical_or(torch.isinf(a), torch.isnan(b)).item() is True
14
15 assert a.isinf().item() is True
16 assert a.isnan().item() is False
17
18 assert b.isinf().item() is False
19 assert b.isnan().item() is True
```

9.6 Note

To clip gradient, use:

```
tot_norm = torch.nn.utils.clip_grad_norm_(model.parameters(), max_norm=5, norm_type=2.0)
if torch.logical_or(tot_norm.isnan(), tot_norm.isinf()):
    # skip this update
    continue
else:
    optimizer.step()
```

9.7 Quantization

9.7.1 Internals

<https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/core/QuantizerBase.h> defines the base class Quantizer.

<https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/quantized/Quantizer.h> defines the subclasses of Quantizer, such as

- PerTensorAffineQuantizer - qscheme is kPerTensorAffine.

QScheme

See <https://github.com/pytorch/pytorch/blob/master/c10/core/QScheme.h>

Listing 74: ./code/qscheme/main.cc

```

1  #include "torch/script.h"
2
3  static void TestQScheme() {
4      TORCH_CHECK(torch::toString(torch::kPerTensorAffine) == "per_tensor_affine");
5
6      TORCH_CHECK(torch::toString(torch::kPerChannelAffine) ==
7                  "per_channel_affine");
8
9      TORCH_CHECK(torch::toString(torch::kPerTensorSymmetric) ==
10                 "per_tensor_symmetric");
11
12     TORCH_CHECK(torch::toString(torch::kPerChannelSymmetric) ==
13                 "per_channel_symmetric");
14
15     TORCH_CHECK(torch::toString(torch::kPerChannelAffineFloatQParams) ==
16                 "per_channel_affine_float_qparams");
17 }
18
19 int main() {
20     TestQScheme();
21     return 0;
22 }
```

PerTensorAffineQuantizer

It has 4 important methods:

- `QScheme qscheme() const`, always returns `kPerTensorAffine`.
- `double scale() const`
- `int64_t zero_point() const`
- `ScalarType scalar_type() const`

It uses `quantize_tensor_per_tensor_affine_cpu` when `FBGEMM` is available.

Otherwise, it uses <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/native/quantized/cpu/kernels/QuantizedOpKernels.cpp>

- For arm, it uses `quantize_tensor_arm`. It is a template with many specializations.
- For x86, it uses `quantize_val`
 - If `FBGEMM` is available, it uses `quantize_val`
 - Otherwise, it uses <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/native/quantized/AffineQuantizerBase.cpp>

```

template <typename T>
T quantize_val(double scale, int64_t zero_point, float value) {
    // std::nearbyint results in nearest integer value according to the current
    // rounding mode and the default rounding mode is rounds to even in half-way
```

(continues on next page)

(continued from previous page)

```

// cases in most popular processor architectures like x86 and ARM. This is
// typically faster than an alternatives like std::round that rounds half-way
// cases away from zero, and can be consistent with SIMD implementations for
// example in x86 using _mm512_cvtps_epi32 or mm512_round_ps with
// _MM_FROUND_CUR_DIRECTION option that also follow the current rounding mode.
int64_t qvalue;
constexpr int64_t qmin = std::numeric_limits<typename T::underlying>::min();
constexpr int64_t qmax = std::numeric_limits<typename T::underlying>::max();
float inv_scale = 1.0f / static_cast<float>(scale);
qvalue = static_cast<int64_t>(zero_point + Round(value * inv_scale));
qvalue = std::max<int64_t>(qvalue, qmin);
qvalue = std::min<int64_t>(qvalue, qmax);
return static_cast<T>(qvalue);
}

```

dequantize_val is defined as:

```

template <typename T>
TORCH_API float dequantize_val(double scale, int64_t zero_point, T value) {
    return static_cast<float>(scale) * (value.val_ - static_cast<int32_t>(zero_point));
}

```

9.7.2 torch.quantize_per_tensor

See https://pytorch.org/docs/stable/generated/torch.quantize_per_tensor.html

```

def test_quantize_per_tensor():
    scale = 0.1
    zero_point = 1
    a = torch.tensor([10.0, 2.0], dtype=torch.float32)
    q = torch.quantize_per_tensor(
        input=a,
        scale=scale,
        zero_point=zero_point,
        dtype=torch.qint8,
    )
    assert isinstance(q, torch.Tensor)
    assert q.is_quantized is True

    assert q.q_scale() == scale
    assert q.q_zero_point() == zero_point
    assert str(q.qscheme()) == "torch.per_tensor_affine"
    assert q.dtype == torch.qint8

    c = q.int_repr()
    assert c[0] == a[0] / scale + zero_point
    assert c[1] == a[1] / scale + zero_point
    assert torch.all(torch.eq(c, torch.tensor([101, 21], dtype=torch.int8)))
    d = q.dequantize()
    assert d.dtype == torch.float32
    assert torch.all(torch.eq(d, a))

```

(continues on next page)

(continued from previous page)

```

f = torch.dequantize(q)
assert torch.all(torch.eq(f, a))
# print(q)
"""
tensor([10.,  2.], size=(2,), dtype=torch.qint8,
        quantization_scheme=torch.per_tensor_affine, scale=0.1, zero_point=1)
"""
assert q[0].item() == 10 # q[0].item() will dequantize() to a float
assert q[1].item() == 2
print(type(q[0].item()))
q[0] = 2.5 # Note: it will quantize 2.5 and store it in q
print(q.int_repr())
"""
tensor([26, 21], dtype=torch.int8)
"""

```

Compress ration

```

def test_size():
    r = torch.rand(100, 100, dtype=torch.float32)
    q = torch.quantize_per_tensor(r, scale=0.1, zero_point=0, dtype=torch.qint8)
    torch.save(r, "float32.pt")
    torch.save(q, "int8.pt")
    float_size = os.path.getsize("float32.pt")
    int8_size = os.path.getsize("int8.pt")
    print("float_size:", float_size)
    print("int8_size:", int8_size)
    print(f"ratio: {float_size}/{int8_size}: {float_size/int8_size:.3f}")
    os.remove("float32.pt")
    os.remove("int8.pt")
    """
    float_size: 40747
    int8_size: 10795
    ratio: 40747/10795: 3.775
    """

```

9.7.3 quantize_per_tensor_dynamic

Listing 75: ./code/quantize_per_tensor_dynamic/main.cc

```

1 // #include "ATen/native/quantized/cpu/QuantUtils.h" // for the latest pytorch
2
3 #include "ATen/native/quantized/cpu/quant_utils.h" // for torch 1.10
4 #include "ATen/ops/quantize_per_tensor_dynamic.h" // needs torch>=1.11
5 #include "torch/script.h"
6
7 // See
8 // https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/native/quantized/cpu/

```

(continues on next page)

(continued from previous page)

```

↪QuantUtils.h#L59
9  static void TestChooseQuantizationParams() {
10     quant_utils::TensorQuantizationParams p;
11     p = quant_utils::ChooseQuantizationParams(-1 /*min*/, 2 /*max*/,
12                                              -128 /*qmin*/, 127 /*qmax*/);
13     std::cout << "zero_point: " << p.zero_point << "\n";
14     std::cout << "scale: " << p.scale << "\n";
15     /**
16      * scale = (max - min) / (qmax - qmin) = 3 / 255 = 0.0117647
17      * zero_point_min = qmin - min/scale = -128 - (-1)/scale = -43
18      * zero_point_max = qmax - max/scale = 127 - 2/scale = -43
19      *
20      * min_error = abs(qmin) - abs(min/scale) = 128 - 1/scale = 43
21      * max_error = abs(qmax) - abs(max/scale) = 127 - 2/scale = -43
22      *
23      * zero_point = (min_error < max_error) ? zero_point_min : zero_point_max
24      */
25 }
26
27 // See
28 // https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/native/quantized/QTensor.
↪cpp#L14
29 // and
30 static void TestQuantizePerTensorDynamic() {
31     torch::Tensor r = torch::tensor({-1, 0, 2}, torch::kFloat32);
32     torch::Tensor q = torch::quantize_per_tensor_dynamic(r, torch::kInt8, false);
33     std::cout << "q: " << q << "\n";
34     #if 0
35     q: -1
36     0
37     2
38     [ QuantizedCPUQInt8Type{3}, qscheme: per_tensor_affine, scale: 0.0117647, zero_point: -
↪43 ]
39     #endif
40     std::cout << "q.int_repr(): " << q.int_repr() << "\n";
41     #if 0
42     q.int_repr(): -128
43     -43
44     127
45     [ CPUCharType{3} ]
46     #endif
47 }
48
49 int main() {
50     TestChooseQuantizationParams();
51     TestQuantizePerTensorDynamic();
52     return 0;
53 }

```

9.7.4 torch.quantize_per_channel

See https://pytorch.org/docs/stable/generated/torch.quantize_per_channel.html#torch.quantize_per_channel

```
def test_quantize_per_channel_2d():
    # (N, C)
    a = torch.tensor(
        [
            [1, 2, 3],
            [4, 5, 6],
        ],
        dtype=torch.float32,
    )
    assert a.shape == (2, 3)
    scales = torch.tensor([0.125, 0.25, 0.5])

    # It will be converted to torch.int64 internally
    zero_points = torch.tensor([10, 20, 30], dtype=torch.int32)
    q = torch.quantize_per_channel(
        input=a,
        scales=scales,
        zero_points=zero_points,
        axis=1,
        dtype=torch.qint8,
    )
    assert q.dtype == torch.qint8

    assert q.q_per_channel_scales().dtype == torch.float64
    assert torch.all(torch.eq(q.q_per_channel_scales(), scales))

    assert q.q_per_channel_zero_points().dtype == torch.int64
    assert torch.all(torch.eq(q.q_per_channel_zero_points(), zero_points))

    assert str(q.qscheme()) == "torch.per_channel_affine"

    assert q.q_per_channel_axis() == 1

    i = q.int_repr()
    expected_i = torch.tensor([[18, 28, 36], [42, 40, 42]], dtype=torch.int8)
    assert i.dtype == torch.int8
    assert torch.all(torch.eq(i, expected_i))

    assert i[0][0].item() == a[0][0].item() / scales[0] + zero_points[0]
    assert i[0][1].item() == a[0][1].item() / scales[1] + zero_points[1]
    assert i[0][2].item() == a[0][2].item() / scales[2] + zero_points[2]

    assert i[1][0].item() == a[1][0].item() / scales[0] + zero_points[0]
    assert i[1][1].item() == a[1][1].item() / scales[1] + zero_points[1]
    assert i[1][2].item() == a[1][2].item() / scales[2] + zero_points[2]

    d = q.dequantize()
    assert torch.all(torch.eq(d, a))
```

(continues on next page)

(continued from previous page)

```

f = torch.dequantize(q)
assert torch.all(torch.eq(f, a))

# print(q)
"""
tensor([[1., 2., 3.],
        [4., 5., 6.]], size=(2, 3), dtype=torch.qint8,
        quantization_scheme=torch.per_channel_affine,
        scale=tensor([0.1250, 0.2500, 0.5000], dtype=torch.float64),
        zero_point=tensor([10, 20, 30]), axis=1)
"""

```

9.7.5 Observer

Listing 76: ./code/observer/ex0.py

```

1  #!/usr/bin/env python3
2
3  import torch
4  from torch.ao.quantization.observer import _with_args, MinMaxObserver
5
6
7  class Foo:
8      def __init__(self, a=1, b=2):
9          self.a = a
10         self.b = b
11
12
13  def test_with_args():
14      Foo.with_args = classmethod(_with_args)
15      foo_builder = Foo.with_args(a=3).with_args(b=4).with_args(a=10)
16      f = foo_builder()
17      assert f.a == 10  # the last a=10 replaces the first a=3
18      assert f.b == 4
19
20      f2 = foo_builder()
21      assert id(f) != id(f2)
22
23
24  def test_min_max_observer():
25      ob = MinMaxObserver(dtype=torch.qint8)
26      print(ob)  # MinMaxObserver(min_val=-inf, max_val=-inf)
27
28      ob(torch.tensor([1, 2, 3]))
29      print(ob)  # MinMaxObserver(min_val=1.0, max_val=3.0)
30
31      ob(torch.tensor([-1, 30]))
32      print(ob)  # MinMaxObserver(min_val=-1.0, max_val=30.0)
33      scale, zero_point = ob.calculate_qparams()
34      print("scale", scale)  # scale tensor([0.1216])

```

(continues on next page)

(continued from previous page)

```

35     print("zero_point", zero_point) # zero_point tensor([-120], dtype=torch.int32)
36
37
38 def main():
39     test_with_args()
40     test_min_max_observer()
41
42
43 if __name__ == "__main__":
44     main()

```

9.7.6 Hello

Listing 77: ./code/ex1.py

```

1  #!/usr/bin/env python3
2
3  import torch
4  import torch.nn as nn
5
6
7  class Model(torch.nn.Module):
8      def __init__(self):
9          super().__init__()
10         self.fc = nn.Linear(1, 1)
11
12     def forward(self, x):
13         x = self.fc(x)
14         return x
15
16
17 def main():
18     m = Model()
19     model_int8 = torch.quantization.quantize_dynamic(
20         model=m,
21         qconfig_spec={torch.nn.Linear},
22         dtype=torch.qint8,
23     )
24     print(model_int8)
25     print(model_int8.fc)
26     assert model_int8.fc.weight().is_quantized
27     assert model_int8.fc.weight().dtype == torch.qint8
28
29     assert model_int8.fc.bias().is_quantized is False
30     assert model_int8.fc.bias().dtype == torch.float32
31     assert isinstance(model_int8.fc, torch.nn.quantized.dynamic.Linear)
32     print(type(model_int8.fc))
33
34     x = torch.tensor([[1.0]], dtype=torch.float32)
35     y = m(x)

```

(continues on next page)

(continued from previous page)

```
36 print(x, y) # tensor([[1.]]) tensor([[ -1.2900]], grad_fn=<AddmmBackward0>)
37
38 qy = model_int8(x)
39 print(qy) # tensor([[ -1.2931]])
40
41
42 if __name__ == "__main__":
43     torch.manual_seed(20220723)
44     main()
```

9.7.7 References

The main implementation is in

<https://github.com/pytorch/pytorch/tree/master/aten/src/ATen/native/quantized>

- Introducing Quantized Tensor
<https://github.com/pytorch/pytorch/wiki/Introducing-Quantized-Tensor>
- Model Quantization for PyTorch (Proposal) #18318
<https://github.com/pytorch/pytorch/issues/18318>
- torch_quantization_design_proposal
https://github.com/pytorch/pytorch/wiki/torch_quantization_design_proposal

Links

- https://github.com/pytorch/pytorch/blob/master/test/quantization/core/test_quantized_tensor.py
- https://github.com/pytorch/pytorch/blob/master/test/quantization/core/experimental/test_quantized_tensor.py
- https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/test/quantized_test.cpp
- <https://github.com/pytorch/pytorch/blob/master/aten/src/ATen/native/quantized/README.md>
- <https://pytorch.org/blog/introduction-to-quantization-on-pytorch/>
- <https://pytorch.org/docs/stable/quantization.html>
- Deep Dive on PyTorch Quantization - Chris Gottbrath
YouTube: <https://www.youtube.com/watch?v=c3MT2qV5f9w>

9.8 android

9.8.1 References

- <https://zhuanlan.zhihu.com/p/54665674>
- Pytorch model to Caffe & ncnn
<https://github.com/starimeL/PytorchConverter>

9.9 onnx

9.9.1 Install

```
pip install onnx onnxruntime
pip install netron # for visualization
# Or go to https://netron.app/
```

API references

- <https://github.com/onnx/onnx/blob/main/docs/PythonAPIOverview.md>
- https://onnxruntime.ai/docs/api/python/api_summary.html#inferencesession

9.9.2 Hello

Listing 78: ./code/hello/ex0.py

```
1  #!/usr/bin/env python3
2
3  import torch
4  import torch.nn as nn
5
6
7  class Foo(nn.Module):
8      def __init__(self, i):
9          super().__init__()
10         self.relu = nn.ReLU()
11         self.i = 1
12
13     def forward(self, x):
14         if x.sum().item() > 0:
15             return self.relu(x + 1)
16         else:
17             return self.relu(x + 2)
18
19
20 def main():
21     f = Foo(1)
22     f.eval() # f.train(False)
23     f = torch.jit.script(f)
24
25     x = torch.rand(2, 3, 4)
26     # [N, T, C]
27     torch.onnx.export(
28         f,
29         x,
30         "f.onnx",
31         verbose=False,
32         input_names=["x"],
```

(continues on next page)

(continued from previous page)

```

33     output_names=["y"],
34     dynamic_axes={"x": {0: "batch_size", 1: "T"}, "y": [0, 1]},
35     # dynamic_axes={"x": [0, 1], "y": [0, 1]},
36 )
37
38
39 if __name__ == "__main__":
40     main()

```

Listing 79: ./code/hello/ex0-1.py

```

1  #!/usr/bin/env python3
2
3  import onnx
4
5
6  def main():
7      model = onnx.load("f.onnx")
8      # print(model)
9      # Check that the model is well formed
10     onnx.checker.check_model(model)
11     # Print a human readable representation of the graph
12     print(onnx.helper.printable_graph(model.graph))
13     onnx.save(model, "f2.onnx")
14
15
16 if __name__ == "__main__":
17     main()

```

Listing 80: ./code/hello/ex0-2.py

```

1  #!/usr/bin/env python3
2
3  import onnxruntime as ort
4  import numpy as np
5
6
7  def main():
8      # https://github.com/microsoft/onnxruntime/issues/10113
9      options = ort.SessionOptions()
10     options.inter_op_num_threads = 1
11     options.intra_op_num_threads = 1
12
13     ort_session = ort.InferenceSession("f.onnx", sess_options=options)
14
15     x = np.arange(24).reshape(2, 3, 4).astype(np.float32)
16     ortvalue = ort.OrtValue.ortvalue_from_numpy(x)
17     assert ortvalue.device_name() == "cpu"
18     assert list(ortvalue.shape()) == list(x.shape)
19     assert ortvalue.data_type() == "tensor(float)"
20     assert ortvalue.is_tensor() is True
21

```

(continues on next page)

(continued from previous page)

```

22     results = ort_session.run(["y"], {"x": ortvalue})
23     print(results)
24
25     ort_inputs = {ort_session.get_inputs()[0].name: x}
26     results = ort_session.run(["y"], ort_inputs)
27     print(results)
28
29     results = ort_session.run(["y"], {"x": x})
30     print(results)
31
32     # https://onnxruntime.ai/docs/api/python/api\_summary.html#onnxruntime.NodeArg
33     inputs = ort_session.get_inputs()
34     assert isinstance(inputs, list)
35     assert len(inputs) == 1
36     assert isinstance(inputs[0], ort.NodeArg)
37     print(inputs[0].name, inputs[0].type, inputs[0].shape)
38     assert inputs[0].name == "x"
39     assert inputs[0].type == "tensor(float)"
40     assert inputs[0].shape == ["batch_size", "T", 4]
41
42     outputs = ort_session.get_outputs()
43     assert isinstance(outputs, list)
44     assert isinstance(outputs[0], ort.NodeArg)
45     assert len(outputs) == 1
46     assert outputs[0].name == "y"
47     assert outputs[0].type == "tensor(float)"
48     assert outputs[0].shape == ["y_dynamic_axes_1", "y_dynamic_axes_2", 4]
49
50
51 if __name__ == "__main__":
52     main()

```

9.9.3 Multiple models

Listing 81: ./code/multiple-models/ex.py

```

1  #!/usr/bin/env python3
2
3  import torch
4  import torch.nn as nn
5  import onnx
6  import onnxruntime as ort
7  import numpy as np
8  import os
9
10
11 class Foo(nn.Module):
12     def forward(self, x):
13         return x + 1
14

```

(continues on next page)

(continued from previous page)

```
15
16 class Bar(nn.Module):
17     def forward(self, x):
18         return x - 1
19
20
21 def export_to_onnx():
22     x = torch.rand(2, 3, dtype=torch.float32)
23     f = Foo()
24     torch.onnx.export(
25         f,
26         x,
27         "f.onnx",
28         verbose=False,
29         input_names=["x1"],
30         output_names=["y1"],
31         dynamic_axes={
32             "x1": {0: "N", 1: "T"},
33             "y1": {0: "N", 1: "T"},
34         },
35     )
36
37     x = torch.rand(1, dtype=torch.float32)
38     b = Bar()
39     torch.onnx.export(
40         b,
41         x,
42         "b.onnx",
43         verbose=False,
44         input_names=["x2"],
45         output_names=["y2"],
46         dynamic_axes={
47             "x2": {0: "N"},
48             "y2": {0: "N"},
49         },
50     )
51
52
53 def merge_models():
54     f = onnx.load("f.onnx")
55     f = onnx.compose.add_prefix(f, prefix="f/")
56     b = onnx.load("b.onnx")
57     combined_model = onnx.compose.merge_models(f, b, io_map={})
58     onnx.save(combined_model, "all.onnx")
59
60
61 def test_merged_model():
62     # https://github.com/microsoft/onnxruntime/issues/10113
63     options = ort.SessionOptions()
64     options.inter_op_num_threads = 1
65     options.intra_op_num_threads = 1
66
```

(continues on next page)

(continued from previous page)

```

67 all_model = onnx.load("all.onnx")
68
69 extractor = onnx.utils.Extractor(all_model)
70
71 f = extractor.extract_model(input_names=["f/x1"], output_names=["f/y1"])
72 f_session = ort.InferenceSession(f.SerializeToString(), sess_options=options)
73 f_inputs = f_session.get_inputs()
74 f_out = f_session.run(["f/y1"], {"f/x1": np.array([[1, 3]], dtype=np.float32)})
75 print(f_out[0]) # [[2. 4.]]
76
77 b = extractor.extract_model(input_names=["x2"], output_names=["y2"])
78 b_session = ort.InferenceSession(b.SerializeToString(), sess_options=options)
79 b_inputs = b_session.get_inputs()
80 b_out = b_session.run(["y2"], {"x2": np.array([1, 3], dtype=np.float32)})
81 print(b_out[0]) # [0. 2.]
82
83
84 def main():
85     export_to_onnx()
86     merge_models()
87     test_merged_model()
88     os.remove("f.onnx")
89     os.remove("b.onnx")
90     os.remove("all.onnx")
91
92
93 if __name__ == "__main__":
94     main()

```

We can first merge multiple models into one and then extract them.

9.9.4 References

- (OPTIONAL) EXPORTING A MODEL FROM PYTORCH TO ONNX AND RUNNING IT USING ONNX RUNTIME
https://pytorch.org/tutorials/advanced/super_resolution_with_onnxruntime.html
- Dynamic dummy input when exporting a PyTorch model? #654
<https://github.com/onnx/onnx/issues/654>
- onnxruntime latest version segment fault #10113
<https://github.com/microsoft/onnxruntime/issues/10113>

9.10 nn.LSTM

See <https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html>

Listing 82: ./code/lstm-test.py

```

1  #!/usr/bin/env python3
2
3  import torch
4  import torch.nn as nn
5
6
7  """
8  self.lstm = LSTM(
9      input_size=2,
10     hidden_size=5,
11     num_layers=1,
12     bias=True,
13     proj_size=2,
14 )
15
16 lstm.weight_ih_l0 [20, 2]
17 lstm.weight_hh_l0 [20, 2]
18 lstm.bias_ih_l0 [20]
19 lstm.bias_hh_l0 [20]
20 lstm.weight_hr_l0 [2, 5]
21 """
22
23
24 class Foo(nn.Module):
25     def __init__(self):
26         super().__init__()
27         self.lstm = nn.LSTM(
28             input_size=3,
29             hidden_size=5,
30             num_layers=1,
31             bias=True,
32             proj_size=4,
33         )
34
35     def forward(self, x, h0, c0):
36         """
37         Args:
38             x:
39                 (T, N, H_in), H_in is input dimension of x
40             h0:
41                 (num_layers, N, H_out), H_out is proj_size
42             c0:
43                 (num_layers, N, H_cell), H_cell is hidden_dim
44         """
45         y, (hx, cx) = self.lstm(x, (h0, c0))
46         return y, hx, cx
47

```

(continues on next page)

(continued from previous page)

```

48
49 @torch.no_grad()
50 def main():
51     f = Foo()
52     dim_in = 3
53     dim_proj = 4
54     dim_hidden = 5
55     x = torch.rand(1, 1, dim_in)
56     h0 = torch.rand(1, 1, dim_proj)
57     c0 = torch.rand(1, 1, dim_hidden)
58     y, hx, cx = f(x, h0, c0)
59
60     w_ih = f.state_dict()["lstm.weight_ih_l0"]
61     w_hh = f.state_dict()["lstm.weight_hh_l0"]
62
63     b_ih = f.state_dict()["lstm.bias_ih_l0"]
64     b_hh = f.state_dict()["lstm.bias_hh_l0"]
65
66     w_hr = f.state_dict()["lstm.weight_hr_l0"]
67
68     w_ii, w_if, w_ig, w_io = w_ih.split(5, dim=0)
69     w_hi, w_hf, w_hg, w_ho = w_hh.split(5, dim=0)
70
71     b_ii, b_if, b_ig, b_io = b_ih.split(5, dim=0)
72     b_hi, b_hf, b_hg, b_ho = b_hh.split(5, dim=0)
73
74     print(y, hx, cx)
75     print(y.shape)
76     print(hx.shape)
77     print(cx.shape)
78
79     i_gate = (x @ w_ii.t() + b_ii + h0 @ w_hi.t() + b_hi).sigmoid()
80     f_gate = (x @ w_if.t() + b_if + h0 @ w_hf.t() + b_hf).sigmoid()
81     g_gate = (x @ w_ig.t() + b_ig + h0 @ w_hg.t() + b_hg).tanh()
82     o_gate = (x @ w_io.t() + b_io + h0 @ w_ho.t() + b_ho).sigmoid()
83     c = f_gate * c0 + i_gate * g_gate
84
85     h = o_gate * c.tanh()
86     h = h @ w_hr.t()
87
88     print(h, h, c)
89
90
91 if __name__ == "__main__":
92     torch.manual_seed(20220903)
93     main()

```


PYTHON

10.1 Install from Source

Go to <https://www.python.org/ftp/python/> to download python, e.g.,

```
wget https://www.python.org/ftp/python/3.8.0/Python-3.8.0.tar.xz
wget https://www.python.org/ftp/python/3.9.0/Python-3.9.0.tar.xz
wget https://www.python.org/ftp/python/3.10.0/Python-3.10.0.tar.xz

mkdir -p $HOME/software/python
tar xvf Python-3.8.0.tar.xz
cd Python-3.8.0
./configure --help
./configure --prefix=$HOME/software/python/3.8.0 --enable-shared --enable-optimizations
```

10.2 asyncio

10.2.1 asyncio.Future

10.2.2 iterator

See <https://peps.python.org/pep-0234/>

10.2.3 yield

10.2.4 Hello World

Exercise 1

Listing 1: ./code/hello_world/ex1.py

```
1 import asyncio
2
3
4 async def hello():
5     print("hello world")
```

(continues on next page)

(continued from previous page)

```
6
7
8 asyncio.run(hello())
```

Exercise 2

Listing 2: ./code/hello_world/ex2.py

```
1 import asyncio
2 import time
3
4
5 loop = asyncio.get_event_loop()
6
7
8 @asyncio.coroutine
9 def hello():
10     print(f"hello {time.strftime('%X')}")
11     yield from asyncio.sleep(1)
12     print(f"world {time.strftime('%X')}")
13
14
15 if __name__ == "__main__":
16     loop.run_until_complete(hello())
```

10.2.5 References

- PEP 234 – Iterators
<https://peps.python.org/pep-0234/>
- Why does defining `__getitem__` on a class make it iterable in python?
<https://localcoder.org/why-does-defining-getitem-on-a-class-make-it-iterable-in-python>
- PEP 255 – Simple Generators
<https://peps.python.org/pep-0255/>
- Curious Course on Coroutines and Concurrency
https://www.youtube.com/watch?v=Z_OAllhXziw&ab_channel=DavidBeazley
By David Beazley.
- Generator Tricks for Systems Programmers
<https://www.dabeaz.com/generators2/>
- Generators: The Final Frontier
<https://www.youtube.com/watch?v=5-qadlG7tWo&ab_channel=DavidBeazley>
By David Beazley.

10.2.6 TODOs

`asyncio.to_thread()` runs the function in an executor, where the default executor is a threadpool executor, which invokes `loop.run_in_executor()` indirectly.

How to set the executor of a loop? Maybe something related to `set_default_executor`?

If we want to schedule a callback to run in the loop from the C++ code, we can use `loop.call_soon_safe()` method.

10.3 argv

From the doc <https://docs.python.org/3/library/sys.html>:

The **list** of command line arguments passed to a Python script. `argv[0]` **is** the script name (it **is** operating system dependent whether this **is** a full pathname **or not**). If the command was executed using the `-c` command line option to the interpreter, `argv[0]` **is set** to the string `'-c'`. If no script name was passed to the Python interpreter, `argv[0]` **is** the empty string.

Note that `argv` is at least of size 1, though `argv[0]` may be an empty string.

```
import sys
print(sys.argv)
```

10.4 TODO

Python with zeroMQ (c extension)

10.5 time

```
import time
print(f'Started at {time.strftime("%X")}')
# do something
print(f'Finished at {time.strftime("%X")}')
```

10.6 Numbers

10.6.1 binary representation

Listing 3: `./code/numbers/representations.py`

```
1 print(bin(1)) # 0b1
2 print(bin(3)) # 0b11
3 print(bin(255)) # 0b11111111
4 print(bin(256)) # 0b100000000
```

(continues on next page)

(continued from previous page)

```

5 assert isinstance(bin(1), str)
6 assert int("11", base=2) == 3
7 assert int("0b11", base=0) == 3
8 assert hex(2) == "0x2"
9 assert hex(10) == "0xa"
10
11 assert oct(10) == "0o12"
12 assert int("12", base=8) == 10
13 assert int("0o12", base=0) == 10
14
15 assert 1_000 == 1000
16 assert 1_000_000 == 1000000

```

10.7 str

10.7.1 format

See <https://docs.python.org/3/library/string.html#formatspec> and <https://peps.python.org/pep-3101/>

Listing 4: ./code/str/format.py

```

1 a = 1
2 b = 2
3 c = 3
4 assert "{}".format(a) == "1"
5 assert "{}".format(b) == "2"
6 assert "{0} {1} {foo}".format(a, b, foo=c) == "1 2 3"
7
8 # 1 - the first positional argument (counting from 0)
9 # foo - it is a keyword argument
10 # 0 - the zeros positional
11 assert "{1} {foo} {0}".format(a, b, foo=c) == "2 3 1"
12
13 assert "{0} {1} {0} {0}".format(a, b) == "1 2 1 1"
14
15 assert "skip braces {0} {}".format(a) == "skip braces 1 {"
16 print("{}") # {}
17 try:
18     print("{} {}".format(a))
19 except IndexError as e:
20     assert str(e) == "Replacement index 1 out of range for positional args tuple"
21
22 assert "{0:2}".format(a) == " 1"
23 assert "{0:02}".format(a) == "01"
24 assert "{0:03}".format(a) == "001"
25 assert "{0:1}".format(-1) == "-1"
26 assert "{0:2}".format(-1) == "-1"
27 assert "{0:3}".format(-1) == "-1"
28 assert "{0:03}".format(-1) == "-01"
29

```

(continues on next page)

(continued from previous page)

```
30 assert "{0:.2f}".format(0.5) == "0.50"  
31 assert "{0:.3f}".format(0.5) == "0.500"
```

10.8 enum

10.8.1 Hello

See <https://docs.python.org/3.11/howto/enum.html>

Enum

Note:

- It is iterable, i.e., supports `__iter__`
- name and value
- alias and `@unique`.
- `__members__`.
- `str` and `repr`.
- `auto`

Listing 5: `./code/hello/ex1.py`

```
1 from enum import Enum  
2  
3  
4 class Color(Enum):  
5     RED = 1  
6     GREEN = 2  
7     BLUE = 3  
8     # BLUE = 4 # TypeError: Attempted to reuse key: 'BLUE'  
9     ALIAS_FOR_RED = 1 # Use @unique to disallow this  
10    MAX_COLOR = 4 # Note the naming convention  
11  
12  
13 assert isinstance(Color.RED, Color)  
14  
15 assert str(Color(1)) == "Color.RED"  
16 assert str(repr(Color(1))) == "<Color.RED: 1>"  
17  
18 assert Color.RED.name == "RED"  
19 assert Color.BLUE.value == 3  
20  
21 print(list(Color))  
22 print(type(list(Color)[0]))  
23 for c in Color:  
24     print(c, type(c))  
25
```

(continues on next page)

(continued from previous page)

```

26 """
27 [<Color.RED: 1>, <Color.GREEN: 2>, <Color.BLUE: 3>, <Color.MAX_COLOR: 4>]
28 <enum 'Color'>
29 Color.RED <enum 'Color'>
30 Color.GREEN <enum 'Color'>
31 Color.BLUE <enum 'Color'>
32 Color.MAX_COLOR <enum 'Color'>
33 """
34
35 assert Color(1) == Color.RED
36 assert Color["RED"] == Color.RED
37 assert Color["ALIAS_FOR_RED"] == Color.RED
38
39 print(Color.__members__)
40 """
41 {'RED': <Color.RED: 1>, 'GREEN': <Color.GREEN: 2>, 'BLUE': <Color.BLUE: 3>, 'ALIAS_FOR_RED':
42  ↳<Color.RED: 1>, 'MAX_COLOR': <Color.MAX_COLOR: 4>}
43 """

```

Flag

Listing 6: ./code/hello/ex2.py

```

1 from enum import Flag
2
3
4 class Weekday(Flag):
5     MONDAY = 1 << 0
6     TUESDAY = 1 << 1
7     WEDNESDAY = 1 << 2
8     THURSDAY = 1 << 3
9     FRIDAY = 1 << 4
10    SATURDAY = 1 << 5
11    SUNDAY = 1 << 6
12
13
14 assert Weekday.MONDAY.value == 1
15 assert Weekday.TUESDAY.value == 2
16 assert Weekday.WEDNESDAY.value == 4
17 assert Weekday.THURSDAY.value == 8
18 assert Weekday.FRIDAY.value == 16
19 assert Weekday.SATURDAY.value == 32
20 assert Weekday.SUNDAY.value == 64
21
22 weekend = Weekday.SATURDAY | Weekday.SUNDAY
23 print(weekend) # Weekday.SUNDAY|SATURDAY
24 print(repr(weekend)) # <Weekday.SUNDAY|SATURDAY: 96>
25 assert Weekday.SATURDAY in weekend
26 assert Weekday.SUNDAY in weekend
27 assert Weekday.MONDAY not in weekend

```

auto

Listing 7: ./code/hello/ex3.py

```
1 from enum import Enum, Flag, auto
2
3
4 class Weekday(Flag):
5     MONDAY = auto() # start from 1
6     TUESDAY = auto()
7     WEDNESDAY = auto()
8     THURSDAY = auto()
9     FRIDAY = auto()
10    SATURDAY = 128
11    SUNDAY = auto()
12
13
14 assert Weekday.MONDAY.value == 1
15 assert Weekday.TUESDAY.value == 2
16 assert Weekday.WEDNESDAY.value == 4
17 assert Weekday.THURSDAY.value == 8
18 assert Weekday.FRIDAY.value == 16
19 assert Weekday.SATURDAY.value == 128
20 assert Weekday.SUNDAY.value == 256
21
22
23 class Color(Enum):
24     RED = auto() # start from 1
25     GREEN = auto()
26     BLUE = auto()
27     YELLOW = 10
28     WHITE = auto()
29
30
31 assert Color.RED.value == 1
32 assert Color.GREEN.value == 2
33 assert Color.BLUE.value == 3
34 assert Color.YELLOW.value == 10
35 assert Color.WHITE.value == 11
```

10.9 socket

10.9.1 AddressFamily

It is an IntEnum and all of its members are exported to socket.

Listing 8: ./code/address-family.py

```
1 import socket
2
3 print(list(socket.AddressFamily))
```

(continues on next page)

(continued from previous page)

```

4  """
5  [<AddressFamily.AF_UNSPEC: 0>, <AddressFamily.AF_UNIX: 1>,
6  <AddressFamily.AF_INET: 2>, <AddressFamily.AF_AX25: 3>,
7  <AddressFamily.AF_IPX: 4>, <AddressFamily.AF_APPLETALK: 5>,
8  <AddressFamily.AF_NETROM: 6>, <AddressFamily.AF_BRIDGE: 7>,
9  <AddressFamily.AF_ATMPVC: 8>, <AddressFamily.AF_X25: 9>,
10 <AddressFamily.AF_INET6: 10>, <AddressFamily.AF_ROSE: 11>,
11 <AddressFamily.AF_NETBEUI: 13>, <AddressFamily.AF_SECURITY: 14>,
12 <AddressFamily.AF_KEY: 15>, <AddressFamily.AF_NETLINK: 16>,
13 <AddressFamily.AF_PACKET: 17>, <AddressFamily.AF_ASH: 18>,
14 <AddressFamily.AF_ECONET: 19>, <AddressFamily.AF_ATMSVC: 20>,
15 <AddressFamily.AF_RDS: 21>, <AddressFamily.AF_SNA: 22>,
16 <AddressFamily.AF_IRDA: 23>, <AddressFamily.AF_PPPOX: 24>,
17 <AddressFamily.AF_WANPIPE: 25>, <AddressFamily.AF_LLC: 26>,
18 <AddressFamily.AF_CAN: 29>, <AddressFamily.AF_TIPC: 30>,
19 <AddressFamily.AF_BLUETOOTH: 31>, <AddressFamily.AF_ALG: 38>,
20 <AddressFamily.AF_VSOCK: 40>, <AddressFamily.AF_QIPCRTR: 42>]
21 """
22
23 assert socket.AF_UNIX == socket.AddressFamily.AF_UNIX
24 assert socket.AF_INET == socket.AddressFamily.AF_INET

```

10.9.2 SocketKind

It is an `IntEnum` and all of its members are exported to `socket`.

Listing 9: `./code/socket-kind.py`

```

1  import socket
2
3  print(list(socket.SocketKind))
4  """
5  [<SocketKind.SOCK_STREAM: 1>, <SocketKind.SOCK_DGRAM: 2>,
6  <SocketKind.SOCK_RAW: 3>, <SocketKind.SOCK_RDM: 4>,
7  <SocketKind.SOCK_SEQPACKET: 5>, <SocketKind.SOCK_NONBLOCK: 2048>,
8  <SocketKind.SOCK_CLOEXEC: 524288>]
9  """
10
11 assert socket.SOCK_STREAM == socket.SocketKind.SOCK_STREAM
12 assert socket.SOCK_DGRAM == socket.SocketKind.SOCK_DGRAM

```

10.9.3 struct sockaddr_in

See also

- https://www.gta.ufrj.br/ensino/eel878/sockets/sockaddr_inman.html
- <https://man7.org/linux/man-pages/man7/ip.7.html>

Listing 10: ./code/sockaddr_in.h

```

1 // https://github.com/lattera/glibc/blob/master/bits/sockaddr.h
2 /* POSIX.1g specifies this type name for the `sa_family' member. */
3 typedef unsigned short int sa_family_t;
4
5 #define __SOCKADDR_COMMON(sa_prefix) sa_family_t sa_prefix##family
6
7 // https://github.com/lattera/glibc/blob/master/bits/socket.h
8
9 struct sockaddr {
10     __SOCKADDR_COMMON(sa_); /* Common data: address family and length. */
11     char sa_data[14];      /* Address data. */
12 };
13
14 // https://github.com/lattera/glibc/blob/master/inet/netinet/in.h
15 struct sockaddr_in {
16     __SOCKADDR_COMMON(sin_);
17     in_port_t sin_port;    /* Port number. */
18     struct in_addr sin_addr; /* Internet address. */
19
20     /* Pad to size of `struct sockaddr'. */
21     unsigned char sin_zero[sizeof(struct sockaddr) - __SOCKADDR_COMMON_SIZE -
22                               sizeof(in_port_t) - sizeof(struct in_addr)];
23 };
24
25 typedef uint32_t in_addr_t;
26 struct in_addr {
27     in_addr_t s_addr;
28 };
29
30 /* Address to accept any incoming messages. */
31 #define INADDR_ANY ((in_addr_t)0x00000000)
32 /* Address to send to all hosts. */
33 #define INADDR_BROADCAST ((in_addr_t)0xffffffff)
34 /* Address indicating an error return. */
35 #define INADDR_NONE ((in_addr_t)0xffffffff)
36
37 /* Network number for local host loopback. */
38 #define IN_LOOPBACKNET 127
39 /* Address to loopback in software to local host. */
40 #ifndef INADDR_LOOPBACK
41 #define INADDR_LOOPBACK ((in_addr_t)0x7f000001) /* Inet 127.0.0.1. */
42 #endif

```

10.9.4 AddressInfo

Listing 11: ./code/address-info.py

```

1 import socket
2
3 print(list(socket.AddressInfo))
4 """
5 [<AddressInfo.AI_PASSIVE: 1>, <AddressInfo.AI_CANONNAME: 2>,
6 <AddressInfo.AI_NUMERICHOST: 4>, <AddressInfo.AI_V4MAPPED: 8>,
7 <AddressInfo.AI_ALL: 16>, <AddressInfo.AI_ADDRCONFIG: 32>,
8 <AddressInfo.AI_NUMERICSERV: 1024>]
9 """
10 assert socket.AI_PASSIVE == socket.AddressInfo.AI_PASSIVE

```

10.9.5 inet_pton

https://man7.org/linux/man-pages/man3/inet_pton.3.html

Representation format to network address.

The resulting network address is in network order, i.e., big endian.

Listing 12: ./code/inet_pton.c

```

1 #include <arpa/inet.h>
2 #include <stdio.h>
3
4 int main() {
5     struct in_addr addr;
6     int res = inet_pton(AF_INET, "192.168.1.2", &addr);
7     printf("%08x\n", addr.s_addr);
8     printf("192: %x\n", 192);
9     printf("168: %x\n", 168);
10    printf("1: %x\n", 1);
11    printf("2: %x\n", 2);
12    return 0;
13 }
14 #if 0
15 ./inet_pton
16 0201a8c0
17 192: c0
18 168: a8
19 1: 1
20 2: 2
21 #endif

```

Its implementation can be found at https://github.com/bminor/glibc/blob/master/resolv/inet_pton.c

Listing 13: ./code/inet_pton_impl.c

```

1 // See https://github.com/bminor/glibc/blob/master/resolv/inet_pton.c
2 //

```

(continues on next page)

(continued from previous page)

```

3  /* Copyright (C) 1996-2022 Free Software Foundation, Inc.
4     This file is part of the GNU C Library.
5
6     The GNU C Library is free software; you can redistribute it and/or
7     modify it under the terms of the GNU Lesser General Public
8     License as published by the Free Software Foundation; either
9     version 2.1 of the License, or (at your option) any later version.
10
11    The GNU C Library is distributed in the hope that it will be useful,
12    but WITHOUT ANY WARRANTY; without even the implied warranty of
13    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
14    Lesser General Public License for more details.
15
16    You should have received a copy of the GNU Lesser General Public
17    License along with the GNU C Library; if not, see
18    <https://www.gnu.org/licenses/>.  */
19
20  /*
21   * Copyright (c) 1996,1999 by Internet Software Consortium.
22   *
23   * Permission to use, copy, modify, and distribute this software for any
24   * purpose with or without fee is hereby granted, provided that the above
25   * copyright notice and this permission notice appear in all copies.
26   *
27   * THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS
28   * ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
29   * OF MERCHANTABILITY AND FITNESS.  IN NO EVENT SHALL INTERNET SOFTWARE
30   * CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL
31   * DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR
32   * PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
33   * ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
34   * SOFTWARE.
35   */
36
37  #include <arpa/inet.h>
38  #include <arpa/nameser.h>
39  #include <ctype.h>
40  #include <errno.h>
41  #include <netinet/in.h>
42  #include <resolv/resolv-internal.h>
43  #include <string.h>
44  #include <sys/socket.h>
45  #include <sys/types.h>
46
47  static int inet_pton4 (const char *src, const char *src_end, u_char *dst);
48  static int inet_pton6 (const char *src, const char *src_end, u_char *dst);
49
50  int
51  __inet_pton_length (int af, const char *src, size_t srclen, void *dst)
52  {
53      switch (af)
54      {

```

(continues on next page)

(continued from previous page)

```

55     case AF_INET:
56         return inet_pton4 (src, src + srclen, dst);
57     case AF_INET6:
58         return inet_pton6 (src, src + srclen, dst);
59     default:
60         __set_errno (EAFNOSUPPORT);
61         return -1;
62     }
63 }
64 libc_hidden_def (__inet_pton_length)
65
66 /* Like __inet_pton_length, but use strlen (SRC) as the length of
67 SRC. */
68 int
69 __inet_pton (int af, const char *src, void *dst)
70 {
71     return __inet_pton_length (af, src, strlen (src), dst);
72 }
73 libc_hidden_def (__inet_pton)
74 weak_alias (__inet_pton, inet_pton)
75 libc_hidden_weak (inet_pton)
76
77 /* Like inet_aton but without all the hexadecimal, octal and shorthand
78 (and trailing garbage is not ignored). Return 1 if SRC is a valid
79 dotted quad, else 0. This function does not touch DST unless it's
80 returning 1.
81 Author: Paul Vixie, 1996. */
82 static int
83 inet_pton4 (const char *src, const char *end, unsigned char *dst)
84 {
85     int saw_digit, octets, ch;
86     unsigned char tmp[NS_INADDRSZ], *tp;
87
88     saw_digit = 0;
89     octets = 0;
90     *(tp = tmp) = 0;
91     while (src < end)
92     {
93         ch = *src++;
94         if (ch >= '0' && ch <= '9')
95         {
96             unsigned int new = *tp * 10 + (ch - '0');
97
98             if (saw_digit && *tp == 0)
99                 return 0;
100             if (new > 255)
101                 return 0;
102             *tp = new;
103             if (! saw_digit)
104             {
105                 if (++octets > 4)
106                     return 0;

```

(continues on next page)

(continued from previous page)

```

107         saw_digit = 1;
108     }
109 }
110 else if (ch == '.' && saw_digit)
111 {
112     if (octets == 4)
113         return 0;
114     *++tp = 0;
115     saw_digit = 0;
116 }
117 else
118     return 0;
119 }
120 if (octets < 4)
121     return 0;
122 memcpy (dst, tmp, NS_INADDRSZ);
123 return 1;
124 }
125
126 /* Return the value of CH as a hexademical digit, or -1 if it is a
127 different type of character. */
128 static int
129 hex_digit_value (char ch)
130 {
131     if ('0' <= ch && ch <= '9')
132         return ch - '0';
133     if ('a' <= ch && ch <= 'f')
134         return ch - 'a' + 10;
135     if ('A' <= ch && ch <= 'F')
136         return ch - 'A' + 10;
137     return -1;
138 }
139
140 /* Convert presentation-level IPv6 address to network order binary
141 form. Return 1 if SRC is a valid [RFC1884 2.2] address, else 0.
142 This function does not touch DST unless it's returning 1.
143 Author: Paul Vixie, 1996. Inspired by Mark Andrews. */
144 static int
145 inet_pton6 (const char *src, const char *src_endp, unsigned char *dst)
146 {
147     unsigned char tmp[NS_IN6ADDRSZ], *tp, *endp, *colonp;
148     const char *curtok;
149     int ch;
150     size_t xdigits_seen;      /* Number of hex digits since colon. */
151     unsigned int val;
152
153     tp = memset (tmp, '\0', NS_IN6ADDRSZ);
154     endp = tp + NS_IN6ADDRSZ;
155     colonp = NULL;
156
157     /* Leading :: requires some special handling. */
158     if (src == src_endp)

```

(continues on next page)

(continued from previous page)

```

159     return 0;
160     if (*src == ':')
161     {
162         ++src;
163         if (src == src_endp || *src != ':')
164             return 0;
165     }
166
167     curtok = src;
168     xdigits_seen = 0;
169     val = 0;
170     while (src < src_endp)
171     {
172         ch = *src++;
173         int digit = hex_digit_value (ch);
174         if (digit >= 0)
175         {
176             if (xdigits_seen == 4)
177                 return 0;
178             val <= 4;
179             val |= digit;
180             if (val > 0xffff)
181                 return 0;
182             ++xdigits_seen;
183             continue;
184         }
185         if (ch == ':')
186         {
187             curtok = src;
188             if (xdigits_seen == 0)
189             {
190                 if (colonp)
191                     return 0;
192                 colonp = tp;
193                 continue;
194             }
195             else if (src == src_endp)
196                 return 0;
197             if (tp + NS_INT16SZ > endp)
198                 return 0;
199             *tp++ = (unsigned char) (val >> 8) & 0xff;
200             *tp++ = (unsigned char) val & 0xff;
201             xdigits_seen = 0;
202             val = 0;
203             continue;
204         }
205         if (ch == '.' && ((tp + NS_INADDRSZ) <= endp)
206             && inet_pton4 (curtok, src_endp, tp) > 0)
207         {
208             tp += NS_INADDRSZ;
209             xdigits_seen = 0;
210             break; /* '\0' was seen by inet_pton4. */

```

(continues on next page)

(continued from previous page)

```

211     }
212     return 0;
213 }
214 if (xdigits_seen > 0)
215 {
216     if (tp + NS_INT16SZ > endp)
217         return 0;
218     *tp++ = (unsigned char) (val >> 8) & 0xff;
219     *tp++ = (unsigned char) val & 0xff;
220 }
221 if (colonp != NULL)
222 {
223     /* Replace :: with zeros. */
224     if (tp == endp)
225         /* :: would expand to a zero-width field. */
226         return 0;
227     size_t n = tp - colonp;
228     memmove (endp - n, colonp, n);
229     memset (colonp, 0, endp - n - colonp);
230     tp = endp;
231 }
232 if (tp != endp)
233     return 0;
234 memcpy (dst, tmp, NS_IN6ADDRSZ);
235 return 1;
236 }

```

10.9.6 inet_ntop

Network address to representation format.

See https://man7.org/linux/man-pages/man3/inet_ntop.3.html

Listing 14: ./code/inet_ntop.c

```

1  #include <arpa/inet.h>
2  #include <stdio.h>
3
4  int main() {
5      struct in_addr addr;
6      uint8_t *p = (uint8_t *)&addr.s_addr;
7      p[0] = 192;
8      p[1] = 168;
9      p[2] = 1;
10     p[3] = 2;
11     char buf[INET_ADDRSTRLEN];
12     const char *ret = inet_ntop(AF_INET, &addr.s_addr, buf, sizeof(buf));
13     printf("%s\n", buf);
14     printf("%p, %p\n", buf, ret);
15     return 0;
16 }

```

(continues on next page)

(continued from previous page)

```

17 #if 0
18 192.168.1.2
19 0x7ffc808b5e80, 0x7ffc808b5e80
20 #endif

```

Its implementation can be found at https://github.com/bminor/glibc/blob/master/resolv/inet_ntop.c

Listing 15: ./code/inet_ntop_impl.c

```

1 // https://github.com/bminor/glibc/blob/master/resolv/inet_ntop.c
2 /*
3  * Copyright (c) 1996-1999 by Internet Software Consortium.
4  *
5  * Permission to use, copy, modify, and distribute this software for any
6  * purpose with or without fee is hereby granted, provided that the above
7  * copyright notice and this permission notice appear in all copies.
8  *
9  * THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS
10 * ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
11 * OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE
12 * CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL
13 * DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR
14 * PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
15 * ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
16 * SOFTWARE.
17 */
18
19 #include <sys/param.h>
20 #include <sys/types.h>
21 #include <sys/socket.h>
22
23 #include <netinet/in.h>
24 #include <arpa/inet.h>
25 #include <arpa/nameser.h>
26
27 #include <errno.h>
28 #include <stdio.h>
29 #include <string.h>
30
31 #ifdef SPRINTF_CHAR
32 # define SPRINTF(x) strlen(sprintf/**/x)
33 #else
34 # define SPRINTF(x) ((size_t)sprintf x)
35 #endif
36
37 /*
38  * WARNING: Don't even consider trying to compile this on a system where
39  * sizeof(int) < 4.  sizeof(int) > 4 is fine; all the world's not a VAX.
40  */
41
42 static const char *inet_ntop4 (const u_char *src, char *dst, socklen_t size);
43 static const char *inet_ntop6 (const u_char *src, char *dst, socklen_t size);

```

(continues on next page)

(continued from previous page)

```

44
45 /* char *
46  * inet_ntop(af, src, dst, size)
47  *      convert a network format address to presentation format.
48  * return:
49  *      pointer to presentation format address ('dst'), or NULL (see errno).
50  * author:
51  *      Paul Vixie, 1996.
52  */
53 const char *
54 inet_ntop (int af, const void *src, char *dst, socklen_t size)
55 {
56     switch (af) {
57     case AF_INET:
58         return (inet_ntop4(src, dst, size));
59     case AF_INET6:
60         return (inet_ntop6(src, dst, size));
61     default:
62         __set_errno (EAFNOSUPPORT);
63         return (NULL);
64     }
65     /* NOTREACHED */
66 }
67 libc_hidden_def (inet_ntop)
68
69 /* const char *
70  * inet_ntop4(src, dst, size)
71  *      format an IPv4 address
72  * return:
73  *      'dst' (as a const)
74  * notes:
75  *      (1) uses no statics
76  *      (2) takes a u_char* not an in_addr as input
77  * author:
78  *      Paul Vixie, 1996.
79  */
80 static const char *
81 inet_ntop4 (const u_char *src, char *dst, socklen_t size)
82 {
83     static const char fmt[] = "%u.%u.%u.%u";
84     char tmp[sizeof "255.255.255.255"];
85
86     if (SPRINTF((tmp, fmt, src[0], src[1], src[2], src[3])) >= size) {
87         __set_errno (ENOSPC);
88         return (NULL);
89     }
90     return strcpy(dst, tmp);
91 }
92
93 /* const char *
94  * inet_ntop6(src, dst, size)
95  *      convert IPv6 binary address into presentation (printable) format

```

(continues on next page)

(continued from previous page)

```

96  * author:
97  *      Paul Vixie, 1996.
98  */
99  static const char *
100 inet_ntop6 (const u_char *src, char *dst, socklen_t size)
101 {
102     /*
103      * Note that int32_t and int16_t need only be "at least" large enough
104      * to contain a value of the specified size.  On some systems, like
105      * Crays, there is no such thing as an integer variable with 16 bits.
106      * Keep this in mind if you think this function should have been coded
107      * to use pointer overlays.  All the world's not a VAX.
108      */
109     char tmp[sizeof "ffff:ffff:ffff:ffff:ffff:ffff:255.255.255.255"], *tp;
110     struct { int base, len; } best, cur;
111     u_int words[NS_IN6ADDRSZ / NS_INT16SZ];
112     int i;
113
114     /*
115      * Preprocess:
116      *      Copy the input (bytewise) array into a wordwise array.
117      *      Find the longest run of 0x00's in src[] for :: shorthanding.
118      */
119     memset(words, '\\0', sizeof words);
120     for (i = 0; i < NS_IN6ADDRSZ; i += 2)
121         words[i / 2] = (src[i] << 8) | src[i + 1];
122     best.base = -1;
123     cur.base = -1;
124     best.len = 0;
125     cur.len = 0;
126     for (i = 0; i < (NS_IN6ADDRSZ / NS_INT16SZ); i++) {
127         if (words[i] == 0) {
128             if (cur.base == -1)
129                 cur.base = i, cur.len = 1;
130             else
131                 cur.len++;
132         } else {
133             if (cur.base != -1) {
134                 if (best.base == -1 || cur.len > best.len)
135                     best = cur;
136                 cur.base = -1;
137             }
138         }
139     }
140     if (cur.base != -1) {
141         if (best.base == -1 || cur.len > best.len)
142             best = cur;
143     }
144     if (best.base != -1 && best.len < 2)
145         best.base = -1;
146
147     /*

```

(continues on next page)

(continued from previous page)

```

148     * Format the result.
149     */
150     tp = tmp;
151     for (i = 0; i < (NS_IN6ADDRSZ / NS_INT16SZ); i++) {
152         /* Are we inside the best run of 0x00's? */
153         if (best.base != -1 && i >= best.base &&
154             i < (best.base + best.len)) {
155             if (i == best.base)
156                 *tp++ = ':';
157             continue;
158         }
159         /* Are we following an initial run of 0x00s or any real hex? */
160         if (i != 0)
161             *tp++ = ':';
162         /* Is this address an encapsulated IPv4? */
163         if (i == 6 && best.base == 0 &&
164             (best.len == 6 || (best.len == 5 && words[5] == 0xffff))) {
165             if (!inet_ntop4(src+12, tp, sizeof tmp - (tp - tmp)))
166                 return (NULL);
167             tp += strlen(tp);
168             break;
169         }
170         tp += SPRINTF((tp, "%x", words[i]));
171     }
172     /* Was it a trailing run of 0x00's? */
173     if (best.base != -1 && (best.base + best.len) ==
174         (NS_IN6ADDRSZ / NS_INT16SZ))
175         *tp++ = ':';
176     *tp++ = '\0';
177
178     /*
179     * Check for overflow, copy, and we're done.
180     */
181     if ((socklen_t)(tp - tmp) > size) {
182         __set_errno (ENOSPC);
183         return (NULL);
184     }
185     return strcpy(dst, tmp);
186 }

```

10.9.7 Echo Server and Client

Server

Listing 16: ./code/echo-hello/server.py

```

1  #!/usr/bin/env python3
2  import socket
3  import threading
4

```

(continues on next page)

(continued from previous page)

```
5 # nc localhost 6006
6
7
8 def run_server():
9     sock = socket.socket(family=socket.AF_INET, type=socket.SOCK_STREAM)
10    sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
11    sock.bind("", 6006)
12    # sock.setblocking(False) # It returns socket.BlockingIOError
13    sock.listen(2) # backlog is 2
14    while True:
15        client_sock, addr = sock.accept()
16        assert isinstance(client_sock, socket.socket)
17        assert isinstance(addr, tuple)
18        assert isinstance(addr[0], str)
19        assert isinstance(addr[1], int)
20        print("Connected from", addr) # Connected from ('127.0.0.1', 54266)
21        threading.Thread(target=handle_client, args=(client_sock,)).start()
22
23
24 def handle_client(sock: socket.socket):
25     while True:
26         data = sock.recv(1024)
27         if not data:
28             break
29         sock.sendall(data.decode("utf-8").upper().encode())
30     print("Disconnected from", sock.getpeername())
31     sock.close()
32
33
34 if __name__ == "__main__":
35     run_server()
```

To test the server, use `nc localhost 6006` or use the following client.

Client

Listing 17: ./code/echo-hello/client.py

```

1 def main():
2     sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
3     sock.connect(("localhost", 6006))
4     for i in range(5):
5         sock.send("hello".encode())
6         b = sock.recv(1024)
7         print(b.decode())
8         time.sleep(0.5)

```

Server2

With `concurrent.futures.ThreadPoolExecutor`.

Listing 18: ./code/echo-hello/server2.py

```

1  #!/usr/bin/env python3
2  import socket
3  import threading
4  from concurrent.futures import ThreadPoolExecutor
5
6  # nc localhost 6006
7  pool = ThreadPoolExecutor(max_workers=3)
8
9
10 def run_server():
11     sock = socket.socket(family=socket.AF_INET, type=socket.SOCK_STREAM)
12     sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
13     sock.bind("", 6006)
14     # sock.setblocking(False) # It returns socket.BlockingIOError
15     sock.listen(2) # backlog is 2
16     while True:
17         client_sock, addr = sock.accept()
18         assert isinstance(client_sock, socket.socket)
19         assert isinstance(addr, tuple)
20         assert isinstance(addr[0], str)
21         assert isinstance(addr[1], int)
22         print("Connected from", addr) # Connected from ('127.0.0.1', 54266)
23         pool.submit(handle_client, client_sock)
24
25
26 def handle_client(sock: socket.socket):
27     while True:
28         data = sock.recv(1024)
29         if not data:
30             break
31         sock.sendall(data.decode("utf-8").upper().encode())
32     print("Disconnected from", sock.getpeername())
33     sock.close()
34
35

```

(continues on next page)

(continued from previous page)

```
36 if __name__ == "__main__":  
37     run_server()
```

10.9.8 TODOs

- Beej's Guide to Network Programming Using Internet Sockets
<https://www.gta.ufrj.br/ensino/eel878/sockets/index.html>
- LWN.net Weekly Edition Archives
<https://lwn.net/Archives/>

10.10 numpy

10.10.1 Basics

Listing 19: ./code/basics.py

```
#!/usr/bin/env python3  
import numpy as np  
  
def test_attr():  
    a = np.array([[1, 2], [3, 4]], dtype=np.float32)  
    assert isinstance(a, np.ndarray)  
  
    assert a.dtype == np.float32  
    assert isinstance(a.dtype, np.dtype)  
    assert a.dtype.name == "float32"  
  
    # number of axes  
    assert a.ndim == 2  
  
    assert a.shape == (2, 2)  
    assert len(a.shape) == a.ndim  
  
    # total number of elements is a.size  
    assert a.size == a.shape[0] * a.shape[1]  
  
    # each float occupies 4 bytes  
    assert a.itemsize == 4  
    assert isinstance(a.data, memoryview)  
  
def test_array_creation():  
    a = np.array([2, 3, 4])  
    assert a.dtype == np.int64  
    assert a.ndim == 1
```

(continues on next page)

(continued from previous page)

```
b = np.array([2.0, 3, 4])
assert b.dtype == np.float64

c = np.array([[1, 2], [3, 4]])
assert c.ndim == 2
assert c.shape == (2, 2)
assert c.dtype == np.int64

d = np.zeros((2, 3))
assert d.dtype == np.float64
assert d.shape == (2, 3)

f = np.ones((2, 2))
assert f.dtype == np.float64

# uninitialized content; containing garbage data.
e = np.empty((5, 5))
assert e.dtype == np.float64

g = np.empty((1, 2), dtype=np.int32)
assert g.dtype == np.int32

h = np.arange(3)
assert h.dtype == np.int64

# start=0, end=3
i = np.arange(3, dtype=np.int32)
assert i.dtype == np.int32
np.testing.assert_equal(i, np.array([0, 1, 2]))

# start=1, end=3
k = np.arange(1, 3)
np.testing.assert_equal(k, np.array([1, 2]))

# start=1, end=8, step=3
m = np.arange(1, 8, 3)
np.testing.assert_equal(m, np.array([1, 4, 7]))

# 4 numbers in the range [2, 5]
n = np.linspace(2, 5, 4)
assert n.dtype == np.float64
np.testing.assert_equal(n, np.array([2, 3, 4, 5]))

# note: The last number is included
m = np.linspace(2, 5, 3)
np.testing.assert_equal(m, np.array([2, 3.5, 5]))

def main():
    test_attr()
    test_array_creation()
```

(continues on next page)

(continued from previous page)

```
if __name__ == "__main__":  
    main()
```

10.11 pip

After creating a virtual environment with:

```
python3 -m venv my_env  
source ./my_env/bin/active
```

We can use:

```
curl https://bootstrap.pypa.io/get-pip.py  
python3 ./get-pip.py
```

to install pip

11.1 Install

11.1.1 formatter

Install <https://github.com/google/google-java-format>

```
wget https://github.com/google/google-java-format/releases/download/v1.15.0/google-java-format-1.15.0-all-deps.jar
```

Create a script with filename `google-java-format`:

```
#!/usr/bin/env bash

java -jar /ceph-sh0/fangjun/download/google-java-format-1.15.0-all-deps.jar $@
```

`chmod +x google-java-format` and add the path to `PATH`.

11.1.2 JDK

Go to <https://www.oracle.com/java/technologies/downloads/#java17> and download

```
wget https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.tar.gz
mkdir /ceph-fj/fangjun/software/
tar xvf jdk-17_linux-x64_bin.tar.gz -C /ceph-fj/fangjun/software
```

And then set the following environment variables:

```
export JAVA_HOME=/ceph-fj/fangjun/software/jdk-17.0.3
export PATH=$JAVA_HOME/bin:$JAVA_HOME
```

The source code of JDK can be download from

```
git clone https://github.com/openjdk/jdk.git
```

11.2 Hello world

Look at the code <https://github.com/openjdk/jdk/blob/master/src/java.base/share/classes/java/lang/System.java> to learn the coding style of java.

All primitive types, such as `int`, `char`, and `double` begin with a lowercase. All other types, such as `String` and `Math`, begin with an uppercase.

```
package java.lang;

// Class names looks like: MyClass
public final class System {}

// method name
private static native void registerNatives();

// constructor
private System() {}

// member variables
public static final InputStream in = null;
private static String notSupportedJnuEncoding;

// constants
private static final int NEVER = 1;
```

It using `//` and `/**/` as comments, same as C++.

Listing 1: Hello.java

```
// Usage 1:
//  java Hello.java
// Usage 2:
//  javac Hello.java
//  java Hello
//
// Note:
//  - "javac Hello.java" generates a file "Hello.class"
//  - "java Hello" takes as input "Hello.class" and executes it
//
// The class name Hello must match the filename Hello.java
// By convention, the class name looks like `MyClass`.
class Hello {
    // Note: How is array defined in java
    public static void main(String[] args) {
        System.out.println("hello world");
    }
} // There is no ';' here
// Every line of runnable code must be in some `class`
```

Listing 2: EqualTest.java

```
class EqualTest {
```

(continues on next page)

(continued from previous page)

```

public int i;

public EqualTest(int a) {
    this.i = a;
}

public boolean equals(Object anObject) {
    if (this == anObject) {
        return true;
    }
    if (anObject instanceof EqualTest) {
        return this.i == ((EqualTest) anObject).i;
    }
    return false;
}

public static void main(String[] args) {
    EqualTest e1 = new EqualTest(10);
    EqualTest e2 = new EqualTest(10);

    System.out.println(e1 == e2); // false, compare the reference
    System.out.println(e1 != e2); // true
    System.out.println(e1.equals(e2)); // true, compare the contained value
}
}

```

Listing 3: Variables.java

```

class Variables {
    public static void main(String[] args) {
        String msg = "Hello world";
        int i = 10;
        i++; // ok
        ++i; // ok
        // i-- // error, should be --i
        --i;
        int ii = i % 2;
        i = 10;
        ii = i / 3; // truncated
        System.out.println(ii); // 3
        ii += 2;
        System.out.println(ii); // 5
        // also has:
        // -=, &=, |=, >>=, <<=, ^=, /=, *=
        // Similar to C/C++

        /* float f = 1.25; // error: possible lossy conversion from double to float */
        // Caution: We have to use `1.25f` instead of `1.25` to assign a float variable
        float f = 1.25f;
        double d = 1.25;
        d = 1.25d; // also ok
    }
}

```

(continues on next page)

(continued from previous page)

```

// f = d; // error: possible lossy conversion from double to float
f = (float) d; // ok, explicit cast
d = f; // ok

char c = 'h'; // a character

i = 10;
System.out.println(msg);
System.out.println(i); // 10
System.out.println(f); // 1.25
System.out.println(d); // 1.25
System.out.println(c); // h

// define multiple variable in the same line
int i1 = 1, i2;
// System.out.println(i2); // variable i2 might not have been initialized
i2 = 3;
System.out.println(i2); // 3
boolean b = false;
System.out.println(b); // false
b = true;
// b = 1; // error: int cannot be converted to boolean
System.out.println(b); // true

// byte b0 = 128; // error: lossy conversion from int to byte
byte b0 = 127; // ok
/* b0 = -129; // error: lossy conversion from int to type */
b0 = -128; // ok

// short s0 = 32768; // error: lossy conversion from int to short
short s0 = 32767; // ok
// s0 = -32769; // error: lossy conversion from int to short
s0 = -32768; // ok

// other types: int, long, float, double, boolean, char
// Note: char has 2-byte
// boolean has 1-bit
// There are no unsigned integers !!

// ternary operator :?
System.out.println(2 > 3 ? "yes" : "no"); // no
}
}

```

Listing 4: Strings.java

```

class Strings {
    public static void main(String[] args) {
        String s = "abc";
        System.out.println(s.length()); // 3
        System.out.println(s.toUpperCase()); // ABC
        System.out.println(s.toUpperCase().toLowerCase()); // abc
    }
}

```

(continues on next page)

(continued from previous page)

```

System.out.println(s.indexOf("ab")); // 0
System.out.println(s.indexOf("a")); // 0
System.out.println(s.indexOf("bc")); // 1
System.out.println(s.indexOf("bca")); // when not found, it is -1

System.out.println(s.charAt(0) == 'a'); // true
System.out.println(s + "123" == "abc123"); // false, == compares the reference
System.out.println((s + "123").equals("abc123")); // true, compare the content
System.out.println(s.concat("123").equals("abc123")); // true, compare the content
System.out.println((s + 123).equals("abc123")); // true, compare the content
System.out.println((123 + s).equals("123abc")); // true, compare the content
s = "a\\b";
}
}
;

```

Listing 5: MathTest.java

```

class MathTest {
    public static void main(String[] args) {
        System.out.println(Math.max(2, 3)); // 3
        System.out.println(Math.max(2, 3.5)); // 3.5
        System.out.println(Math.min(2, 3.5)); // 2
        System.out.println(Math.sqrt(6.25)); // 2.5
        System.out.println(Math.sqrt(9)); // 3.0, note it is a double
        System.out.println(Math.abs(-9)); // -9
        System.out.println(Math.abs(-9.0)); // -9.0
        System.out.println(Math.random()); // a number in the range [0, 1)
    }
}

```

Listing 6: ArrayTest.java

```

class ArrayTest {
    public static void main(String[] args) {
        int[] a = {1, 2, 3};
        for (int i : a) {
            System.out.println(i);
        }
        /*
        1
        2
        3
        */
        System.out.println(a[0]); // 1
        System.out.println(a.length); // 3
        for (int i = 0; i != a.length; ++i) {
            System.out.println(a[i]);
        }
        /*
        1
        2

```

(continues on next page)

(continued from previous page)

```
3
    */
}
}
```

11.2.1 TODOs

1. How to define an array? How to iterate an array?
2. What methods does the `String` class have
3. What are final variables?
4. How to define a package?
5. What is static import?
6. What is a lambda expression?
7. How to write comments using javadoc?
8. How to generate documentation from javadoc?
9. How to use a for-each loop?
10. How to import a class from another file?

11.3 Reference

- <https://www.w3schools.com/java/default.asp>
- <https://docs.oracle.com/javase/tutorial/>
- <https://docs.oracle.com/en/java/javase/17/docs/api/index.html>
- <https://github.com/openjdk/jdk.git>

Clone it and you can find the source code in `src/java.base/share/classes/java/lang/System.java` for `java.lang.System`.

JAVASCRIPT

12.1 Hello world

- case sensitive
- numbers (all numbers are 64-bit floating type)
- strings (' ' and " "), there is no character type
- Like python, strings are immutable
- strings uses utf-16, like c#
- array
- boolean, true and false
- null and undefined
- string interpolation

```
`hello ${someVariable}`
```

- semicolon is optional. Better to always add it
- garbage collector
- == supports type conversion
- === does not support type conversion. (Recommended to use this one)

```
console.log('hello world')  
console.log(eval('3 + 5'))
```

To write multi-line javascript, use shift + Enter for a new line.

```
(function(){  
  "use strict";  
  /* Start of your code */  
  function greetMe(yourName) {  
    alert('Hello ' + yourName);  
  }  
  
  greetMe('World');  
  /* End of your code */  
})();
```

It is case sensitive. Statements are separated by `;`. Comments are the same as in C/C++.

12.1.1 variables

Listing 1: `./code/hello_world/variables.js`

```
1  // use const to define a constant
2  const pizza = true;
3  console.log(pizza); // true
4
5  var foo = 'foo';
6  if (foo) {
7      var foo = 'fooz'
8      console.log('foo is', foo); // foo is fooz
9  }
10
11 // note it outputs foo is fooz below since
12 // we use `var` inside the `if` statement
13 console.log('foo is', foo); // foo is fooz
14
15 var bar = 'bar';
16
17 if (bar) {
18     let bar = 'barz'
19     console.log('bar is', bar); // foo is barz
20 }
21
22 // note it outputs bar is bar below since
23 // we use `let` inside the `if` statement
24 console.log('bar is', bar); // foo is bar
25
26 // we should use let as much as possible
27 // variables defined by `let` is scoped inside a block {}
28
29 array = [];
30 for (var i = 0; i < 3; i++) {
31
32     array.push(function() { console.log(i); });
33 }
34 array[0](); // 3
35 array[1](); // 3
36 array[2](); // 3
37 // since we are using var in the for loop, the closures capture the same i
38
39 array2 = [];
40 for (let k = 0; k < 3; k++) {
41
42     array2.push(function() { console.log(k); });
43 }
44 array2[0](); // 0
45 array2[1](); // 1
46 array2[2](); // 2
```

12.1.2 array

Listing 2: ./code/hello_world/array.js

```

1  let a = [ 1, 2, 3 ];
2  function sum(arr) {
3      let s = 0;
4      for (let x of arr) {
5          s += x;
6      }
7      return s;
8  }
9  // Sum of the array [1,2,3] is 6
10 console.log('Sum of the array [' + a + '] is ' + sum(a));
11
12 function sum2(arr) {
13     let s = 0;
14     for (let i = 0; i != arr.length; ++i) {
15         s += arr[i];
16     }
17     return s;
18 }
19 console.log(sum2(a)); // 6

```

Note that there are two ways to iterate an array:

- `for(let x of array)`
- `for(let i = 0; i != array.length; ++i) { ... }`

To run the above code, use:

```
node array.js
```

12.1.3 class

Listing 3: ./code/hello_world/class.js

```

1  class Point {
2      constructor(x, y) {
3          this.x = x;
4          this.y = y;
5      }
6
7      distance() { return Math.sqrt(this.x * this.x + this.y * this.y); }
8  }
9
10 let p = new Point(1, 1);
11 console.log(p.distance()); // 1.4142135623730951

```

It defines a `Point` class with two fields `x`, `y`. `Point` has two methods: a constructor and a method `distance()`.

Note that class names are by convention capitalized.

12.1.4 template strings

Listing 4: ./code/hello_world/template_strings.js

```
1 let a = "a";
2 let b = "b";
3 console.log("a is " + a + ", b is " + b); // a is a, b is b
4
5 // using string interpolation
6 console.log(`a is ${a}, b is ${b}`); // a is a, b is b
7
8 // Note that it uses `${}`.
9 // It preserves spaces and line breaks
10
11 // multiline examples
12 let s = `
13 a is ${a}
14 b is ${b}
15 `
16 console.log(s);
```

Note that it is similar to the F-string, `f" ", f' ', f"" " ", f''' '''` in Python.

12.1.5 functions

Listing 5: ./code/hello_world/functions.js

```
1 // note that is supports default arguments
2 function hello(msg = "hi") { console.log(msg); }
3 // the above is a function declaration, it is hoisted to the top
4 // in other words, we can invoke it before seeing it.
5
6 hello(); // hi
7 hello('world'); // world
8
9 // this is a function expression.
10 const hi = function(msg = 'hi') { console.log(msg); };
11 hi(); // hi
12 hi('world'); // world
13
14 const add = function(a, b = 1) { return a + b; };
15 console.log(add(1, 2)); // 3
16 console.log(add(1)); // 2
17
18 // arrow functions
19 // there is no keyword `function` and no `return`.
20 const inc = (a, b = 1) => a + b;
21 console.log(inc(1, 2)); // 3
22 console.log(inc(1)); // 2
23
24 const dec = a => a - 1;
25 console.log(dec(10)); // 9
```

(continues on next page)

(continued from previous page)

```

26
27 // for multiline statement, use {}
28 const sub = a => {
29   let b = a - 1;
30   return b;
31 };
32 console.log(sub(100)); // 99

```

Note that it does not have keyword arguments.

12.2 Numbers in javascript

Listing 6: ./code/numbers.js

```

1 // all numbers are of 64-bit floating point type
2
3 let a = 32; // decimal integer
4 let b = 0x20; // hexadecimal integer
5 let c = 0X20;
6 console.log(a, b, c); // 32 32 32
7
8 let d = 0b00010000; // binary integer
9 let e = 0o20;
10 console.log(d, e); // 16 16
11
12 console.log(a / 5); // 6.4. Note that it is not an integer!
13
14 d = 0b1000_0000; // use _ to separate digit for readability
15 console.log(d); // 128
16
17 console.log(2 ** 3); // 8, ** means power
18
19 // Note that we don't need to import Math
20 //
21 // Round to the nearest integer
22 console.log(Math.round(0.6)); // 1
23 console.log(Math.round(0.5)); // 1
24 console.log(Math.round(0.2)); // 0
25
26 console.log(Math.trunc(3.9)); // 3
27
28 // Round to the nearest integer
29 console.log(Math.round(-0.6)); // -1
30
31 console.log(Math.ceil(-0.6)); // -0
32 console.log(Math.ceil(0.6)); // 1
33 console.log(Math.random()); // uniform distribution in the range 0 <= x < 1.0
34
35 console.log(Math.PI); // 3.141592653589793
36 console.log(Math.E); // 2.718281828459045

```

(continues on next page)

```
37
38 // max accepts arbitrary number of arguments
39 console.log(Math.max(2, 3)); // 3
40 console.log(Math.max(2, 3, 10)); // 10
41 console.log(Math.max(2, 3, 10, 100)); // 100
42
43 console.log(Math.pow(2, 5)); // 32
44
45 // note that sin() is not in degrees
46 console.log(Math.sin(30)); // -0.988
47 console.log(Math.sin(Math.PI / 6)); // 0.4999999
48
49 // natural log. base in 2.718
50 console.log(Math.log(Math.E)); // 1
51
52 console.log(Math.log(100) / Math.LN10); // 2
53 console.log(Math.log10(100)); // 2
54 console.log(Math.log2(1024)); // 10
55
56 // log1p(x) == log(1 + x)
57 console.log(Math.log1p(3), Math.log(1 + 3)); // 1.38629, 1.38629
58
59 console.log(Math.exp(2)); // e^2, 7.38905609893065
60
61 // division by 0 is not an error!
62 console.log(1 / 0); // Infinity
63 console.log(-1 / 0); // -Infinity
64 console.log(0 / 0); // NaN
65 console.log(Infinity / Infinity); // NaN
66 console.log(Number.isNaN(NaN)); // true
67
68 // a string to a number
69 console.log(Number("123") === 123); // true
70
71 console.log(parseInt("20") === 20); // true
72
73 // parseInt supports hexadecimal!
74 // It does not support octal or binary.
75 console.log(parseInt("0x20") === 32); // true
76
77 console.log(parseInt("20", 16) === 32); // true
78
79 // octal
80 console.log(parseInt("20", 8) === 16); // true
```

12.3 Boolean in javascript

Listing 7: ./code/boolean.js

```
1  /*
2  null
3  undefined
4  0
5  -0
6  NaN
7  ""
8
9  The above 6 values are automatically converted to false.
10
11 All other values are converted to true. Even empty array is converted to true
12 */
13
14 console.log(true && null);      // null
15 console.log(true && undefined); // undefined
16 console.log([] && true);        // true
17 console.log([] || false);       // []
```

12.4 Strings in javascript

Listing 8: ./code/strings.js

```
1  let s = "abcdefg";
2  console.log(s.length);    // 7
3  console.log(`s is ${s}`); // s is abcdefg
4
5  // start 1 (inclusive)
6  // end 4 (exclusive)
7  console.log(s.substring(1, 4)); // bcd
8  console.log(s.slice(1, 4));     // bcd
9
10 // last 3 characters
11 console.log(s.slice(-3)); // efg
12
13 // start 3 (inclusive)
14 // to the end of the string
15 console.log(s.slice(3)); // defg
16
17 // split using delimiter 'd'
18 console.log(s.split('d')) // ['abc', 'efg']
19
20 // split using delimiter 'de'
21 console.log(s.split('de')) // ['abc', 'fg']
22
23 s = "1232526314";
24 // note: the delimiter is the whole string '23`
```

(continues on next page)

(continued from previous page)

```
25 console.log(s.split('23')) // ['1', '2526314']
26
27 console.log(s.split('2')) // ['1', '3', '5', '6314']
28
29 s = "abcdefga";
30 console.log(s.indexOf('a')); // 0, only the first occurrence
31 console.log(s.indexOf('a', 2)); // 7, search starting from the 2nd position
32 console.log(s.indexOf('c')); // 2
33 console.log(s.indexOf('c', 1)); // 2
34 console.log(s.indexOf('c', 2)); // 2
35 console.log(s.indexOf('c', 3)); // -1, not found
36 console.log(s.indexOf('x')); // -1
37
38 console.log(s.lastIndexOf('a')); // 7
39 console.log(s.indexOf('bc')); // 1
40
41 console.log(s.startsWith('a')); // true
42 console.log(s.startsWith('ab')); // true
43 console.log(s.startsWith('ac')); // false
44
45 console.log(s.endsWith('a')); // true
46 console.log(s.endsWith('ga')); // true
47 console.log(s.endsWith('da')); // false
48
49 // include a substring
50 console.log(s.includes('de')); // true
51 console.log(s.includes('ded')); // false
52
53 // s is not changed. It return a new string
54 console.log(s.replace('ab', 'AB')); // ABcdefga
55 console.log(s.toUpperCase()); // ABCDEFGA
56 console.log(s.toLowerCase()); // abcdefga
57
58 console.log(s[0]); // a
59 console.log(s[1]); // b
60 console.log(s.charAt(0)); // a
61 console.log(s.charAt(1)); // b
62
63 // pad spaces to the left to make the length 3
64 console.log('a' +
65     'x'.padStart(3)) // "a x"
66
67 console.log('a' +
68     'x'.padStart(3, '-')) // "a--x"
69
70 // pad with '-' to the left to make the total length 5
71 console.log('a' +
72     'xy'.padStart(5, '-')) // "a---xy"
73
74 console.log('a' +
75     ' b '.trim() + 'c'); // abc
76
```

(continues on next page)

(continued from previous page)

```

77 console.log('a' +
78     ' b '.trimStart() + 'c'); // ab c
79
80 console.log('a' +
81     ' b '.trimEnd() + 'c'); // a bc
82
83 console.log('ab'.repeat(3)); // ababab
84
85 console.log(String(123) === "123"); // true, a number to a string
86
87 let k = 20
88 console.log(k.toString() === "20"); // true
89 console.log("0x" + k.toString(16)); // 0x14, hexadecimal
90
91 k = 12.3456;
92 console.log(k.toFixed(0)); // 12
93
94 k = 12.5;
95 console.log(k.toFixed(0)); // 13, note that it performs rounding
96
97 k = 12.3456;
98 console.log(k.toFixed(1)); // 12.3
99 console.log(k.toFixed(2)); // 12.35, note that it performs rounding
100 console.log(k.toFixed(5)); // 12.34560
101
102 console.log(k.toExponential(1)); // 1.2e+1
103 console.log(k.toExponential(2)); // 1.23e+1
104 console.log(k.toExponential(3)); // 1.235e+1
105
106 console.log(k.toPrecision(1)); // 1e+1
107 console.log(k.toPrecision(2)); // 12
108 console.log(k.toPrecision(3)); // 12.3
109 console.log(k.toPrecision(4)); // 12.35
110 console.log(k.toPrecision(5)); // 12.346

```

12.5 array in javascript

Listing 9: ./code/array.js

```

1  let a = [ 1, 2, 3 ];
2  let b = a; // a reference
3  b[0] = 100;
4  console.log(a[0] === 100); // true
5
6  a = [ 1, 2, 3 ];
7  let c = [];
8  for (let i = 0; i < a.length; ++i) {
9      c[i] = a[i]; // note: no need to pre-allocate space for c
10 }
11 console.log(c); // [1, 2, 3]

```

(continues on next page)

(continued from previous page)

```
12
13 c[10] = 20;
14 console.log(c);           // [1, 2, 3, <7 empty items>, 20]
15 console.log(c[4]);        // undefined
16 console.log(c[5]);        // undefined
17 console.log(c.length);    // 11, note is 1 larger than 10
18
19 let s = "";
20 for (let i in c) {
21   s += ` ${i}`;
22 }
23 // note that indexes with undefined are not printed
24 console.log(s); // 0 1 2 10
25
26 s = "";
27 for (let i of c) {
28   s += ` ${i}`;
29 }
30
31 // there are seven undefined below
32 console.log(s); // 1 2 3 undefined undefined .... 20
33
34 a = [ 1, 2, 3 ]
35 c = Array.from(a); // return a copy of the array a
36 c[0] = 100;
37 console.log(a[0] == 1); // true
38
39 function equalArrays(a, b) {
40   if (a === b) {
41     return true;
42   }
43
44   if (a.length !== b.length) {
45     return false;
46   }
47
48   for (let i = 0; i < a.length; i++) {
49     if (a[i] !== b[i]) {
50       return false;
51     }
52   }
53
54   return true;
55 }
56
57 console.log(equalArrays(a, c));           // false
58 console.log(equalArrays(a, a));           // true
59 console.log(equalArrays(a, [ 1, 2, 3 ])); // true
60 c[0] = 1;
61 console.log(equalArrays(a, c)); // true
62
63 a = [ 1, 2, 3 ];
```

(continues on next page)

(continued from previous page)

```

64 a.push(4); // use push to append()
65 a.push(5);
66 console.log(a); // [1, 2, 3, 4, 5]
67 a.push(6, 'ten'); // push two elements
68 console.log(a); // [1, 2, 3, 4, 5, 6, 'ten']

```

12.6 typed array in javascript

Typed array is a view into a buffer.

When creating from an array, the array is copied to a newly created buffer.

When creating from a length, a buffer is allocated, which is initialized with 0s.

When creating from a buffer, the buffer is shared.

See

- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/TypedArray
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Typed_arrays

Listing 10: ./code/typed_array.js

```

1 // https://nodejs.org/api/assert.html
2 const assert = require('node:assert');
3
4 assert.equal(Int8Array.BYTES_PER_ELEMENT, 1);
5 assert.equal(Uint8Array.BYTES_PER_ELEMENT, 1);
6
7 assert.equal(Int16Array.BYTES_PER_ELEMENT, 2);
8 assert.equal(Uint16Array.BYTES_PER_ELEMENT, 2);
9
10 assert.equal(Int32Array.BYTES_PER_ELEMENT, 4);
11 assert.equal(Uint32Array.BYTES_PER_ELEMENT, 4);
12
13 assert.equal(Float32Array.BYTES_PER_ELEMENT, 4);
14 assert.equal(Float64Array.BYTES_PER_ELEMENT, 8);

```

Listing 11: ./code/int8_array.js

```

1 // https://nodejs.org/api/assert.html
2 const assert = require('node:assert');
3
4 // From an array
5 let a = [ 1, 2, 3 ];
6 let b = new Int8Array(a); // copy a into b
7 b[0] = 10
8 assert.equal(a[0], 1);
9 assert.equal(b[0], 10);
10 assert(b.length, 3);
11
12 // From a length. 0 initialized

```

(continues on next page)

(continued from previous page)

```
13 c = new Int8Array(2);
14 assert.equal(c.length, 2);
15 for (let i of c) {
16   assert.equal(i, 0);
17 }
18
19 assert.equal(c.BYTES_PER_ELEMENT, 1);
```

Listing 12: ./code/int32_array.js

```
1 // https://nodejs.org/api/assert.html
2 const assert = require('node:assert');
3
4 // 8 bytes buffer
5 let buffer = new ArrayBuffer(8);
6
7 let a = new Int32Array(buffer);
8 assert.equal(a.length, 2); // 2 ints
9 assert.equal(a.byteLength, 8); // 8 bytes
10 assert.equal(a.byteOffset, 0);
11
12 // a and b share the same buffer
13 let b = new Int8Array(buffer);
14
15 a[0] = 0x12345678;
16 assert.equal(b[0], 0x78); // little endian
17 assert.equal(b[1], 0x56); // little endian
18 assert.equal(b[2], 0x34); // little endian
19 assert.equal(b[3], 0x12); // little endian
20
21 // subarray(start, end) -> No copy
22 let c = b.subarray(1, 3); // view to the same buffer, no copy
23 assert.equal(c.length, 2);
24 assert.equal(c.byteOffset, 1);
25 b[1] = 0x23;
26 b[2] = 0x32;
27 assert.equal(c[0], 0x23);
28 assert.equal(c[1], 0x32);
29
30 // slice(start, end) -> copy
31 let d = b.slice(1, 3); // copy
32 assert.equal(d.length, 2);
33 assert.equal(d[0], 0x23);
34 assert.equal(d[1], 0x32);
35 b[0] = 0x11;
36 b[1] = 0x22;
37 assert.equal(d[0], 0x23);
38 assert.equal(d[1], 0x32);
39
40 // share the same buffer
41 let e = new Int8Array(b.buffer);
42 b[0] = 0x30;
```

(continues on next page)

(continued from previous page)

```
43 assert.equal(e[0], 0x30);
```

12.7 Regular expressions in javascript

Listing 13: ./code/regular_expression.js

```
1  /*
2  /^HTML/    Matches the letters HTML at the start of a string
3
4  /^HTML/i    Matches the letters HTML (ignore cases) at the start of a string
5
6  */
7
8  // /[0-9][0-9]*/    Match a non-zero digit, followed by any \# of digits
9  //
10 // /\bjavascript\b/i    Match "javascript" as a word, case-insensitive
11
12 console.log(/^HTML/.test("HTML ABC")); // true
13 console.log(/^HTML/.test("Html ABC")); // false
14 console.log(/^HTML/i.test("Html ABC")); // true, ignore cases
15
16 console.log(/[0-9][0-9]*/.test("0")); // true
17 console.log(/[0-9][0-9]*/.test("01234")); // true
18
19 console.log(/\bjavascript\b/i.test("hello javascript")); // true
20 console.log(/\bjavascript\b/i.test("hello_javascript")); // false
21
22 let text = "testing: 1, 2, 3"
23 let pattern = /\d+/g // matches all instances of one or more digits
24
25 console.log(pattern.test(text)); // true, a match exists
26
27 // return the first matched position
28 console.log(text.search(pattern)); // 9
29 console.log(text.match(pattern)); // ['1', '2', '3'], array of all matches
30 console.log(text.replace(pattern, '#')); // testing: #, #, #
31
32 console.log(text.split(/\D+/)) // split on non-digit, ["", '1', '2', '3']
```

12.8 node

Go to <https://nodejs.org/en/download/> to download pre-built binaries:

```
wget https://nodejs.org/dist/v16.15.1/node-v16.15.1-linux-x64.tar.xz
tar xvf node-v16.15.1-linux-x64.tar.xz
```

and then add `/path/to/node-v16.15.1-linux-x64/bin/` to `PATH`.

```
node -v
```

to show the version, e.g., `v18.16.0`.

It also contains the executable `npm`. In python we have `requirements.txt` and we can use `pip install -r ./requirements.txt`. Similarly, there is `package.json` in javascript and we can use `npm install` in the folder containing `package.json`.

To create a new project, run `npm init -y`, it will generate `package.json`.

To install a package, use `npm install package-name`. To remove a package, use `npm remove package-name`.

An alternative for `npm` is `yarn`: `npm install -g yarn`.

We can run `yarn` inside the directory containing `package.json`.

If a project contains a file `yarn.lock`, then it is using the tool `yarn`.

12.9 TODOs

1. This page https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Installing_basic_software lists some tools to minify code:
 - WebPack: <https://webpack.js.org/>
 - Grunt: <https://gruntjs.com/>
 - Gulp: <https://gulpjs.com/>
2. Color picker tool: https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Colors/Color_picker_tool
3. Google font: <https://fonts.google.com/> and https://developers.google.com/fonts/docs/getting_started

12.10 npm

All packages can be found at <https://npmjs.org/>.

12.10.1 TODO

Read the following packages:

- <https://github.com/martinandert/date-names>
- <https://github.com/dcousens/ordinal.js>
- <https://github.com/TooTallNate/node-wav>

12.11 module

Nodejs supports both CommonJS and ESmodules. Browsers supports only ESmodules.

12.11.1 CommonJS modules

Export a single function:

```
# mod1.js
const mod1Function = () => console.log('Mod1 is alive!')
module.exports = mod1Function

# main.js
mod1Function = require('./mod1.js')
mod1Function()
```

Export more function:

```
# mod1.js
const mod1Function = () => console.log('Mod1 is alive!')
const mod1Function2 = () => console.log('Mod1 is rolling, baby!')

module.exports = { mod1Function, mod1Function2 }

# main.js
({ mod1Function, mod1Function2 } = require('./mod1.js'))
mod1Function()
mod1Function2()
```

12.11.2 ESmodules

Use `npm init -y` to create `package.json`:

```
{
  "name": "modulestestapp",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "type": "module"
}
```

We need to manually add `"type": "module"`.

Export a single function:

```
// mod1.js
const mod1Function = () => console.log('Mod1 is alive!')
export { mod1Function }

// main.js
import { mod1Function } from './mod1.js'
// import { mod1Function } from 'mod1' // or use this one
mod1Function()
```

Export two functions:

```
// mod1.js
const mod1Function = () => console.log('Mod1 is alive!')
const mod1Function2 = () => console.log('Mod1 is rolling, baby!')

export { mod1Function, mod1Function2 }

// main.js
import { mod1Function, mod1Function2 } from './mod1.js'
// import { mod1Function, mod1Function2 } from 'mod1' // or use this one
mod1Function()
mod1Function2()

// main.js (rename imported functions)
import { mod1Function as funct1, mod1Function2 as funct2 } from './mod1.js'
funct1()
funct2()

// main.js (rename2)
import * as mod1 from './mod1.js'
mod1.mod1Function()
mod1.mod1Function2()
```

Default export:

```
// mod1.js
const mod1Function = () => console.log('Mod1 is alive!')
const mod1Function2 = () => console.log('Mod1 is rolling, baby!')

export default mod1Function
export { mod1Function2 }

// main.js
import mod1Function, { mod1Function2 } from './mod1.js'
// import mod1Function, { mod1Function2 } from 'mod1' // or use this one
mod1Function()
mod1Function2()

// main.js (rename it to lalala for the default export)
import lalala, { mod1Function2 } from './mod1.js'
lalala()
mod1Function2()
```

To use a module in html, use

```
<script src="./main.js" type="module"></script>
```

12.11.3 webpack

To bundle all files of a module into a single file:

```
npm install --save-dev webpack webpack-cli
```

```
/* webpack.config.js */
const path = require('path');

module.exports = {
  entry: './main.js',
  output: {
    path: path.resolve(__dirname, 'dist'),
    filename: 'bundle.js',
  },
};
```

```
// package.json
{
  "name": "testappv2",
  "version": "1.0.0",
  "description": "",
  "main": "main.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "build": "webpack"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "webpack": "^5.72.0",
    "webpack-cli": "^4.9.2"
  }
}
```

After running `npm run build`, it will generate `dist/bundle.js`. We can use

```
<script src="./dist/bundle.js" type="module"></script>
```

See more at <https://www.freecodecamp.org/news/modules-in-javascript/>

13.1 Hello world

Listing 1: hello_world.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Hello World</title>
  </head>
  <body>
    <p>Hello world</p>
  </body>
</html>
```

13.1.1 comments

```
<!-- this is a comment -->
```

13.1.2 images

```
</img>
</img>
</img>
```

13.1.3 ordered lists

```
<p> The following points </p>

<ol>
  <li> First </li>
  <li> Second </li>
</ol>
```

13.1.4 unordered lists

```
<p> The following points </p>

<ul>
  <li> foo </li>
  <li> bar </li>
</ul>
```

13.1.5 links

```
<a href="https://www.google.com">some text</a>
```

13.1.6 div

```
<label for="quota">Number of primes:</label>
<input type="text" id="quota" name="quota" value="1000000" />

<button id="generate">Generate primes</button>
<button id="reload">Reload</button>

<div id="output"></div>

<pre readonly class="event-log"></pre>

// js
const quota = document.querySelector("#quota");
const output = document.querySelector("#output");
document.querySelector("#generate").addEventListener("click", () => {
  const primes = generatePrimes(quota.value);

  // note we use output.textContent
  output.textContent = `Finished generating ${quota.value} primes!`;
});

document.querySelector("#reload").addEventListener("click", () => {
  document.location.reload();
});

const log = document.querySelector(".event-log");
log.textContent = "";
```


13.2 References

- Structuring the web with HTML

<https://developer.mozilla.org/en-US/docs/Learn/HTML>

14.1 Hello world

14.1.1 comment

```
/* this is a comment */
```

```
p { color: red; }
```

Then, in some html file, use:

```
<link href="abc/foo.css" rel="stylesheet">
```

14.1.2 Selector

- tag name or element name: e.g., p selects <p>; h1 selects <h1>.
- ID:, e.g., #my-id selects or <p id="my-id">
- class: e.g., .my-class selects and <p class="my-class">
- attribute: e.g., img[src] selects but not

See https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics#different_types_of_selectors and https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Selectors for more.

Example with multiple rules:

Listing 1: Example with multiple rules

```
p {  
  color: red;  
  width: 500px;  
  border: 1px solid black;  
}
```

Example with multiple selectors:

Listing 2: Example with multiple selectors

```
p, li, h1 { color: red; }
```

14.2 References

- CSS basics

https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics

PYBIND11

15.1 GIL

PROTOCOL BUFFERS

16.1 Installation

16.1.1 C++

See <https://github.com/protocolbuffers/protobuf/blob/main/src/README.md>.

```
make protocol-buffers
cd protocol-buffers
wget https://github.com/protocolbuffers/protobuf/releases/download/v3.20.1/protobuf-all-
↪3.20.1.tar.gz
tar xvf protobuf-all-3.20.1.tar.gz
cd protobuf-all-3.20.1
./configure --prefix=$HOME/software/protobuf-3.20.1
make -j 20
make -j 10 check
make install 2>&1 | tee my-log.txt
cd $HOME/software/protobuf-3.20.1
tree . > tree-log.txt
```

```
$ export PKG_CONFIG_PATH=$HOME/software/protobuf-3.20.1:$PKG_CONFIG_PATH

$ pkg-config --cflags protobuf
-I/root/fangjun/software/protobuf-3.20.1/include

$ pkg-config --libs protobuf
-L/root/fangjun/software/protobuf-3.20.1/lib -lprotobuf

$ pkg-config --cflags --libs protobuf
-I/root/fangjun/software/protobuf-3.20.1/include -L/root/fangjun/software/protobuf-3.20.
↪1/lib -lprotobuf

$ pkg-config --libs-only-L protobuf
-L/root/fangjun/software/protobuf-3.20.1/lib

$ pkg-config --libs-only-l protobuf
-lprotobuf
```

```
$ export PATH=$HOME/software/protobuf-3.20.1/bin:$PATH
$ protoc --version
libprotoc 3.20.1
```

Listing 1: ./code/my-log.txt (Installation logs)

```
1 Making install in .
2 make[1]: Entering directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1'
3 make[2]: Entering directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1'
4 make[2]: Nothing to be done for 'install-exec-am'.
5 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/lib/pkgconfig'
6 /usr/bin/install -c -m 644 protobuf.pc protobuf-lite.pc '/root/fangjun/software/
  ↳ protobuf-3.20.1/lib/pkgconfig'
7 make[2]: Leaving directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1'
8 make[1]: Leaving directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1'
9 Making install in src
10 make[1]: Entering directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1/src'
11 make[2]: Entering directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1/src'
12 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/lib'
13 /bin/bash ../libtool --mode=install /usr/bin/install -c libprotobuf-lite.la
  ↳ libprotobuf.la libprotoc.la '/root/fangjun/software/protobuf-3.20.1/lib'
14 libtool: install: /usr/bin/install -c .libs/libprotobuf-lite.so.31.0.1 /root/fangjun/
  ↳ software/protobuf-3.20.1/lib/libprotobuf-lite.so.31.0.1
15 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f
  ↳ libprotobuf-lite.so.31.0.1 libprotobuf-lite.so.31 || { rm -f libprotobuf-lite.so.31 &&
  ↳ ln -s libprotobuf-lite.so.31.0.1 libprotobuf-lite.so.31; }; })
16 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f
  ↳ libprotobuf-lite.so.31.0.1 libprotobuf-lite.so || { rm -f libprotobuf-lite.so && ln -s
  ↳ libprotobuf-lite.so.31.0.1 libprotobuf-lite.so; }; })
17 libtool: install: /usr/bin/install -c .libs/libprotobuf-lite.lai /root/fangjun/software/
  ↳ protobuf-3.20.1/lib/libprotobuf-lite.la
18 libtool: install: /usr/bin/install -c .libs/libprotobuf.so.31.0.1 /root/fangjun/software/
  ↳ protobuf-3.20.1/lib/libprotobuf.so.31.0.1
19 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f
  ↳ libprotobuf.so.31.0.1 libprotobuf.so.31 || { rm -f libprotobuf.so.31 && ln -s
  ↳ libprotobuf.so.31.0.1 libprotobuf.so.31; }; })
20 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f
  ↳ libprotobuf.so.31.0.1 libprotobuf.so || { rm -f libprotobuf.so && ln -s libprotobuf.so.
  ↳ 31.0.1 libprotobuf.so; }; })
21 libtool: install: /usr/bin/install -c .libs/libprotobuf.lai /root/fangjun/software/
  ↳ protobuf-3.20.1/lib/libprotobuf.la
22 libtool: warning: relinking 'libprotoc.la'
23 libtool: install: (cd /root/fangjun/open-source-2/protocol-buffers/protobuf-3.20.1/src; /
  ↳ bin/bash "/root/fangjun/open-source-2/protocol-buffers/protobuf-3.20.1/libtool" --
  ↳ silent --tag CXX --mode=relink g++ -pthread -DHAVE_PTHREAD=1 -DHAVE_ZLIB=1 -Wall -Wno-
  ↳ sign-compare -O2 -g -std=c++11 -DDEBUG -version-info 31:1:0 -export-dynamic -no-
```

(continues on next page)

(continued from previous page)

```

↳ undefined -Wl,--version-script=./libprotoc.map -o libprotoc.la -rpath /root/fangjun/
↳ software/protobuf-3.20.1/lib google/protobuf/compiler/code_generator.lo google/
↳ protobuf/compiler/command_line_interface.lo google/protobuf/compiler/cpp/cpp_enum.lo
↳ google/protobuf/compiler/cpp/cpp_enum_field.lo google/protobuf/compiler/cpp/cpp_
↳ extension.lo google/protobuf/compiler/cpp/cpp_field.lo google/protobuf/compiler/cpp/
↳ cpp_file.lo google/protobuf/compiler/cpp/cpp_generator.lo google/protobuf/compiler/cpp/
↳ cpp_helpers.lo google/protobuf/compiler/cpp/cpp_map_field.lo google/protobuf/compiler/
↳ cpp/cpp_message.lo google/protobuf/compiler/cpp/cpp_message_field.lo google/protobuf/
↳ compiler/cpp/cpp_padding_optimizer.lo google/protobuf/compiler/cpp/cpp_parse_function_
↳ generator.lo google/protobuf/compiler/cpp/cpp_primitive_field.lo google/protobuf/
↳ compiler/cpp/cpp_service.lo google/protobuf/compiler/cpp/cpp_string_field.lo google/
↳ protobuf/compiler/csharp/csharp_doc_comment.lo google/protobuf/compiler/csharp/csharp_
↳ enum.lo google/protobuf/compiler/csharp/csharp_enum_field.lo google/protobuf/compiler/
↳ csharp/csharp_field_base.lo google/protobuf/compiler/csharp/csharp_generator.lo google/
↳ protobuf/compiler/csharp/csharp_helpers.lo google/protobuf/compiler/csharp/csharp_map_
↳ field.lo google/protobuf/compiler/csharp/csharp_message.lo google/protobuf/compiler/
↳ csharp/csharp_message_field.lo google/protobuf/compiler/csharp/csharp_primitive_field.
↳ lo google/protobuf/compiler/csharp/csharp_reflection_class.lo google/protobuf/compiler/
↳ csharp/csharp_repeated_enum_field.lo google/protobuf/compiler/csharp/csharp_repeated_
↳ message_field.lo google/protobuf/compiler/csharp/csharp_repeated_primitive_field.lo
↳ google/protobuf/compiler/csharp/csharp_source_generator_base.lo google/protobuf/
↳ compiler/csharp/csharp_wrapper_field.lo google/protobuf/compiler/java/java_context.lo
↳ google/protobuf/compiler/java/java_doc_comment.lo google/protobuf/compiler/java/java_
↳ enum.lo google/protobuf/compiler/java/java_enum_field.lo google/protobuf/compiler/java/
↳ java_enum_field_lite.lo google/protobuf/compiler/java/java_enum_lite.lo google/
↳ protobuf/compiler/java/java_extension.lo google/protobuf/compiler/java/java_extension_
↳ lite.lo google/protobuf/compiler/java/java_field.lo google/protobuf/compiler/java/java_
↳ file.lo google/protobuf/compiler/java/java_generator.lo google/protobuf/compiler/java/
↳ java_generator_factory.lo google/protobuf/compiler/java/java_helpers.lo google/
↳ protobuf/compiler/java/java_kotlin_generator.lo google/protobuf/compiler/java/java_map_
↳ field.lo google/protobuf/compiler/java/java_map_field_lite.lo google/protobuf/compiler/
↳ java/java_message.lo google/protobuf/compiler/java/java_message_builder.lo google/
↳ protobuf/compiler/java/java_message_builder_lite.lo google/protobuf/compiler/java/java_
↳ message_field.lo google/protobuf/compiler/java/java_message_field_lite.lo google/
↳ protobuf/compiler/java/java_message_lite.lo google/protobuf/compiler/java/java_name_
↳ resolver.lo google/protobuf/compiler/java/java_primitive_field.lo google/protobuf/
↳ compiler/java/java_primitive_field_lite.lo google/protobuf/compiler/java/java_service.
↳ lo google/protobuf/compiler/java/java_shared_code_generator.lo google/protobuf/
↳ compiler/java/java_string_field.lo google/protobuf/compiler/java/java_string_field_
↳ lite.lo google/protobuf/compiler/js/js_generator.lo google/protobuf/compiler/js/well_
↳ known_types_embed.lo google/protobuf/compiler/objectivec/objectivec_enum.lo google/
↳ protobuf/compiler/objectivec/objectivec_enum_field.lo google/protobuf/compiler/
↳ objectivec/objectivec_extension.lo google/protobuf/compiler/objectivec/objectivec_
↳ field.lo google/protobuf/compiler/objectivec/objectivec_file.lo google/protobuf/
↳ compiler/objectivec/objectivec_generator.lo google/protobuf/compiler/objectivec/
↳ objectivec_helpers.lo google/protobuf/compiler/objectivec/objectivec_map_field.lo
↳ google/protobuf/compiler/objectivec/objectivec_message.lo google/protobuf/compiler/
↳ objectivec/objectivec_message_field.lo google/protobuf/compiler/objectivec/objectivec_
↳ oneof.lo google/protobuf/compiler/objectivec/objectivec_primitive_field.lo google/
↳ protobuf/compiler/php/php_generator.lo google/protobuf/compiler/plugin.lo google/
↳ protobuf/compiler/plugin.pb.lo google/protobuf/compiler/python/python_generator.lo
↳ google/protobuf/compiler/python/python_helpers.lo google/protobuf/compiler/python/

```

(continues on next page)

(continued from previous page)

```

↳python_pyi_generator.lo google/protobuf/compiler/ruby/ruby_generator.lo google/
↳protobuf/compiler/subprocess.lo google/protobuf/compiler/zip_writer.lo -lpthread
↳libprotobuf.la -lz )
24 libtool: install: /usr/bin/install -c .libs/libprotoc.so.31.0.1T /root/fangjun/software/
↳protobuf-3.20.1/lib/libprotoc.so.31.0.1
25 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f libprotoc.
↳so.31.0.1 libprotoc.so.31 || { rm -f libprotoc.so.31 && ln -s libprotoc.so.31.0.1
↳libprotoc.so.31; }; })
26 libtool: install: (cd /root/fangjun/software/protobuf-3.20.1/lib && { ln -s -f libprotoc.
↳so.31.0.1 libprotoc.so || { rm -f libprotoc.so && ln -s libprotoc.so.31.0.1 libprotoc.
↳so; }; })
27 libtool: install: /usr/bin/install -c .libs/libprotoc.lai /root/fangjun/software/
↳protobuf-3.20.1/lib/libprotoc.la
28 libtool: install: /usr/bin/install -c .libs/libprotobuf-lite.a /root/fangjun/software/
↳protobuf-3.20.1/lib/libprotobuf-lite.a
29 libtool: install: chmod 644 /root/fangjun/software/protobuf-3.20.1/lib/libprotobuf-lite.a
30 libtool: install: ranlib /root/fangjun/software/protobuf-3.20.1/lib/libprotobuf-lite.a
31 libtool: install: /usr/bin/install -c .libs/libprotobuf.a /root/fangjun/software/
↳protobuf-3.20.1/lib/libprotobuf.a
32 libtool: install: chmod 644 /root/fangjun/software/protobuf-3.20.1/lib/libprotobuf.a
33 libtool: install: ranlib /root/fangjun/software/protobuf-3.20.1/lib/libprotobuf.a
34 libtool: install: /usr/bin/install -c .libs/libprotoc.a /root/fangjun/software/protobuf-
↳3.20.1/lib/libprotoc.a
35 libtool: install: chmod 644 /root/fangjun/software/protobuf-3.20.1/lib/libprotoc.a
36 libtool: install: ranlib /root/fangjun/software/protobuf-3.20.1/lib/libprotoc.a
37 libtool: finish: PATH="/ceph-fj/fangjun/software/py38/bin:/ceph-fj/fangjun/software/jdk-
↳17.0.3/bin:/ceph-fj/fangjun/software/cmake/bin:/ceph-fj/fangjun/software/texlive2021-
↳20210325/bin/x86_64-linux:/ceph-sh1/fangjun/software/cuda-10.2.89/bin:/ceph-fj/fangjun/
↳software/bin:/ceph-sh1/fangjun/software/bin:/ceph-sh1/fangjun/software/nvim-linux64/
↳bin:/ceph-fj/fangjun/software/py38/bin:/ceph-fj/fangjun/software/cmake/bin:/ceph-fj/
↳fangjun/software/texlive2021-20210325/bin/x86_64-linux:/ceph-sh1/fangjun/software/cuda-
↳10.2.89/bin:/ceph-sh1/fangjun/software/nvim-linux64/bin:/usr/local/sbin:/usr/local/
↳bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/sbin" ldconfig -n /root/fangjun/software/
↳protobuf-3.20.1/lib
38 -----
39 Libraries have been installed in:
40 /root/fangjun/software/protobuf-3.20.1/lib
41
42 If you ever happen to want to link against installed libraries
43 in a given directory, LIBDIR, you must either use libtool, and
44 specify the full pathname of the library, or use the '-LLIBDIR'
45 flag during linking and do at least one of the following:
46 - add LIBDIR to the 'LD_LIBRARY_PATH' environment variable
47   during execution
48 - add LIBDIR to the 'LD_RUN_PATH' environment variable
49   during linking
50 - use the '-Wl,-rpath -Wl,LIBDIR' linker flag
51 - have your system administrator add LIBDIR to '/etc/ld.so.conf'
52
53 See any operating system documentation about shared libraries for
54 more information, such as the ld(1) and ld.so(8) manual pages.
55 -----

```

(continues on next page)

(continued from previous page)

```

56 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/bin'
57 /bin/bash ../libtool --mode=install /usr/bin/install -c protoc '/root/fangjun/
↳ software/protobuf-3.20.1/bin'
58 libtool: install: /usr/bin/install -c .libs/protoc /root/fangjun/software/protobuf-3.20.
↳ 1/bin/protoc
59 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include'
60 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf'
61 /usr/bin/install -c -m 644 google/protobuf/any.proto google/protobuf/api.proto google/
↳ protobuf/descriptor.proto google/protobuf/duration.proto google/protobuf/empty.proto
↳ google/protobuf/field_mask.proto google/protobuf/source_context.proto google/protobuf/
↳ struct.proto google/protobuf/timestamp.proto google/protobuf/type.proto google/
↳ protobuf/wrappers.proto '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf
↳ '
62 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler'
63 /usr/bin/install -c -m 644 google/protobuf/compiler/plugin.proto '/root/fangjun/
↳ software/protobuf-3.20.1/include/google/protobuf/compiler'
64 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include'
65 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf'
66 /usr/bin/install -c -m 644 google/protobuf/any.h google/protobuf/any.pb.h google/
↳ protobuf/api.pb.h google/protobuf/arena.h google/protobuf/arena_impl.h google/protobuf/
↳ arenastring.h google/protobuf/arenaz_sampler.h google/protobuf/descriptor.h google/
↳ protobuf/descriptor.pb.h google/protobuf/descriptor_database.h google/protobuf/
↳ duration.pb.h google/protobuf/dynamic_message.h google/protobuf/empty.pb.h google/
↳ protobuf/explicitly_constructed.h google/protobuf/extension_set.h google/protobuf/
↳ extension_set_inl.h google/protobuf/field_access_listener.h google/protobuf/field_mask.
↳ pb.h google/protobuf/generated_enum_reflection.h google/protobuf/generated_enum_util.h
↳ google/protobuf/generated_message_bases.h google/protobuf/generated_message_reflection.
↳ h google/protobuf/generated_message_tctable_decl.h google/protobuf/generated_message_
↳ tctable_impl.h google/protobuf/generated_message_util.h google/protobuf/has_bits.h
↳ google/protobuf/implicit_weak_message.h google/protobuf/inlined_string_field.h google/
↳ protobuf/map.h google/protobuf/map_entry.h google/protobuf/map_entry_lite.h google/
↳ protobuf/map_field.h google/protobuf/map_field_inl.h google/protobuf/map_field_lite.h
↳ google/protobuf/map_type_handler.h google/protobuf/message.h google/protobuf/message_
↳ lite.h google/protobuf/metadata.h google/protobuf/metadata_lite.h google/protobuf/
↳ parse_context.h '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf'
67 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↳ js'
68 /usr/bin/install -c -m 644 google/protobuf/compiler/js/js_generator.h '/root/fangjun/
↳ software/protobuf-3.20.1/include/google/protobuf/compiler/js'
69 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↳ cpp'
70 /usr/bin/install -c -m 644 google/protobuf/compiler/cpp/cpp_file.h google/protobuf/
↳ compiler/cpp/cpp_generator.h google/protobuf/compiler/cpp/cpp_helpers.h google/
↳ protobuf/compiler/cpp/cpp_names.h '/root/fangjun/software/protobuf-3.20.1/include/
↳ google/protobuf/compiler/cpp'
71 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↳ ruby'
72 /usr/bin/install -c -m 644 google/protobuf/compiler/ruby/ruby_generator.h '/root/
↳ fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/ruby'
73 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↳ python'
74 /usr/bin/install -c -m 644 google/protobuf/compiler/python/python_generator.h google/

```

(continues on next page)

(continued from previous page)

```

75 ↪protobuf/compiler/python/python_pyi_generator.h '/root/fangjun/software/protobuf-3.20.
76 ↪1/include/google/protobuf/compiler/python'
75 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/util'
76 /usr/bin/install -c -m 644 google/protobuf/util/delimited_message_util.h google/
↪protobuf/util/field_comparator.h google/protobuf/util/field_mask_util.h google/
↪protobuf/util/json_util.h google/protobuf/util/message_differencer.h google/protobuf/
↪util/time_util.h google/protobuf/util/type_resolver.h google/protobuf/util/type_
↪resolver_util.h '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/util'
77 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/io'
78 /usr/bin/install -c -m 644 google/protobuf/io/coded_stream.h google/protobuf/io/gzip_
↪stream.h google/protobuf/io/io_win32.h google/protobuf/io/printer.h google/protobuf/io/
↪strtod.h google/protobuf/io/tokenizer.h google/protobuf/io/zero_copy_stream.h google/
↪protobuf/io/zero_copy_stream_impl.h google/protobuf/io/zero_copy_stream_impl_lite.h '/'
79 ↪root/fangjun/software/protobuf-3.20.1/include/google/protobuf/io'
↪/bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↪csharp'
80 /usr/bin/install -c -m 644 google/protobuf/compiler/csharp/csharp_doc_comment.h google/
↪protobuf/compiler/csharp/csharp_generator.h google/protobuf/compiler/csharp/csharp_
↪names.h google/protobuf/compiler/csharp/csharp_options.h '/root/fangjun/software/
↪protobuf-3.20.1/include/google/protobuf/compiler/csharp'
81 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↪php'
82 /usr/bin/install -c -m 644 google/protobuf/compiler/php/php_generator.h '/root/fangjun/
↪software/protobuf-3.20.1/include/google/protobuf/compiler/php'
83 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/stubs'
84 /usr/bin/install -c -m 644 google/protobuf/stubs/bytestream.h google/protobuf/stubs/
↪callback.h google/protobuf/stubs/casts.h google/protobuf/stubs/common.h google/
↪protobuf/stubs/hash.h google/protobuf/stubs/logging.h google/protobuf/stubs/macros.h
↪google/protobuf/stubs/map_util.h google/protobuf/stubs/mutex.h google/protobuf/stubs/
↪once.h google/protobuf/stubs/platform_macros.h google/protobuf/stubs/port.h google/
↪protobuf/stubs/status.h google/protobuf/stubs/stl_util.h google/protobuf/stubs/
↪stringpiece.h google/protobuf/stubs/strutil.h google/protobuf/stubs/template_util.h '/'
85 ↪root/fangjun/software/protobuf-3.20.1/include/google/protobuf/stubs'
↪/bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↪java'
86 /usr/bin/install -c -m 644 google/protobuf/compiler/java/java_generator.h google/
↪protobuf/compiler/java/java_kotlin_generator.h google/protobuf/compiler/java/java_
↪names.h '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/java'
87 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf'
88 /usr/bin/install -c -m 644 google/protobuf/port.h google/protobuf/port_def.inc google/
↪protobuf/port_undef.inc google/protobuf/reflection.h google/protobuf/reflection_ops.h
↪google/protobuf/repeated_field.h google/protobuf/repeated_ptr_field.h google/protobuf/
↪service.h google/protobuf/source_context.pb.h google/protobuf/struct.pb.h google/
↪protobuf/text_format.h google/protobuf/timestamp.pb.h google/protobuf/type.pb.h google/
↪protobuf/unknown_field_set.h google/protobuf/wire_format.h google/protobuf/wire_format_
↪lite.h google/protobuf/wrappers.pb.h '/root/fangjun/software/protobuf-3.20.1/include/
↪google/protobuf'
89 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler/
↪objectivec'
90 /usr/bin/install -c -m 644 google/protobuf/compiler/objectivec/objectivec_generator.h
↪google/protobuf/compiler/objectivec/objectivec_helpers.h '/root/fangjun/software/
↪protobuf-3.20.1/include/google/protobuf/compiler/objectivec'

```

(continues on next page)

(continued from previous page)

```

91 /bin/mkdir -p '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler'
92 /usr/bin/install -c -m 644 google/protobuf/compiler/code_generator.h google/protobuf/
  ↳ compiler/command_line_interface.h google/protobuf/compiler/importer.h google/protobuf/
  ↳ compiler/parser.h google/protobuf/compiler/plugin.h google/protobuf/compiler/plugin.pb.
  ↳ h '/root/fangjun/software/protobuf-3.20.1/include/google/protobuf/compiler'
93 make[2]: Leaving directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1/src'
94 make[1]: Leaving directory '/ceph-fj/fangjun/open-source-2/protocol-buffers/protobuf-3.
  ↳ 20.1/src'

```

Listing 2: ./code/tree-log.txt (Installed files)

```

1 .
2 |-- bin
3 |   |-- protoc
4 |-- include
5 |   |-- google
6 |       |-- protobuf
7 |           |-- any.h
8 |           |-- any.pb.h
9 |           |-- any.proto
10 |          |-- api.pb.h
11 |          |-- api.proto
12 |          |-- arena.h
13 |          |-- arena_impl.h
14 |          |-- arenastring.h
15 |          |-- arenaz_sampler.h
16 |          |-- compiler
17 |              |-- code_generator.h
18 |              |-- command_line_interface.h
19 |              |-- cpp
20 |                  |-- cpp_file.h
21 |                  |-- cpp_generator.h
22 |                  |-- cpp_helpers.h
23 |                  |-- cpp_names.h
24 |              |-- csharp
25 |                  |-- csharp_doc_comment.h
26 |                  |-- csharp_generator.h
27 |                  |-- csharp_names.h
28 |                  |-- csharp_options.h
29 |              |-- importer.h
30 |              |-- java
31 |                  |-- java_generator.h
32 |                  |-- java_kotlin_generator.h
33 |                  |-- java_names.h
34 |              |-- js
35 |                  |-- js_generator.h
36 |              |-- objectivec
37 |                  |-- objectivec_generator.h
38 |                  |-- objectivec_helpers.h
39 |              |-- parser.h
40 |              |-- php

```

(continues on next page)

(continued from previous page)

```
41 | | | `-- php_generator.h
42 | | | |-- plugin.h
43 | | | |-- plugin.pb.h
44 | | | |-- plugin.proto
45 | | | |-- python
46 | | | | |-- python_generator.h
47 | | | | `-- python_pyi_generator.h
48 | | | `-- ruby
49 | | |     |-- ruby_generator.h
50 | |-- descriptor.h
51 | |-- descriptor.pb.h
52 | |-- descriptor.proto
53 | |-- descriptor_database.h
54 | |-- duration.pb.h
55 | |-- duration.proto
56 | |-- dynamic_message.h
57 | |-- empty.pb.h
58 | |-- empty.proto
59 | |-- explicitly_constructed.h
60 | |-- extension_set.h
61 | |-- extension_set_inl.h
62 | |-- field_access_listener.h
63 | |-- field_mask.pb.h
64 | |-- field_mask.proto
65 | |-- generated_enum_reflection.h
66 | |-- generated_enum_util.h
67 | |-- generated_message_bases.h
68 | |-- generated_message_reflection.h
69 | |-- generated_message_tctable_decl.h
70 | |-- generated_message_tctable_impl.h
71 | |-- generated_message_util.h
72 | |-- has_bits.h
73 | |-- implicit_weak_message.h
74 | |-- inlined_string_field.h
75 | |-- io
76 | | |-- coded_stream.h
77 | | |-- gzip_stream.h
78 | | |-- io_win32.h
79 | | |-- printer.h
80 | | |-- strtod.h
81 | | |-- tokenizer.h
82 | | |-- zero_copy_stream.h
83 | | |-- zero_copy_stream_impl.h
84 | | `-- zero_copy_stream_impl_lite.h
85 | |-- map.h
86 | |-- map_entry.h
87 | |-- map_entry_lite.h
88 | |-- map_field.h
89 | |-- map_field_inl.h
90 | |-- map_field_lite.h
91 | |-- map_type_handler.h
92 | |-- message.h
```

(continues on next page)

(continued from previous page)

```

93 |         |-- message_lite.h
94 |         |-- metadata.h
95 |         |-- metadata_lite.h
96 |         |-- parse_context.h
97 |         |-- port.h
98 |         |-- port_def.inc
99 |         |-- port_undef.inc
100 |        |-- reflection.h
101 |        |-- reflection_ops.h
102 |        |-- repeated_field.h
103 |        |-- repeated_ptr_field.h
104 |        |-- service.h
105 |        |-- source_context.pb.h
106 |        |-- source_context.proto
107 |        |-- struct.pb.h
108 |        |-- struct.proto
109 |        |-- stubs
110 |        |   |-- bytestream.h
111 |        |   |-- callback.h
112 |        |   |-- casts.h
113 |        |   |-- common.h
114 |        |   |-- hash.h
115 |        |   |-- logging.h
116 |        |   |-- macros.h
117 |        |   |-- map_util.h
118 |        |   |-- mutex.h
119 |        |   |-- once.h
120 |        |   |-- platform_macros.h
121 |        |   |-- port.h
122 |        |   |-- status.h
123 |        |   |-- stl_util.h
124 |        |   |-- stringpiece.h
125 |        |   |-- strutil.h
126 |        |   `-- template_util.h
127 |        |-- text_format.h
128 |        |-- timestamp.pb.h
129 |        |-- timestamp.proto
130 |        |-- type.pb.h
131 |        |-- type.proto
132 |        |-- unknown_field_set.h
133 |        |-- util
134 |        |   |-- delimited_message_util.h
135 |        |   |-- field_comparator.h
136 |        |   |-- field_mask_util.h
137 |        |   |-- json_util.h
138 |        |   |-- message_differencer.h
139 |        |   |-- time_util.h
140 |        |   |-- type_resolver.h
141 |        |   `-- type_resolver_util.h
142 |        |-- wire_format.h
143 |        |-- wire_format_lite.h
144 |        |-- wrappers.pb.h

```

(continues on next page)

(continued from previous page)

```

145 |         `-- wrappers.proto
146 | -- lib
147 |   |-- libprotobuf-lite.a
148 |   |-- libprotobuf-lite.la
149 |   |-- libprotobuf-lite.so -> libprotobuf-lite.so.31.0.1
150 |   |-- libprotobuf-lite.so.31 -> libprotobuf-lite.so.31.0.1
151 |   |-- libprotobuf-lite.so.31.0.1
152 |   |-- libprotobuf.a
153 |   |-- libprotobuf.la
154 |   |-- libprotobuf.so -> libprotobuf.so.31.0.1
155 |   |-- libprotobuf.so.31 -> libprotobuf.so.31.0.1
156 |   |-- libprotobuf.so.31.0.1
157 |   |-- libprotoc.a
158 |   |-- libprotoc.la
159 |   |-- libprotoc.so -> libprotoc.so.31.0.1
160 |   |-- libprotoc.so.31 -> libprotoc.so.31.0.1
161 |   |-- libprotoc.so.31.0.1
162 |   `-- pkgconfig
163 |       |-- protobuf-lite.pc
164 |       `-- protobuf.pc
165 | -- tree-log.txt
166 |
167 | 18 directories, 146 files

```

16.1.2 Install with cmake

```

make protocol-buffers
cd protocol-buffers
wget https://github.com/protocolbuffers/protobuf/releases/download/v3.20.1/protobuf-all-
↪3.20.1.tar.gz
tar xvf protobuf-all-3.20.1.tar.gz
cd protobuf-all-3.20.1
mkdir my-build
cd my-build
cmake -Dprotobuf_BUILD_SHARED_LIBS=ON -DCMAKE_INSTALL_PREFIX=/ceph-fj/fangjun/software/
↪protobuf-3.20.1-cmake/ ../cmake 2>&1 | tee cmake-configure-shared-lib.log
make -j10 2>&1 | tee make-shared.log
make install 2>&1 | tee make-shared-install.log

cd $HOME/software/protobuf-3.20.1-cmake
tree . > tree-cmake-log.txt

```


16.2 Hello

16.2.1 hello.proto

See

- <https://developers.google.com/protocol-buffers/docs/cpptutorial>
- <https://developers.google.com/protocol-buffers/docs/pythontutorial>

Listing 3: ./code/hello.proto

```
1 syntax = "proto2";
2
3 package tutorial;
4
5 // available types:
6 // bool, int32, float, double, string
7 message Person {
8     optional string name = 1;
9     optional int32 id = 2;
10    optional string email = 3;
11
12    enum PhoneType {
13        MOBILE = 0;
14        HOME = 1;
15        WORK = 2;
16    }
17
18    message PhoneNumber {
19        optional string number = 1;
20        optional PhoneType type = 2 [ default = HOME ];
21    }
22
23    repeated PhoneNumber phones = 4;
24 }
25
26 message AddressBook { repeated Person people = 1; }
```

16.2.2 makefile

Listing 4: ./code/Makefile

```
1 .PHONY: all clean
2
3 all: hello.pb.cc hello.pb.h hello_pb2.py
4
5 hello.pb.cc hello.pb.h: hello.proto
6     protoc -I=./ --cpp_out=./ ./hello.proto
7
8 hello_pb2.py: hello.proto
9     protoc -I=./ --python_out=./ ./hello.proto
```

(continues on next page)

(continued from previous page)

```

10
11 clean:
12     $(RM) hello.pb.cc hello.pb.h hello_pb2.py

```

16.2.3 hello.pb.h

Listing 5: ./code/hello.pb.h

```

1  // Generated by the protocol buffer compiler.  DO NOT EDIT!
2  // source: hello.proto
3
4  #ifndef GOOGLE_PROTOBUF_INCLUDED_hello_2eproto
5  #define GOOGLE_PROTOBUF_INCLUDED_hello_2eproto
6
7  #include <limits>
8  #include <string>
9
10 #include <google/protobuf/port_def.inc>
11 #if PROTOBUF_VERSION < 3020000
12 #error This file was generated by a newer version of protoc which is
13 #error incompatible with your Protocol Buffer headers. Please update
14 #error your headers.
15 #endif
16 #if 3020001 < PROTOBUF_MIN_PROTOC_VERSION
17 #error This file was generated by an older version of protoc which is
18 #error incompatible with your Protocol Buffer headers. Please
19 #error regenerate this file with a newer version of protoc.
20 #endif
21
22 #include <google/protobuf/port_undef.inc>
23 #include <google/protobuf/io/coded_stream.h>
24 #include <google/protobuf/arena.h>
25 #include <google/protobuf/arenastring.h>
26 #include <google/protobuf/generated_message_util.h>
27 #include <google/protobuf/metadata_lite.h>
28 #include <google/protobuf/generated_message_reflection.h>
29 #include <google/protobuf/message.h>
30 #include <google/protobuf/repeated_field.h> // IWYU pragma: export
31 #include <google/protobuf/extension_set.h> // IWYU pragma: export
32 #include <google/protobuf/generated_enum_reflection.h>
33 #include <google/protobuf/unknown_field_set.h>
34 // @@protoc_insertion_point(includes)
35 #include <google/protobuf/port_def.inc>
36 #define PROTOBUF_INTERNAL_EXPORT_hello_2eproto
37 PROTOBUF_NAMESPACE_OPEN
38 namespace internal {
39 class AnyMetadata;
40 } // namespace internal
41 PROTOBUF_NAMESPACE_CLOSE
42

```

(continues on next page)

(continued from previous page)

```

43 // Internal implementation detail -- do not use these members.
44 struct TableStruct_hello_2eproto {
45     static const uint32_t offsets[];
46 };
47 extern const ::PROTOBUF_NAMESPACE_ID::internal::DescriptorTable descriptor_table_hello_
    ↪ 2eproto;
48 namespace tutorial {
49 class AddressBook;
50 struct AddressBookDefaultTypeInternal;
51 extern AddressBookDefaultTypeInternal _AddressBook_default_instance_;
52 class Person;
53 struct PersonDefaultTypeInternal;
54 extern PersonDefaultTypeInternal _Person_default_instance_;
55 class Person_PhoneNumber;
56 struct Person_PhoneNumberDefaultTypeInternal;
57 extern Person_PhoneNumberDefaultTypeInternal _Person_PhoneNumber_default_instance_;
58 } // namespace tutorial
59 PROTOBUF_NAMESPACE_OPEN
60 template<> ::tutorial::AddressBook* Arena::CreateMaybeMessage<::tutorial::AddressBook>
    ↪ (Arena*);
61 template<> ::tutorial::Person* Arena::CreateMaybeMessage<::tutorial::Person>(Arena*);
62 template<> ::tutorial::Person_PhoneNumber* Arena::CreateMaybeMessage<::tutorial::Person_
    ↪ PhoneNumber>(Arena*);
63 PROTOBUF_NAMESPACE_CLOSE
64 namespace tutorial {
65
66 enum Person_PhoneType : int {
67     Person_PhoneType_MOBILE = 0,
68     Person_PhoneType_HOME = 1,
69     Person_PhoneType_WORK = 2
70 };
71 bool Person_PhoneType_IsValid(int value);
72 constexpr Person_PhoneType Person_PhoneType_PhoneType_MIN = Person_PhoneType_MOBILE;
73 constexpr Person_PhoneType Person_PhoneType_PhoneType_MAX = Person_PhoneType_WORK;
74 constexpr int Person_PhoneType_PhoneType_ARRAYSIZE = Person_PhoneType_PhoneType_MAX + 1;
75
76 const ::PROTOBUF_NAMESPACE_ID::EnumDescriptor* Person_PhoneType_descriptor();
77 template<typename T>
78 inline const std::string& Person_PhoneType_Name(T enum_t_value) {
79     static_assert(::std::is_same<T, Person_PhoneType>::value ||
80         ::std::is_integral<T>::value,
81         "Incorrect type passed to function Person_PhoneType_Name.");
82     return ::PROTOBUF_NAMESPACE_ID::internal::NameOfEnum(
83         Person_PhoneType_descriptor(), enum_t_value);
84 }
85 inline bool Person_PhoneType_Parse(
86     ::PROTOBUF_NAMESPACE_ID::ConstStringParam name, Person_PhoneType* value) {
87     return ::PROTOBUF_NAMESPACE_ID::internal::ParseNamedEnum<Person_PhoneType>(
88         Person_PhoneType_descriptor(), name, value);
89 }
90 // =====
91

```

(continues on next page)

(continued from previous page)

```

92 class Person_PhoneNumber final :
93     public ::PROTOBUF_NAMESPACE_ID::Message /* @@protoc_insertion_point(class_
↪ definition:tutorial.Person.PhoneNumber) */ {
94 public:
95     inline Person_PhoneNumber() : Person_PhoneNumber(nullptr) {}
96     ~Person_PhoneNumber() override;
97     explicit PROTOBUF_CONSTEXPR Person_PhoneNumber(::PROTOBUF_NAMESPACE_
↪ ID::internal::ConstantInitialized);
98
99     Person_PhoneNumber(const Person_PhoneNumber& from);
100     Person_PhoneNumber(Person_PhoneNumber&& from) noexcept
101         : Person_PhoneNumber() {
102         *this = ::std::move(from);
103     }
104
105     inline Person_PhoneNumber& operator=(const Person_PhoneNumber& from) {
106         CopyFrom(from);
107         return *this;
108     }
109     inline Person_PhoneNumber& operator=(Person_PhoneNumber&& from) noexcept {
110         if (this == &from) return *this;
111         if (GetOwningArena() == from.GetOwningArena())
112             #ifdef PROTOBUF_FORCE_COPY_IN_MOVE
113                 && GetOwningArena() != nullptr
114             #endif // !PROTOBUF_FORCE_COPY_IN_MOVE
115         ) {
116             InternalSwap(&from);
117         } else {
118             CopyFrom(from);
119         }
120         return *this;
121     }
122
123     inline const ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet& unknown_fields() const {
124         return _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
↪ (::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance);
125     }
126     inline ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet* mutable_unknown_fields() {
127         return _internal_metadata_.mutable_unknown_fields<::PROTOBUF_NAMESPACE_
↪ ID::UnknownFieldSet>();
128     }
129
130     static const ::PROTOBUF_NAMESPACE_ID::Descriptor* descriptor() {
131         return GetDescriptor();
132     }
133     static const ::PROTOBUF_NAMESPACE_ID::Descriptor* GetDescriptor() {
134         return default_instance().GetMetadata().descriptor;
135     }
136     static const ::PROTOBUF_NAMESPACE_ID::Reflection* GetReflection() {
137         return default_instance().GetMetadata().reflection;
138     }
139     static const Person_PhoneNumber& default_instance() {

```

(continues on next page)

(continued from previous page)

```

140     return *internal_default_instance();
141 }
142 static inline const Person_PhoneNumber* internal_default_instance() {
143     return reinterpret_cast<const Person_PhoneNumber*>(
144         &_Person_PhoneNumber_default_instance_);
145 }
146 static constexpr int kIndexInFileMessages =
147     0;
148
149 friend void swap(Person_PhoneNumber& a, Person_PhoneNumber& b) {
150     a.Swap(&b);
151 }
152 inline void Swap(Person_PhoneNumber* other) {
153     if (other == this) return;
154 #ifdef PROTOBUF_FORCE_COPY_IN_SWAP
155     if (GetOwningArena() != nullptr &&
156         GetOwningArena() == other->GetOwningArena()) {
157 #else // PROTOBUF_FORCE_COPY_IN_SWAP
158     if (GetOwningArena() == other->GetOwningArena()) {
159 #endif // !PROTOBUF_FORCE_COPY_IN_SWAP
160         InternalSwap(other);
161     } else {
162         ::PROTOBUF_NAMESPACE_ID::internal::GenericSwap(this, other);
163     }
164 }
165 void UnsafeArenaSwap(Person_PhoneNumber* other) {
166     if (other == this) return;
167     GOOGLE_DCHECK(GetOwningArena() == other->GetOwningArena());
168     InternalSwap(other);
169 }
170
171 // implements Message -----
172
173 Person_PhoneNumber* New(::PROTOBUF_NAMESPACE_ID::Arena* arena = nullptr) const final {
174     return CreateMaybeMessage<Person_PhoneNumber>(arena);
175 }
176 using ::PROTOBUF_NAMESPACE_ID::Message::CopyFrom;
177 void CopyFrom(const Person_PhoneNumber& from);
178 using ::PROTOBUF_NAMESPACE_ID::Message::MergeFrom;
179 void MergeFrom(const Person_PhoneNumber& from);
180 private:
181 static void MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to, const ::PROTOBUF_NAMESPACE_ID::Message& from);
182 public:
183 PROTOBUF_ATTRIBUTE_REINITIALIZES void Clear() final;
184 bool IsInitialized() const final;
185
186 size_t ByteSizeLong() const final;
187 const char* _InternalParse(const char* ptr, ::PROTOBUF_NAMESPACE_ID::internal::ParseContext* ctx) final;
188 uint8_t* _InternalSerialize(
189     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const;

```

(continues on next page)

(continued from previous page)

```

190 ↪final;
191 int GetCachedSize() const final { return _cached_size_.Get(); }
192
193 private:
194 void SharedCtor();
195 void SharedDtor();
196 void SetCachedSize(int size) const final;
197 void InternalSwap(Person_PhoneNumber* other);
198
199 private:
200 friend class ::PROTOBUF_NAMESPACE_ID::internal::AnyMetadata;
201 static ::PROTOBUF_NAMESPACE_ID::StringPiece FullMessageName() {
202     return "tutorial.Person.PhoneNumber";
203 }
204 protected:
205 explicit Person_PhoneNumber(::PROTOBUF_NAMESPACE_ID::Arena* arena,
206                             bool is_message_owned = false);
207
208 public:
209 static const ClassData _class_data_;
210 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData* GetClassData() const final;
211 ::PROTOBUF_NAMESPACE_ID::Metadata GetMetadata() const final;
212
213 // nested types -----
214
215 // accessors -----
216
217 enum : int {
218     kNumberFieldName = 1,
219     kTypeFieldName = 2,
220 };
221 // optional string number = 1;
222 bool has_number() const;
223 private:
224 bool _internal_has_number() const;
225 public:
226 void clear_number();
227 const std::string& number() const;
228 template <typename ArgT0 = const std::string&, typename... ArgT>
229 void set_number(ArgT0&& arg0, ArgT... args);
230 std::string* mutable_number();
231 PROTOBUF_NODISCARD std::string* release_number();
232 void set_allocated_number(std::string* number);
233 private:
234 const std::string& _internal_number() const;
235 inline PROTOBUF_ALWAYS_INLINE void _internal_set_number(const std::string& value);
236 std::string* _internal_mutable_number();
237 public:
238
239 // optional .tutorial.Person.PhoneType type = 2 [default = HOME];
240 bool has_type() const;

```

(continues on next page)

(continued from previous page)

```

241 private:
242 bool _internal_has_type() const;
243 public:
244 void clear_type();
245 ::tutorial::Person_PhoneType type() const;
246 void set_type(::tutorial::Person_PhoneType value);
247 private:
248 ::tutorial::Person_PhoneType _internal_type() const;
249 void _internal_set_type(::tutorial::Person_PhoneType value);
250 public:
251
252 // @@protoc_insertion_point(class_scope:tutorial.Person.PhoneNumber)
253 private:
254 class _Internal;
255
256 template <typename T> friend class ::PROTOBUF_NAMESPACE_ID::Arena::InternalHelper;
257 typedef void InternalArenaConstructable_;
258 typedef void DestructorSkippable_;
259 ::PROTOBUF_NAMESPACE_ID::internal::HasBits<1> _has_bits_;
260 mutable ::PROTOBUF_NAMESPACE_ID::internal::CachedSize _cached_size_;
261 ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr number_;
262 int type_;
263 friend struct ::TableStruct_hello_2eproto;
264 };
265 // -----
266
267 class Person final :
268     public ::PROTOBUF_NAMESPACE_ID::Message /* @@protoc_insertion_point(class_
↳ definition:tutorial.Person) */ {
269 public:
270     inline Person() : Person(nullptr) {}
271     ~Person() override;
272     explicit PROTOBUF_CONSTEXPR Person(::PROTOBUF_NAMESPACE_
↳ ID::internal::ConstantInitialized);
273
274     Person(const Person& from);
275     Person(Person&& from) noexcept
276         : Person() {
277         *this = ::std::move(from);
278     }
279
280     inline Person& operator=(const Person& from) {
281         CopyFrom(from);
282         return *this;
283     }
284     inline Person& operator=(Person&& from) noexcept {
285         if (this == &from) return *this;
286         if (GetOwningArena() == from.GetOwningArena()
287 #ifdef PROTOBUF_FORCE_COPY_IN_MOVE
288             && GetOwningArena() != nullptr
289 #endif // !PROTOBUF_FORCE_COPY_IN_MOVE
290         ) {

```

(continues on next page)

(continued from previous page)

```

291     InternalSwap(&from);
292 } else {
293     CopyFrom(from);
294 }
295 return *this;
296 }
297
298 inline const ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet& unknown_fields() const {
299     return _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
↪ (::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance);
300 }
301 inline ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet* mutable_unknown_fields() {
302     return _internal_metadata_.mutable_unknown_fields<::PROTOBUF_NAMESPACE_
↪ ID::UnknownFieldSet>();
303 }
304
305 static const ::PROTOBUF_NAMESPACE_ID::Descriptor* descriptor() {
306     return GetDescriptor();
307 }
308 static const ::PROTOBUF_NAMESPACE_ID::Descriptor* GetDescriptor() {
309     return default_instance().GetMetadata().descriptor;
310 }
311 static const ::PROTOBUF_NAMESPACE_ID::Reflection* GetReflection() {
312     return default_instance().GetMetadata().reflection;
313 }
314 static const Person& default_instance() {
315     return *internal_default_instance();
316 }
317 static inline const Person* internal_default_instance() {
318     return reinterpret_cast<const Person*>(
319         &_Person_default_instance_);
320 }
321 static constexpr int kIndexInFileMessages =
322     1;
323
324 friend void swap(Person& a, Person& b) {
325     a.Swap(&b);
326 }
327 inline void Swap(Person* other) {
328     if (other == this) return;
329 #ifdef PROTOBUF_FORCE_COPY_IN_SWAP
330     if (GetOwningArena() != nullptr &&
331         GetOwningArena() == other->GetOwningArena()) {
332         #else // PROTOBUF_FORCE_COPY_IN_SWAP
333         if (GetOwningArena() == other->GetOwningArena()) {
334         #endif // !PROTOBUF_FORCE_COPY_IN_SWAP
335         InternalSwap(other);
336     } else {
337         ::PROTOBUF_NAMESPACE_ID::internal::GenericSwap(this, other);
338     }
339 }
340 void UnsafeArenaSwap(Person* other) {

```

(continues on next page)

(continued from previous page)

```

341     if (other == this) return;
342     GOOGLE_DCHECK(GetOwningArena() == other->GetOwningArena());
343     InternalSwap(other);
344 }
345
346 // implements Message -----
347
348 Person* New(::PROTOBUF_NAMESPACE_ID::Arena* arena = nullptr) const final {
349     return CreateMaybeMessage<Person>(arena);
350 }
351 using ::PROTOBUF_NAMESPACE_ID::Message::CopyFrom;
352 void CopyFrom(const Person& from);
353 using ::PROTOBUF_NAMESPACE_ID::Message::MergeFrom;
354 void MergeFrom(const Person& from);
355 private:
356 static void MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to, const ::PROTOBUF_NAMESPACE_
↵ ID::Message& from);
357 public:
358 PROTOBUF_ATTRIBUTE_REINITIALIZES void Clear() final;
359 bool IsInitialized() const final;
360
361 size_t ByteSizeLong() const final;
362 const char* _InternalParse(const char* ptr, ::PROTOBUF_NAMESPACE_
↵ ID::internal::ParseContext* ctx) final;
363 uint8_t* _InternalSerialize(
364     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const,
↵ final;
365 int GetCachedSize() const final { return _cached_size_.Get(); }
366
367 private:
368 void SharedCtor();
369 void SharedDtor();
370 void SetCachedSize(int size) const final;
371 void InternalSwap(Person* other);
372
373 private:
374 friend class ::PROTOBUF_NAMESPACE_ID::internal::AnyMetadata;
375 static ::PROTOBUF_NAMESPACE_ID::StringPiece FullMessageName() {
376     return "tutorial.Person";
377 }
378 protected:
379 explicit Person(::PROTOBUF_NAMESPACE_ID::Arena* arena,
380                 bool is_message_owned = false);
381 public:
382
383 static const ClassData _class_data_;
384 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData* GetClassData() const final;
385
386 ::PROTOBUF_NAMESPACE_ID::Metadata GetMetadata() const final;
387
388 // nested types -----
389

```

(continues on next page)

(continued from previous page)

```

390 typedef Person_PhoneNumber PhoneNumber;
391
392 typedef Person_PhoneType PhoneType;
393 static constexpr PhoneType MOBILE =
394     Person_PhoneType_MOBILE;
395 static constexpr PhoneType HOME =
396     Person_PhoneType_HOME;
397 static constexpr PhoneType WORK =
398     Person_PhoneType_WORK;
399 static inline bool PhoneType_IsValid(int value) {
400     return Person_PhoneType_IsValid(value);
401 }
402 static constexpr PhoneType PhoneType_MIN =
403     Person_PhoneType_PhoneType_MIN;
404 static constexpr PhoneType PhoneType_MAX =
405     Person_PhoneType_PhoneType_MAX;
406 static constexpr int PhoneType_ARRAYSIZE =
407     Person_PhoneType_PhoneType_ARRAYSIZE;
408 static inline const ::PROTOBUF_NAMESPACE_ID::EnumDescriptor*
409 PhoneType_descriptor() {
410     return Person_PhoneType_descriptor();
411 }
412 template<typename T>
413 static inline const std::string& PhoneType_Name(T enum_t_value) {
414     static_assert(::std::is_same<T, PhoneType>::value ||
415         ::std::is_integral<T>::value,
416         "Incorrect type passed to function PhoneType_Name.");
417     return Person_PhoneType_Name(enum_t_value);
418 }
419 static inline bool PhoneType_Parse(::PROTOBUF_NAMESPACE_ID::ConstStringParam name,
420     PhoneType* value) {
421     return Person_PhoneType_Parse(name, value);
422 }
423
424 // accessors -----
425
426 enum : int {
427     kPhonesFieldNumber = 4,
428     kNameFieldNumber = 1,
429     kEmailFieldNumber = 3,
430     kIdFieldNumber = 2,
431 };
432 // repeated .tutorial.Person.PhoneNumber phones = 4;
433 int phones_size() const;
434 private:
435 int _internal_phones_size() const;
436 public:
437 void clear_phones();
438 ::tutorial::Person_PhoneNumber* mutable_phones(int index);
439 ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person_PhoneNumber >*
440     mutable_phones();
441 private:

```

(continues on next page)

(continued from previous page)

```

442  const ::tutorial::Person_PhoneNumber& _internal_phones(int index) const;
443  ::tutorial::Person_PhoneNumber* _internal_add_phones();
444  public:
445  const ::tutorial::Person_PhoneNumber& phones(int index) const;
446  ::tutorial::Person_PhoneNumber* add_phones();
447  const ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person_PhoneNumber >&
448      phones() const;
449
450  // optional string name = 1;
451  bool has_name() const;
452  private:
453  bool _internal_has_name() const;
454  public:
455  void clear_name();
456  const std::string& name() const;
457  template <typename ArgT0 = const std::string&, typename... ArgT>
458  void set_name(ArgT0&& arg0, ArgT... args);
459  std::string* mutable_name();
460  PROTOBUF_NODISCARD std::string* release_name();
461  void set_allocated_name(std::string* name);
462  private:
463  const std::string& _internal_name() const;
464  inline PROTOBUF_ALWAYS_INLINE void _internal_set_name(const std::string& value);
465  std::string* _internal_mutable_name();
466  public:
467
468  // optional string email = 3;
469  bool has_email() const;
470  private:
471  bool _internal_has_email() const;
472  public:
473  void clear_email();
474  const std::string& email() const;
475  template <typename ArgT0 = const std::string&, typename... ArgT>
476  void set_email(ArgT0&& arg0, ArgT... args);
477  std::string* mutable_email();
478  PROTOBUF_NODISCARD std::string* release_email();
479  void set_allocated_email(std::string* email);
480  private:
481  const std::string& _internal_email() const;
482  inline PROTOBUF_ALWAYS_INLINE void _internal_set_email(const std::string& value);
483  std::string* _internal_mutable_email();
484  public:
485
486  // optional int32 id = 2;
487  bool has_id() const;
488  private:
489  bool _internal_has_id() const;
490  public:
491  void clear_id();
492  int32_t id() const;
493  void set_id(int32_t value);

```

(continues on next page)

(continued from previous page)

```

494 private:
495     int32_t _internal_id() const;
496     void _internal_set_id(int32_t value);
497 public:
498
499     // @@protoc_insertion_point(class_scope:tutorial.Person)
500 private:
501     class _Internal;
502
503     template <typename T> friend class ::PROTOBUF_NAMESPACE_ID::Arena::InternalHelper;
504     typedef void InternalArenaConstructable_;
505     typedef void DestructorSkippable_;
506     ::PROTOBUF_NAMESPACE_ID::internal::HasBits<1> _has_bits_;
507     mutable ::PROTOBUF_NAMESPACE_ID::internal::CachedSize _cached_size_;
508     ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person_PhoneNumber > phones_;
509     ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr name_;
510     ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr email_;
511     int32_t id_;
512     friend struct ::TableStruct_hello_2eproto;
513 };
514 // -----
515
516 class AddressBook final :
517     public ::PROTOBUF_NAMESPACE_ID::Message /* @@protoc_insertion_point(class_
↳ definition:tutorial.AddressBook) */ {
518 public:
519     inline AddressBook() : AddressBook(nullptr) {}
520     ~AddressBook() override;
521     explicit PROTOBUF_CONSTEXPR AddressBook(::PROTOBUF_NAMESPACE_
↳ ID::internal::ConstantInitialized);
522
523     AddressBook(const AddressBook& from);
524     AddressBook(AddressBook&& from) noexcept
525         : AddressBook() {
526         *this = ::std::move(from);
527     }
528
529     inline AddressBook& operator=(const AddressBook& from) {
530         CopyFrom(from);
531         return *this;
532     }
533     inline AddressBook& operator=(AddressBook&& from) noexcept {
534         if (this == &from) return *this;
535         if (GetOwningArena() == from.GetOwningArena()
536 #ifdef PROTOBUF_FORCE_COPY_IN_MOVE
537             && GetOwningArena() != nullptr
538 #endif // !PROTOBUF_FORCE_COPY_IN_MOVE
539         ) {
540             InternalSwap(&from);
541         } else {
542             CopyFrom(from);
543         }

```

(continues on next page)

(continued from previous page)

```

544     return *this;
545 }
546
547 inline const ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet& unknown_fields() const {
548     return _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
↪ (::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance);
549 }
550 inline ::PROTOBUF_NAMESPACE_ID::UnknownFieldSet* mutable_unknown_fields() {
551     return _internal_metadata_.mutable_unknown_fields<::PROTOBUF_NAMESPACE_
↪ ID::UnknownFieldSet>();
552 }
553
554 static const ::PROTOBUF_NAMESPACE_ID::Descriptor* descriptor() {
555     return GetDescriptor();
556 }
557 static const ::PROTOBUF_NAMESPACE_ID::Descriptor* GetDescriptor() {
558     return default_instance().GetMetadata().descriptor;
559 }
560 static const ::PROTOBUF_NAMESPACE_ID::Reflection* GetReflection() {
561     return default_instance().GetMetadata().reflection;
562 }
563 static const AddressBook& default_instance() {
564     return *internal_default_instance();
565 }
566 static inline const AddressBook* internal_default_instance() {
567     return reinterpret_cast<const AddressBook*>(
568         &_AddressBook_default_instance_);
569 }
570 static constexpr int kIndexInFileMessages =
571     2;
572
573 friend void swap(AddressBook& a, AddressBook& b) {
574     a.Swap(&b);
575 }
576 inline void Swap(AddressBook* other) {
577     if (other == this) return;
578 #ifdef PROTOBUF_FORCE_COPY_IN_SWAP
579     if (GetOwningArena() != nullptr &&
580         GetOwningArena() == other->GetOwningArena()) {
581 #else // PROTOBUF_FORCE_COPY_IN_SWAP
582     if (GetOwningArena() == other->GetOwningArena()) {
583 #endif // !PROTOBUF_FORCE_COPY_IN_SWAP
584         InternalSwap(other);
585     } else {
586         ::PROTOBUF_NAMESPACE_ID::internal::GenericSwap(this, other);
587     }
588 }
589 void UnsafeArenaSwap(AddressBook* other) {
590     if (other == this) return;
591     GOOGLE_DCHECK(GetOwningArena() == other->GetOwningArena());
592     InternalSwap(other);
593 }

```

(continues on next page)

(continued from previous page)

```

594
595 // implements Message -----
596
597 AddressBook* New(::PROTOBUF_NAMESPACE_ID::Arena* arena = nullptr) const final {
598     return CreateMaybeMessage<AddressBook>(arena);
599 }
600 using ::PROTOBUF_NAMESPACE_ID::Message::CopyFrom;
601 void CopyFrom(const AddressBook& from);
602 using ::PROTOBUF_NAMESPACE_ID::Message::MergeFrom;
603 void MergeFrom(const AddressBook& from);
604 private:
605 static void MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to, const ::PROTOBUF_NAMESPACE_
↪ ID::Message& from);
606 public:
607 PROTOBUF_ATTRIBUTE_REINITIALIZES void Clear() final;
608 bool IsInitialized() const final;
609
610 size_t ByteSizeLong() const final;
611 const char* _InternalParse(const char* ptr, ::PROTOBUF_NAMESPACE_
↪ ID::internal::ParseContext* ctx) final;
612 uint8_t* _InternalSerialize(
613     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const,
↪ final;
614 int GetCachedSize() const final { return _cached_size_.Get(); }
615
616 private:
617 void SharedCtor();
618 void SharedDtor();
619 void SetCachedSize(int size) const final;
620 void InternalSwap(AddressBook* other);
621
622 private:
623 friend class ::PROTOBUF_NAMESPACE_ID::internal::AnyMetadata;
624 static ::PROTOBUF_NAMESPACE_ID::StringPiece FullMessageName() {
625     return "tutorial.AddressBook";
626 }
627 protected:
628 explicit AddressBook(::PROTOBUF_NAMESPACE_ID::Arena* arena,
629                     bool is_message_owned = false);
630 public:
631
632 static const ClassData _class_data_;
633 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData* GetClassData() const final;
634
635 ::PROTOBUF_NAMESPACE_ID::Metadata GetMetadata() const final;
636
637 // nested types -----
638
639 // accessors -----
640
641 enum : int {
642     kPeopleFieldName = 1,

```

(continues on next page)

(continued from previous page)

```

643 };
644 // repeated .tutorial.Person people = 1;
645 int people_size() const;
646 private:
647 int _internal_people_size() const;
648 public:
649 void clear_people();
650 ::tutorial::Person* mutable_people(int index);
651 ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person >*
652     mutable_people();
653 private:
654 const ::tutorial::Person& _internal_people(int index) const;
655 ::tutorial::Person* _internal_add_people();
656 public:
657 const ::tutorial::Person& people(int index) const;
658 ::tutorial::Person* add_people();
659 const ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person >&
660     people() const;
661
662 // @@protoc_insertion_point(class_scope:tutorial.AddressBook)
663 private:
664 class _Internal;
665
666 template <typename T> friend class ::PROTOBUF_NAMESPACE_ID::Arena::InternalHelper;
667 typedef void InternalArenaConstructable_;
668 typedef void DestructorSkippable_;
669 ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person > people_;
670 mutable ::PROTOBUF_NAMESPACE_ID::internal::CachedSize _cached_size_;
671 friend struct ::TableStruct_hello_2eproto;
672 };
673 // =====
674
675 // =====
676
677 #ifndef __GNUC__
678 #pragma GCC diagnostic push
679 #pragma GCC diagnostic ignored "-Wstrict-aliasing"
680 #endif // __GNUC__
681 // Person_PhoneNumber
682
683 // optional string number = 1;
684 inline bool Person_PhoneNumber::_internal_has_number() const {
685     bool value = (_has_bits_[0] & 0x00000001u) != 0;
686     return value;
687 }
688
689 inline bool Person_PhoneNumber::has_number() const {
690     return _internal_has_number();
691 }
692
693 inline void Person_PhoneNumber::clear_number() {
694     number_.ClearToEmpty();
695     _has_bits_[0] &= ~0x00000001u;

```

(continues on next page)

(continued from previous page)

```

695 }
696 inline const std::string& Person_PhoneNumber::number() const {
697     // @@protoc_insertion_point(field_get:tutorial.Person.PhoneNumber.number)
698     return _internal_number();
699 }
700 template <typename ArgT0, typename... ArgT>
701 inline PROTOBUF_ALWAYS_INLINE
702 void Person_PhoneNumber::set_number(ArgT0&& arg0, ArgT... args) {
703     _has_bits_[0] |= 0x000000001u;
704     number_.Set(static_cast<ArgT0 &&>(arg0), args..., GetArenaForAllocation());
705     // @@protoc_insertion_point(field_set:tutorial.Person.PhoneNumber.number)
706 }
707 inline std::string* Person_PhoneNumber::mutable_number() {
708     std::string* _s = _internal_mutable_number();
709     // @@protoc_insertion_point(field_mutable:tutorial.Person.PhoneNumber.number)
710     return _s;
711 }
712 inline const std::string& Person_PhoneNumber::_internal_number() const {
713     return number_.Get();
714 }
715 inline void Person_PhoneNumber::_internal_set_number(const std::string& value) {
716     _has_bits_[0] |= 0x000000001u;
717     number_.Set(value, GetArenaForAllocation());
718 }
719 inline std::string* Person_PhoneNumber::_internal_mutable_number() {
720     _has_bits_[0] |= 0x000000001u;
721     return number_.Mutable(GetArenaForAllocation());
722 }
723 inline std::string* Person_PhoneNumber::release_number() {
724     // @@protoc_insertion_point(field_release:tutorial.Person.PhoneNumber.number)
725     if (!_internal_has_number()) {
726         return nullptr;
727     }
728     _has_bits_[0] &= ~0x000000001u;
729     auto* p = number_.Release();
730 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
731     if (number_.IsDefault()) {
732         number_.Set("", GetArenaForAllocation());
733     }
734 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
735     return p;
736 }
737 inline void Person_PhoneNumber::set_allocated_number(std::string* number) {
738     if (number != nullptr) {
739         _has_bits_[0] |= 0x000000001u;
740     } else {
741         _has_bits_[0] &= ~0x000000001u;
742     }
743     number_.SetAllocated(number, GetArenaForAllocation());
744 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
745     if (number_.IsDefault()) {
746         number_.Set("", GetArenaForAllocation());

```

(continues on next page)

(continued from previous page)

```

747     }
748 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
749     // @@protoc_insertion_point(field_set_allocated:tutorial.Person.PhoneNumber.number)
750 }
751
752 // optional .tutorial.Person.PhoneType type = 2 [default = HOME];
753 inline bool Person_PhoneNumber::_internal_has_type() const {
754     bool value = (_has_bits_[0] & 0x000000002u) != 0;
755     return value;
756 }
757 inline bool Person_PhoneNumber::has_type() const {
758     return _internal_has_type();
759 }
760 inline void Person_PhoneNumber::clear_type() {
761     type_ = 1;
762     _has_bits_[0] &= ~0x000000002u;
763 }
764 inline ::tutorial::Person_PhoneType Person_PhoneNumber::_internal_type() const {
765     return static_cast< ::tutorial::Person_PhoneType >(type_);
766 }
767 inline ::tutorial::Person_PhoneType Person_PhoneNumber::type() const {
768     // @@protoc_insertion_point(field_get:tutorial.Person.PhoneNumber.type)
769     return _internal_type();
770 }
771 inline void Person_PhoneNumber::_internal_set_type(::tutorial::Person_PhoneType value) {
772     assert(::tutorial::Person_PhoneType_IsValid(value));
773     _has_bits_[0] |= 0x000000002u;
774     type_ = value;
775 }
776 inline void Person_PhoneNumber::set_type(::tutorial::Person_PhoneType value) {
777     _internal_set_type(value);
778     // @@protoc_insertion_point(field_set:tutorial.Person.PhoneNumber.type)
779 }
780
781 // -----
782
783 // Person
784
785 // optional string name = 1;
786 inline bool Person::_internal_has_name() const {
787     bool value = (_has_bits_[0] & 0x000000001u) != 0;
788     return value;
789 }
790 inline bool Person::has_name() const {
791     return _internal_has_name();
792 }
793 inline void Person::clear_name() {
794     name_.ClearToEmpty();
795     _has_bits_[0] &= ~0x000000001u;
796 }
797 inline const std::string& Person::name() const {
798     // @@protoc_insertion_point(field_get:tutorial.Person.name)

```

(continues on next page)

(continued from previous page)

```

799     return _internal_name();
800 }
801 template <typename ArgT0, typename... ArgT>
802 inline PROTOBUF_ALWAYS_INLINE
803 void Person::set_name(ArgT0&& arg0, ArgT... args) {
804     _has_bits_[0] |= 0x000000001u;
805     name_.Set(static_cast<ArgT0 &&>(arg0), args..., GetArenaForAllocation());
806     // @@protoc_insertion_point(field_set:tutorial.Person.name)
807 }
808 inline std::string* Person::mutable_name() {
809     std::string* _s = _internal_mutable_name();
810     // @@protoc_insertion_point(field_mutable:tutorial.Person.name)
811     return _s;
812 }
813 inline const std::string& Person::_internal_name() const {
814     return name_.Get();
815 }
816 inline void Person::_internal_set_name(const std::string& value) {
817     _has_bits_[0] |= 0x000000001u;
818     name_.Set(value, GetArenaForAllocation());
819 }
820 inline std::string* Person::_internal_mutable_name() {
821     _has_bits_[0] |= 0x000000001u;
822     return name_.Mutable(GetArenaForAllocation());
823 }
824 inline std::string* Person::release_name() {
825     // @@protoc_insertion_point(field_release:tutorial.Person.name)
826     if (!_internal_has_name()) {
827         return nullptr;
828     }
829     _has_bits_[0] &= ~0x000000001u;
830     auto* p = name_.Release();
831 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
832     if (name_.IsDefault()) {
833         name_.Set("", GetArenaForAllocation());
834     }
835 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
836     return p;
837 }
838 inline void Person::set_allocated_name(std::string* name) {
839     if (name != nullptr) {
840         _has_bits_[0] |= 0x000000001u;
841     } else {
842         _has_bits_[0] &= ~0x000000001u;
843     }
844     name_.SetAllocated(name, GetArenaForAllocation());
845 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
846     if (name_.IsDefault()) {
847         name_.Set("", GetArenaForAllocation());
848     }
849 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
850     // @@protoc_insertion_point(field_set_allocated:tutorial.Person.name)

```

(continues on next page)

(continued from previous page)

```

851 }
852
853 // optional int32 id = 2;
854 inline bool Person::_internal_has_id() const {
855     bool value = (_has_bits_[0] & 0x00000004u) != 0;
856     return value;
857 }
858 inline bool Person::has_id() const {
859     return _internal_has_id();
860 }
861 inline void Person::clear_id() {
862     id_ = 0;
863     _has_bits_[0] &= ~0x00000004u;
864 }
865 inline int32_t Person::_internal_id() const {
866     return id_;
867 }
868 inline int32_t Person::id() const {
869     // @@protoc_insertion_point(field_get:tutorial.Person.id)
870     return _internal_id();
871 }
872 inline void Person::_internal_set_id(int32_t value) {
873     _has_bits_[0] |= 0x00000004u;
874     id_ = value;
875 }
876 inline void Person::set_id(int32_t value) {
877     _internal_set_id(value);
878     // @@protoc_insertion_point(field_set:tutorial.Person.id)
879 }
880
881 // optional string email = 3;
882 inline bool Person::_internal_has_email() const {
883     bool value = (_has_bits_[0] & 0x00000002u) != 0;
884     return value;
885 }
886 inline bool Person::has_email() const {
887     return _internal_has_email();
888 }
889 inline void Person::clear_email() {
890     email_.ClearToEmpty();
891     _has_bits_[0] &= ~0x00000002u;
892 }
893 inline const std::string& Person::email() const {
894     // @@protoc_insertion_point(field_get:tutorial.Person.email)
895     return _internal_email();
896 }
897 template <typename ArgT0, typename... ArgT>
898 inline PROTOBUF_ALWAYS_INLINE
899 void Person::set_email(ArgT0&& arg0, ArgT... args) {
900     _has_bits_[0] |= 0x00000002u;
901     email_.Set(static_cast<ArgT0 &&>(arg0), args..., GetArenaForAllocation());
902     // @@protoc_insertion_point(field_set:tutorial.Person.email)

```

(continues on next page)

(continued from previous page)

```

903 }
904 inline std::string* Person::mutable_email() {
905     std::string* _s = _internal_mutable_email();
906     // @@protoc_insertion_point(field_mutable:tutorial.Person.email)
907     return _s;
908 }
909 inline const std::string& Person::_internal_email() const {
910     return email_.Get();
911 }
912 inline void Person::_internal_set_email(const std::string& value) {
913     _has_bits_[0] |= 0x000000002u;
914     email_.Set(value, GetArenaForAllocation());
915 }
916 inline std::string* Person::_internal_mutable_email() {
917     _has_bits_[0] |= 0x000000002u;
918     return email_.Mutable(GetArenaForAllocation());
919 }
920 inline std::string* Person::release_email() {
921     // @@protoc_insertion_point(field_release:tutorial.Person.email)
922     if (!_internal_has_email()) {
923         return nullptr;
924     }
925     _has_bits_[0] &= ~0x000000002u;
926     auto* p = email_.Release();
927 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
928     if (email_.IsDefault()) {
929         email_.Set("", GetArenaForAllocation());
930     }
931 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
932     return p;
933 }
934 inline void Person::set_allocated_email(std::string* email) {
935     if (email != nullptr) {
936         _has_bits_[0] |= 0x000000002u;
937     } else {
938         _has_bits_[0] &= ~0x000000002u;
939     }
940     email_.SetAllocated(email, GetArenaForAllocation());
941 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
942     if (email_.IsDefault()) {
943         email_.Set("", GetArenaForAllocation());
944     }
945 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
946     // @@protoc_insertion_point(field_set_allocated:tutorial.Person.email)
947 }
948
949 // repeated .tutorial.Person.PhoneNumber phones = 4;
950 inline int Person::_internal_phones_size() const {
951     return phones_.size();
952 }
953 inline int Person::phones_size() const {
954     return _internal_phones_size();

```

(continues on next page)

(continued from previous page)

```

955 }
956 inline void Person::clear_phones() {
957     phones_.Clear();
958 }
959 inline ::tutorial::Person_PhoneNumber* Person::mutable_phones(int index) {
960     // @@protoc_insertion_point(field_mutable:tutorial.Person.phones)
961     return phones_.Mutable(index);
962 }
963 inline ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person_PhoneNumber >*
964 Person::mutable_phones() {
965     // @@protoc_insertion_point(field_mutable_list:tutorial.Person.phones)
966     return &phones_;
967 }
968 inline const ::tutorial::Person_PhoneNumber& Person::_internal_phones(int index) const {
969     return phones_.Get(index);
970 }
971 inline const ::tutorial::Person_PhoneNumber& Person::phones(int index) const {
972     // @@protoc_insertion_point(field_get:tutorial.Person.phones)
973     return _internal_phones(index);
974 }
975 inline ::tutorial::Person_PhoneNumber* Person::_internal_add_phones() {
976     return phones_.Add();
977 }
978 inline ::tutorial::Person_PhoneNumber* Person::add_phones() {
979     ::tutorial::Person_PhoneNumber* _add = _internal_add_phones();
980     // @@protoc_insertion_point(field_add:tutorial.Person.phones)
981     return _add;
982 }
983 inline const ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person_PhoneNumber >&
984 Person::phones() const {
985     // @@protoc_insertion_point(field_list:tutorial.Person.phones)
986     return phones_;
987 }
988
989 // -----
990
991 // AddressBook
992
993 // repeated .tutorial.Person people = 1;
994 inline int AddressBook::_internal_people_size() const {
995     return people_.size();
996 }
997 inline int AddressBook::people_size() const {
998     return _internal_people_size();
999 }
1000 inline void AddressBook::clear_people() {
1001     people_.Clear();
1002 }
1003 inline ::tutorial::Person* AddressBook::mutable_people(int index) {
1004     // @@protoc_insertion_point(field_mutable:tutorial.AddressBook.people)
1005     return people_.Mutable(index);
1006 }

```

(continues on next page)

(continued from previous page)

```

1007 inline ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person >*
1008 AddressBook::mutable_people() {
1009     // @@protoc_insertion_point(field_mutable_list:tutorial.AddressBook.people)
1010     return &people_;
1011 }
1012 inline const ::tutorial::Person& AddressBook::_internal_people(int index) const {
1013     return people_.Get(index);
1014 }
1015 inline const ::tutorial::Person& AddressBook::people(int index) const {
1016     // @@protoc_insertion_point(field_get:tutorial.AddressBook.people)
1017     return _internal_people(index);
1018 }
1019 inline ::tutorial::Person* AddressBook::_internal_add_people() {
1020     return people_.Add();
1021 }
1022 inline ::tutorial::Person* AddressBook::add_people() {
1023     ::tutorial::Person* _add = _internal_add_people();
1024     // @@protoc_insertion_point(field_add:tutorial.AddressBook.people)
1025     return _add;
1026 }
1027 inline const ::PROTOBUF_NAMESPACE_ID::RepeatedPtrField< ::tutorial::Person >&
1028 AddressBook::people() const {
1029     // @@protoc_insertion_point(field_list:tutorial.AddressBook.people)
1030     return people_;
1031 }
1032
1033 #ifdef __GNUC__
1034     #pragma GCC diagnostic pop
1035 #endif // __GNUC__
1036 // -----
1037
1038 // -----
1039
1040 // @@protoc_insertion_point(namespace_scope)
1041
1042 } // namespace tutorial
1043
1044 PROTOBUF_NAMESPACE_OPEN
1045
1046 template <> struct is_proto_enum< ::tutorial::Person_PhoneType> : ::std::true_type {};
1047 template <>
1048 inline const EnumDescriptor* GetEnumDescriptor< ::tutorial::Person_PhoneType>() {
1049     return ::tutorial::Person_PhoneType_descriptor();
1050 }
1051 }
1052
1053 PROTOBUF_NAMESPACE_CLOSE
1054
1055 // @@protoc_insertion_point(global_scope)
1056
1057 #include <google/protobuf/port_undef.inc>
1058 #endif // GOOGLE_PROTOBUF_INCLUDED_GOOGLE_PROTOBUF_INCLUDED_hello_2eproto

```

16.2.4 hello.pb.cc

Listing 6: ./code/hello.pb.cc

```

1  // Generated by the protocol buffer compiler.  DO NOT EDIT!
2  // source: hello.proto
3
4  #include "hello.pb.h"
5
6  #include <algorithm>
7
8  #include <google/protobuf/io/coded_stream.h>
9  #include <google/protobuf/extension_set.h>
10 #include <google/protobuf/wire_format_lite.h>
11 #include <google/protobuf/descriptor.h>
12 #include <google/protobuf/generated_message_reflection.h>
13 #include <google/protobuf/reflection_ops.h>
14 #include <google/protobuf/wire_format.h>
15 // @@protoc_insertion_point(includes)
16 #include <google/protobuf/port_def.inc>
17
18 PROTOBUF_PRAGMA_INIT_SEG
19
20 namespace _pb = ::PROTOBUF_NAMESPACE_ID;
21 namespace _pbi = _pb::internal;
22
23 namespace tutorial {
24 PROTOBUF_CONSTEXPR Person_PhoneNumber::Person_PhoneNumber(
25     ::_pbi::ConstantInitialized)
26     : number_(&::_pbi::fixed_address_empty_string, ::_pbi::ConstantInitialized{})
27     , type_(1)
28 {}
29 struct Person_PhoneNumberDefaultTypeInternal {
30     PROTOBUF_CONSTEXPR Person_PhoneNumberDefaultTypeInternal()
31         : _instance(::_pbi::ConstantInitialized{}) {}
32     ~Person_PhoneNumberDefaultTypeInternal() {}
33     union {
34         Person_PhoneNumber _instance;
35     };
36 };
37 PROTOBUF_ATTRIBUTE_NO_DESTROY PROTOBUF_CONSTINIT PROTOBUF_ATTRIBUTE_INIT_PRIORITY1
38 ↪ Person_PhoneNumberDefaultTypeInternal _Person_PhoneNumber_default_instance_;
39 PROTOBUF_CONSTEXPR Person::Person(
40     ::_pbi::ConstantInitialized)
41     : phones_()
42     , name_(&::_pbi::fixed_address_empty_string, ::_pbi::ConstantInitialized{})
43     , email_(&::_pbi::fixed_address_empty_string, ::_pbi::ConstantInitialized{})
44     , id_(0){}
45 struct PersonDefaultTypeInternal {
46     PROTOBUF_CONSTEXPR PersonDefaultTypeInternal()
47         : _instance(::_pbi::ConstantInitialized{}) {}
48     ~PersonDefaultTypeInternal() {}
49     union {

```

(continues on next page)

(continued from previous page)

```

49     Person _instance;
50 };
51 };
52 PROTOBUF_ATTRIBUTE_NO_DESTROY PROTOBUF_CONSTINIT PROTOBUF_ATTRIBUTE_INIT_PRIORITY1_
↳ PersonDefaultTypeInternal _Person_default_instance_;
53 PROTOBUF_CONSTEXPR AddressBook::AddressBook(
54     ::_pb::ConstantInitialized)
55     : people_(){}
56 struct AddressBookDefaultTypeInternal {
57     PROTOBUF_CONSTEXPR AddressBookDefaultTypeInternal()
58         : _instance(::_pb::ConstantInitialized{}) {}
59     ~AddressBookDefaultTypeInternal() {}
60     union {
61         AddressBook _instance;
62     };
63 };
64 PROTOBUF_ATTRIBUTE_NO_DESTROY PROTOBUF_CONSTINIT PROTOBUF_ATTRIBUTE_INIT_PRIORITY1_
↳ AddressBookDefaultTypeInternal _AddressBook_default_instance_;
65 } // namespace tutorial
66 static ::_pb::Metadata file_level_metadata_hello_2eproto[3];
67 static const ::_pb::EnumDescriptor* file_level_enum_descriptors_hello_2eproto[1];
68 static constexpr ::_pb::ServiceDescriptor const** file_level_service_descriptors_hello_
↳ 2eproto = nullptr;
69
70 const uint32_t TableStruct_hello_2eproto::offsets[] PROTOBUF_SECTION_VARIABLE(protodesc_
↳ cold) = {
71     PROTOBUF_FIELD_OFFSET(::tutorial::Person_PhoneNumber, _has_bits_),
72     PROTOBUF_FIELD_OFFSET(::tutorial::Person_PhoneNumber, _internal_metadata_),
73     ~0u, // no _extensions_
74     ~0u, // no _oneof_case_
75     ~0u, // no _weak_field_map_
76     ~0u, // no _inlined_string_donated_
77     PROTOBUF_FIELD_OFFSET(::tutorial::Person_PhoneNumber, number_),
78     PROTOBUF_FIELD_OFFSET(::tutorial::Person_PhoneNumber, type_),
79     0,
80     1,
81     PROTOBUF_FIELD_OFFSET(::tutorial::Person, _has_bits_),
82     PROTOBUF_FIELD_OFFSET(::tutorial::Person, _internal_metadata_),
83     ~0u, // no _extensions_
84     ~0u, // no _oneof_case_
85     ~0u, // no _weak_field_map_
86     ~0u, // no _inlined_string_donated_
87     PROTOBUF_FIELD_OFFSET(::tutorial::Person, name_),
88     PROTOBUF_FIELD_OFFSET(::tutorial::Person, id_),
89     PROTOBUF_FIELD_OFFSET(::tutorial::Person, email_),
90     PROTOBUF_FIELD_OFFSET(::tutorial::Person, phones_),
91     0,
92     2,
93     1,
94     ~0u,
95     ~0u, // no _has_bits_
96     PROTOBUF_FIELD_OFFSET(::tutorial::AddressBook, _internal_metadata_),

```

(continues on next page)

(continued from previous page)

```

97  ~0u, // no _extensions_
98  ~0u, // no _oneof_case_
99  ~0u, // no _weak_field_map_
100 ~0u, // no _inlined_string_donated_
101  PROTOBUF_FIELD_OFFSET(::tutorial::AddressBook, people_),
102 };
103 static const ::_pbi::MigrationSchema schemas[] PROTOBUF_SECTION_VARIABLE(protodesc_cold) = {
104     ↪= {
105         { 0, 8, -1, sizeof(::tutorial::Person_PhoneNumber)},
106         { 10, 20, -1, sizeof(::tutorial::Person)},
107         { 24, -1, -1, sizeof(::tutorial::AddressBook)},
108     };
109 static const ::_pb::Message* const file_default_instances[] = {
110     &::tutorial::Person_PhoneNumber_default_instance._instance,
111     &::tutorial::Person_default_instance._instance,
112     &::tutorial::AddressBook_default_instance._instance,
113 };
114
115 const char descriptor_table_protodef_hello_2eproto[] PROTOBUF_SECTION_VARIABLE(protodesc_
116     ↪cold) =
117     "\n\013hello.proto\022\010tutorial\"333\001\n\006Person\022\014\n\004na"
118     "me\030\001 \001(\t\022\n\n\002id\030\002 \001(\005\022\r\n\005email\030\003 \001(\t\
119     ↪\022,\n\006p"
120     "hones\030\004 \003(\0132\034.tutorial.Person.PhoneNumbe"
121     "r\032M\n\013PhoneNumber\022\016\n\006number\030\001 \001(\t\022.\n\004type"
122     "\030\002 \001(\0162\032.tutorial.Person.PhoneType:\004HOME"
123     "\"+\n\tPhoneType\022\n\n\006MOBILE\020\000\022\010\n\004HOME\020\001\022\010\n\004W"
124     "ORK\020\002\"/\n\013AddressBook\022 \n\006people\030\001 \003(\0132\020.t"
125     "utorial.Person"
126     ;
127 static ::_pbi::once_flag descriptor_table_hello_2eproto_once;
128 const ::_pbi::DescriptorTable descriptor_table_hello_2eproto = {
129     false, false, 294, descriptor_table_protodef_hello_2eproto,
130     "hello.proto",
131     &descriptor_table_hello_2eproto_once, nullptr, 0, 3,
132     schemas, file_default_instances, TableStruct_hello_2eproto::offsets,
133     file_level_metadata_hello_2eproto, file_level_enum_descriptors_hello_2eproto,
134     file_level_service_descriptors_hello_2eproto,
135 };
136 PROTOBUF_ATTRIBUTE_WEAK const ::_pbi::DescriptorTable* descriptor_table_hello_2eproto_
137     ↪getter() {
138     return &descriptor_table_hello_2eproto;
139 }
140
141 // Force running AddDescriptors() at dynamic initialization time.
142 PROTOBUF_ATTRIBUTE_INIT_PRIORITY2 static ::_pbi::AddDescriptorsRunner dynamic_init_dummy_
143     ↪hello_2eproto(&descriptor_table_hello_2eproto);
144 namespace tutorial {
145 const ::PROTOBUF_NAMESPACE_ID::EnumDescriptor* Person_PhoneType_descriptor() {
146     ::PROTOBUF_NAMESPACE_ID::internal::AssignDescriptors(&descriptor_table_hello_2eproto);
147     return file_level_enum_descriptors_hello_2eproto[0];
148 }

```

(continues on next page)

(continued from previous page)

```

144 }
145 bool Person_PhoneType_IsValid(int value) {
146     switch (value) {
147         case 0:
148         case 1:
149         case 2:
150             return true;
151         default:
152             return false;
153     }
154 }
155
156 #if (__cplusplus < 201703) && (!defined(_MSC_VER) || (_MSC_VER >= 1900 && _MSC_VER <
157 ↪ 1912))
158 constexpr Person_PhoneType Person::MOBILE;
159 constexpr Person_PhoneType Person::HOME;
160 constexpr Person_PhoneType Person::WORK;
161 constexpr Person_PhoneType Person::PhoneType_MIN;
162 constexpr Person_PhoneType Person::PhoneType_MAX;
163 constexpr int Person::PhoneType_ARRAYSIZE;
164 #endif // (__cplusplus < 201703) && (!defined(_MSC_VER) || (_MSC_VER >= 1900 && _MSC_
165 ↪ VER < 1912))
166
167 // =====
168
169 class Person_PhoneNumber::_Internal {
170 public:
171     using HasBits = decltype(std::declval<Person_PhoneNumber>()._has_bits_);
172     static void set_has_number(HasBits* has_bits) {
173         (*has_bits)[0] |= 1u;
174     }
175     static void set_has_type(HasBits* has_bits) {
176         (*has_bits)[0] |= 2u;
177     }
178 };
179
180 Person_PhoneNumber::Person_PhoneNumber(::PROTOBUF_NAMESPACE_ID::Arena* arena,
181     bool is_message_owned)
182     : ::PROTOBUF_NAMESPACE_ID::Message(arena, is_message_owned) {
183     SharedCtor();
184     // @@protoc_insertion_point(arena_constructor:tutorial.Person.PhoneNumber)
185 }
186
187 Person_PhoneNumber::Person_PhoneNumber(const Person_PhoneNumber& from)
188     : ::PROTOBUF_NAMESPACE_ID::Message(),
189     _has_bits_(from._has_bits_) {
190     _internal_metadata_.MergeFrom<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
191 ↪ metadata_);
192     number_.InitDefault();
193     #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
194         number_.Set("", GetArenaForAllocation());
195     #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
196     if (from._internal_has_number()) {

```

(continues on next page)

(continued from previous page)

```

193     number_.Set(from._internal_number(),
194         GetArenaForAllocation());
195 }
196 type_ = from.type_;
197 // @@protoc_insertion_point(copy_constructor:tutorial.Person.PhoneNumber)
198 }
199
200 inline void Person_PhoneNumber::SharedCtor() {
201     number_.InitDefault();
202     #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
203         number_.Set("", GetArenaForAllocation());
204     #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
205     type_ = 1;
206 }
207
208 Person_PhoneNumber::~Person_PhoneNumber() {
209     // @@protoc_insertion_point(destructor:tutorial.Person.PhoneNumber)
210     if (auto *arena = _internal_metadata_.DeleteReturnArena<:PROTOBUF_NAMESPACE_
211 ↪ ID::UnknownFieldSet>()) {
212         (void)arena;
213         return;
214     }
215     SharedDtor();
216 }
217
218 inline void Person_PhoneNumber::SharedDtor() {
219     GOOGLE_DCHECK(GetArenaForAllocation() == nullptr);
220     number_.Destroy();
221 }
222
223 void Person_PhoneNumber::SetCachedSize(int size) const {
224     _cached_size_.Set(size);
225 }
226
227 void Person_PhoneNumber::Clear() {
228     // @@protoc_insertion_point(message_clear_start:tutorial.Person.PhoneNumber)
229     uint32_t cached_has_bits = 0;
230     // Prevent compiler warnings about cached_has_bits being unused
231     (void)cached_has_bits;
232
233     cached_has_bits = _has_bits_[0];
234     if (cached_has_bits & 0x000000003u) {
235         if (cached_has_bits & 0x000000001u) {
236             number_.ClearNonDefaultToEmpty();
237         }
238         type_ = 1;
239     }
240     _has_bits_.Clear();
241     _internal_metadata_.Clear<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>();
242 }
243
244 const char* Person_PhoneNumber::_InternalParse(const char* ptr, ::_pbi::ParseContext*_)

```

(continues on next page)

(continued from previous page)

```

244 ↪ctx) {
245 #define CHK_(x) if (PROTOBUF_PREDICT_FALSE(!(x))) goto failure
246 _Internal::HasBits has_bits{};
247 while (!ctx->Done(&ptr)) {
248     uint32_t tag;
249     ptr = ::_pbi::ReadTag(ptr, &tag);
250     switch (tag >> 3) {
251         // optional string number = 1;
252         case 1:
253             if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 10)) {
254                 auto str = _internal_mutable_number();
255                 ptr = ::_pbi::InlineGreedyStringParser(str, ptr, ctx);
256                 CHK_(ptr);
257                 #ifndef NDEBUG
258                 ::_pbi::VerifyUTF8(str, "tutorial.Person.PhoneNumber.number");
259                 #endif // !NDEBUG
260             } else
261                 goto handle_unusual;
262             continue;
263         // optional .tutorial.Person.PhoneType type = 2 [default = HOME];
264         case 2:
265             if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 16)) {
266                 uint64_t val = ::PROTOBUF_NAMESPACE_ID::internal::ReadVarint64(&ptr);
267                 CHK_(ptr);
268                 if (PROTOBUF_PREDICT_TRUE(::tutorial::Person_PhoneType_IsValid(val))) {
269                     _internal_set_type(static_cast<::tutorial::Person_PhoneType>(val));
270                 } else {
271                     ::PROTOBUF_NAMESPACE_ID::internal::WriteVarint(2, val, mutable_unknown_
272 ↪fields());
273                 }
274             } else
275                 goto handle_unusual;
276             continue;
277         default:
278             goto handle_unusual;
279     } // switch
280 handle_unusual:
281     if ((tag == 0) || ((tag & 7) == 4)) {
282         CHK_(ptr);
283         ctx->SetLastTag(tag);
284         goto message_done;
285     }
286     ptr = UnknownFieldParse(
287         tag,
288         _internal_metadata_.mutable_unknown_fields<::PROTOBUF_NAMESPACE_
289 ↪ID::UnknownFieldSet>(),
290         ptr, ctx);
291     CHK_(ptr != nullptr);
292 } // while
293 message_done:
294     _has_bits_.Or(has_bits);
295     return ptr;

```

(continues on next page)

(continued from previous page)

```

293 failure:
294     ptr = nullptr;
295     goto message_done;
296 #undef CHK_
297 }
298
299 uint8_t* Person_PhoneNumber::_InternalSerialize(
300     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const {
301     // @@protoc_insertion_point(serialize_to_array_start:tutorial.Person.PhoneNumber)
302     uint32_t cached_has_bits = 0;
303     (void) cached_has_bits;
304
305     cached_has_bits = _has_bits_[0];
306     // optional string number = 1;
307     if (cached_has_bits & 0x000000001u) {
308         ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::VerifyUTF8StringNamedField(
309             this->_internal_number().data(), static_cast<int>(this->_internal_number().
310 ↪length()),
311             ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::SERIALIZE,
312             "tutorial.Person.PhoneNumber.number");
313         target = stream->WriteStringMaybeAliased(
314             1, this->_internal_number(), target);
315     }
316
317     // optional .tutorial.Person.PhoneType type = 2 [default = HOME];
318     if (cached_has_bits & 0x000000002u) {
319         target = stream->EnsureSpace(target);
320         target = ::_pbi::WireFormatLite::WriteEnumToArray(
321             2, this->_internal_type(), target);
322     }
323
324     if (PROTOBUF_PREDICT_FALSE(_internal_metadata_.have_unknown_fields())) {
325         target = ::_pbi::WireFormat::InternalSerializeUnknownFieldsToArray(
326             _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
327 ↪(::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance), target, stream);
328     }
329     // @@protoc_insertion_point(serialize_to_array_end:tutorial.Person.PhoneNumber)
330     return target;
331 }
332
333 size_t Person_PhoneNumber::ByteSizeLong() const {
334     // @@protoc_insertion_point(message_byte_size_start:tutorial.Person.PhoneNumber)
335     size_t total_size = 0;
336
337     uint32_t cached_has_bits = 0;
338     // Prevent compiler warnings about cached_has_bits being unused
339     (void) cached_has_bits;
340
341     cached_has_bits = _has_bits_[0];
342     if (cached_has_bits & 0x000000003u) {
343         // optional string number = 1;
344         if (cached_has_bits & 0x000000001u) {

```

(continues on next page)

(continued from previous page)

```

343     total_size += 1 +
344         ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::StringSize(
345         this->_internal_number());
346     }
347
348     // optional .tutorial.Person.PhoneType type = 2 [default = HOME];
349     if (cached_has_bits & 0x000000002u) {
350         total_size += 1 +
351             ::_pbi::WireFormatLite::EnumSize(this->_internal_type());
352     }
353
354 }
355 return MaybeComputeUnknownFieldsSize(total_size, &_cached_size_);
356 }
357
358 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData Person_PhoneNumber::_class_data_ = {
359     ::PROTOBUF_NAMESPACE_ID::Message::CopyWithSizeCheck,
360     Person_PhoneNumber::MergeImpl
361 };
362 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData*Person_PhoneNumber::GetClassData() {
363     ↪const { return &_class_data_; }
364
365 void Person_PhoneNumber::MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to,
366     const ::PROTOBUF_NAMESPACE_ID::Message& from) {
367     static_cast<Person_PhoneNumber*>(to)->MergeFrom(
368         static_cast<const Person_PhoneNumber&>(from));
369 }
370
371 void Person_PhoneNumber::MergeFrom(const Person_PhoneNumber& from) {
372     // @@protoc_insertion_point(class_specific_merge_from_start:tutorial.Person.PhoneNumber)
373     GOOGLE_DCHECK_NE(&from, this);
374     uint32_t cached_has_bits = 0;
375     (void) cached_has_bits;
376
377     cached_has_bits = from._has_bits_[0];
378     if (cached_has_bits & 0x000000003u) {
379         if (cached_has_bits & 0x000000001u) {
380             _internal_set_number(from._internal_number());
381         }
382         if (cached_has_bits & 0x000000002u) {
383             type_ = from.type_;
384         }
385         _has_bits_[0] |= cached_has_bits;
386     }
387     _internal_metadata_.MergeFrom<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
388     ↪metadata_);
389 }
390
391 void Person_PhoneNumber::CopyFrom(const Person_PhoneNumber& from) {
392     // @@protoc_insertion_point(class_specific_copy_from_start:tutorial.Person.PhoneNumber)
393     if (&from == this) return;

```

(continues on next page)

(continued from previous page)

```

393     Clear();
394     MergeFrom(from);
395 }
396
397 bool Person_PhoneNumber::IsInitialized() const {
398     return true;
399 }
400
401 void Person_PhoneNumber::InternalSwap(Person_PhoneNumber* other) {
402     using std::swap;
403     auto* lhs_arena = GetArenaForAllocation();
404     auto* rhs_arena = other->GetArenaForAllocation();
405     _internal_metadata_.InternalSwap(&other->_internal_metadata_);
406     swap(_has_bits_[0], other->_has_bits_[0]);
407     ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr::InternalSwap(
408         &number_, lhs_arena,
409         &other->number_, rhs_arena
410     );
411     swap(type_, other->type_);
412 }
413
414 ::PROTOBUF_NAMESPACE_ID::Metadata Person_PhoneNumber::GetMetadata() const {
415     return ::_pbi::AssignDescriptors(
416         &descriptor_table_hello_2eproto_getter, &descriptor_table_hello_2eproto_once,
417         file_level_metadata_hello_2eproto[0]);
418 }
419
420 // =====
421
422 class Person::_Internal {
423 public:
424     using HasBits = decltype(std::declval<Person>()._has_bits_);
425     static void set_has_name(HasBits* has_bits) {
426         (*has_bits)[0] |= 1u;
427     }
428     static void set_has_id(HasBits* has_bits) {
429         (*has_bits)[0] |= 4u;
430     }
431     static void set_has_email(HasBits* has_bits) {
432         (*has_bits)[0] |= 2u;
433     }
434 };
435
436 Person::Person(::PROTOBUF_NAMESPACE_ID::Arena* arena,
437               bool is_message_owned)
438     : ::PROTOBUF_NAMESPACE_ID::Message(arena, is_message_owned),
439     phones_(arena) {
440     SharedCtor();
441     // @@protoc_insertion_point(arena_constructor:tutorial.Person)
442 }
443
444 Person::Person(const Person& from)
445     : ::PROTOBUF_NAMESPACE_ID::Message(),

```

(continues on next page)

(continued from previous page)

```

445     _has_bits_(from._has_bits_),
446     phones_(from.phones_) {
447     _internal_metadata_.MergeFrom<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
↳ metadata_);
448     name_.InitDefault();
449     #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
450     name_.Set("", GetArenaForAllocation());
451     #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
452     if (from._internal_has_name()) {
453     name_.Set(from._internal_name(),
454     GetArenaForAllocation());
455     }
456     email_.InitDefault();
457     #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
458     email_.Set("", GetArenaForAllocation());
459     #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
460     if (from._internal_has_email()) {
461     email_.Set(from._internal_email(),
462     GetArenaForAllocation());
463     }
464     id_ = from.id_;
465     // @@protoc_insertion_point(copy_constructor:tutorial.Person)
466 }
467
468 inline void Person::SharedCtor() {
469 name_.InitDefault();
470 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
471 name_.Set("", GetArenaForAllocation());
472 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
473 email_.InitDefault();
474 #ifdef PROTOBUF_FORCE_COPY_DEFAULT_STRING
475 email_.Set("", GetArenaForAllocation());
476 #endif // PROTOBUF_FORCE_COPY_DEFAULT_STRING
477 id_ = 0;
478 }
479
480 Person::~Person() {
481     // @@protoc_insertion_point(destructor:tutorial.Person)
482     if (auto *arena = _internal_metadata_.DeleteReturnArena<:PROTOBUF_NAMESPACE_
↳ ID::UnknownFieldSet>()) {
483     (void)arena;
484     return;
485     }
486     SharedDtor();
487 }
488
489 inline void Person::SharedDtor() {
490     GOOGLE_DCHECK(GetArenaForAllocation() == nullptr);
491     name_.Destroy();
492     email_.Destroy();
493 }
494

```

(continues on next page)

(continued from previous page)

```

495 void Person::SetCachedSize(int size) const {
496     _cached_size_.Set(size);
497 }
498
499 void Person::Clear() {
500     // @@protoc_insertion_point(message_clear_start:tutorial.Person)
501     uint32_t cached_has_bits = 0;
502     // Prevent compiler warnings about cached_has_bits being unused
503     (void) cached_has_bits;
504
505     phones_.Clear();
506     cached_has_bits = _has_bits_[0];
507     if (cached_has_bits & 0x000000003u) {
508         if (cached_has_bits & 0x000000001u) {
509             name_.ClearNonDefaultToEmpty();
510         }
511         if (cached_has_bits & 0x000000002u) {
512             email_.ClearNonDefaultToEmpty();
513         }
514     }
515     id_ = 0;
516     _has_bits_.Clear();
517     _internal_metadata_.Clear<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>();
518 }
519
520 const char* Person::_InternalParse(const char* ptr, ::_pbi::ParseContext* ctx) {
521     #define CHK_(x) if (PROTOBUF_PREDICT_FALSE(!(x))) goto failure
522     _Internal::HasBits has_bits{};
523     while (!ctx->Done(&ptr)) {
524         uint32_t tag;
525         ptr = ::_pbi::ReadTag(ptr, &tag);
526         switch (tag >> 3) {
527             // optional string name = 1;
528             case 1:
529                 if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 10)) {
530                     auto str = _internal_mutable_name();
531                     ptr = ::_pbi::InlineGreedyStringParser(str, ptr, ctx);
532                     CHK_(ptr);
533                     #ifndef NDEBUG
534                     ::_pbi::VerifyUTF8(str, "tutorial.Person.name");
535                     #endif // !NDEBUG
536                 } else
537                     goto handle_unusual;
538                 continue;
539             // optional int32 id = 2;
540             case 2:
541                 if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 16)) {
542                     _Internal::set_has_id(&has_bits);
543                     id_ = ::PROTOBUF_NAMESPACE_ID::internal::ReadVarint32(&ptr);
544                     CHK_(ptr);
545                 } else
546                     goto handle_unusual;

```

(continues on next page)

(continued from previous page)

```

547     continue;
548     // optional string email = 3;
549     case 3:
550         if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 26)) {
551             auto str = _internal_mutable_email();
552             ptr = ::_pbi::InlineGreedyStringParser(str, ptr, ctx);
553             CHK_(ptr);
554             #ifndef NDEBUG
555             ::_pbi::VerifyUTF8(str, "tutorial.Person.email");
556             #endif // !NDEBUG
557         } else
558             goto handle_unusual;
559         continue;
560     // repeated .tutorial.Person.PhoneNumber phones = 4;
561     case 4:
562         if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 34)) {
563             ptr -= 1;
564             do {
565                 ptr += 1;
566                 ptr = ctx->ParseMessage(_internal_add_phones(), ptr);
567                 CHK_(ptr);
568                 if (!ctx->DataAvailable(ptr)) break;
569             } while (::PROTOBUF_NAMESPACE_ID::internal::ExpectTag<34>(ptr));
570         } else
571             goto handle_unusual;
572         continue;
573     default:
574         goto handle_unusual;
575 } // switch
576 handle_unusual:
577     if ((tag == 0) || ((tag & 7) == 4)) {
578         CHK_(ptr);
579         ctx->SetLastTag(tag);
580         goto message_done;
581     }
582     ptr = UnknownFieldParse(
583         tag,
584         _internal_metadata_.mutable_unknown_fields<::PROTOBUF_NAMESPACE_
585 ↪ ID::UnknownFieldSet>(),
586         ptr, ctx);
587     CHK_(ptr != nullptr);
588 } // while
589 message_done:
590     _has_bits_.Or(has_bits);
591     return ptr;
592 failure:
593     ptr = nullptr;
594     goto message_done;
595 #undef CHK_
596 }
597 uint8_t* Person::_InternalSerialize(

```

(continues on next page)

(continued from previous page)

```

598     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const {
599     // @@protoc_insertion_point(serialize_to_array_start:tutorial.Person)
600     uint32_t cached_has_bits = 0;
601     (void) cached_has_bits;
602
603     cached_has_bits = _has_bits_[0];
604     // optional string name = 1;
605     if (cached_has_bits & 0x0000000001u) {
606         ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::VerifyUTF8StringNamedField(
607             this->_internal_name().data(), static_cast<int>(this->_internal_name().length()),
608             ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::SERIALIZE,
609             "tutorial.Person.name");
610         target = stream->WriteStringMaybeAliased(
611             1, this->_internal_name(), target);
612     }
613
614     // optional int32 id = 2;
615     if (cached_has_bits & 0x0000000004u) {
616         target = stream->EnsureSpace(target);
617         target = ::_pbi::WireFormatLite::WriteInt32ToArray(2, this->_internal_id(), target);
618     }
619
620     // optional string email = 3;
621     if (cached_has_bits & 0x0000000002u) {
622         ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::VerifyUTF8StringNamedField(
623             this->_internal_email().data(), static_cast<int>(this->_internal_email().length()),
624             ::PROTOBUF_NAMESPACE_ID::internal::WireFormat::SERIALIZE,
625             "tutorial.Person.email");
626         target = stream->WriteStringMaybeAliased(
627             3, this->_internal_email(), target);
628     }
629
630     // repeated .tutorial.Person.PhoneNumber phones = 4;
631     for (unsigned i = 0,
632          n = static_cast<unsigned>(this->_internal_phones_size()); i < n; i++) {
633         const auto& repfield = this->_internal_phones(i);
634         target = ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::
635             InternalWriteMessage(4, repfield, repfield.GetCachedSize(), target, stream);
636     }
637
638     if (PROTOBUF_PREDICT_FALSE(_internal_metadata_.have_unknown_fields())) {
639         target = ::_pbi::WireFormat::InternalSerializeUnknownFieldsToArray(
640             _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
641             (&::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance), target, stream);
642     }
643     // @@protoc_insertion_point(serialize_to_array_end:tutorial.Person)
644     return target;
645 }
646
647 size_t Person::ByteSizeLong() const {
648     // @@protoc_insertion_point(message_byte_size_start:tutorial.Person)
649     size_t total_size = 0;

```

(continues on next page)

(continued from previous page)

```

649
650 uint32_t cached_has_bits = 0;
651 // Prevent compiler warnings about cached_has_bits being unused
652 (void) cached_has_bits;
653
654 // repeated .tutorial.Person.PhoneNumber phones = 4;
655 total_size += 1UL * this->_internal_phones_size();
656 for (const auto& msg : this->phones_) {
657     total_size +=
658         ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::MessageSize(msg);
659 }
660
661 cached_has_bits = _has_bits_[0];
662 if (cached_has_bits & 0x00000007u) {
663     // optional string name = 1;
664     if (cached_has_bits & 0x00000001u) {
665         total_size += 1 +
666             ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::StringSize(
667                 this->_internal_name());
668     }
669
670     // optional string email = 3;
671     if (cached_has_bits & 0x00000002u) {
672         total_size += 1 +
673             ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::StringSize(
674                 this->_internal_email());
675     }
676
677     // optional int32 id = 2;
678     if (cached_has_bits & 0x00000004u) {
679         total_size += ::_pbi::WireFormatLite::Int32SizePlusOne(this->_internal_id());
680     }
681 }
682
683 return MaybeComputeUnknownFieldsSize(total_size, &cached_size_);
684 }
685
686 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData Person::_class_data_ = {
687     ::PROTOBUF_NAMESPACE_ID::Message::CopyWithSizeCheck,
688     Person::MergeImpl
689 };
690 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData*Person::GetClassData() const { return &
691     ↪_class_data_; }
692
693 void Person::MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to,
694     const ::PROTOBUF_NAMESPACE_ID::Message& from) {
695     static_cast<Person*>(to)->MergeFrom(
696         static_cast<const Person&>(from));
697 }
698
699 void Person::MergeFrom(const Person& from) {

```

(continues on next page)

(continued from previous page)

```

700 // @@protoc_insertion_point(class_specific_merge_from_start:tutorial.Person)
701 GOOGLE_DCHECK_NE(&from, this);
702 uint32_t cached_has_bits = 0;
703 (void) cached_has_bits;
704
705 phones_.MergeFrom(from.phones_);
706 cached_has_bits = from._has_bits_[0];
707 if (cached_has_bits & 0x00000007u) {
708     if (cached_has_bits & 0x00000001u) {
709         _internal_set_name(from._internal_name());
710     }
711     if (cached_has_bits & 0x00000002u) {
712         _internal_set_email(from._internal_email());
713     }
714     if (cached_has_bits & 0x00000004u) {
715         id_ = from.id_;
716     }
717     _has_bits_[0] |= cached_has_bits;
718 }
719 _internal_metadata_.MergeFrom<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
↪ metadata_);
720 }
721
722 void Person::CopyFrom(const Person& from) {
723 // @@protoc_insertion_point(class_specific_copy_from_start:tutorial.Person)
724 if (&from == this) return;
725 Clear();
726 MergeFrom(from);
727 }
728
729 bool Person::IsInitialized() const {
730     return true;
731 }
732
733 void Person::InternalSwap(Person* other) {
734     using std::swap;
735     auto* lhs_arena = GetArenaForAllocation();
736     auto* rhs_arena = other->GetArenaForAllocation();
737     _internal_metadata_.InternalSwap(&other->_internal_metadata_);
738     swap(_has_bits_[0], other->_has_bits_[0]);
739     phones_.InternalSwap(&other->phones_);
740     ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr::InternalSwap(
741         &name_, lhs_arena,
742         &other->name_, rhs_arena
743     );
744     ::PROTOBUF_NAMESPACE_ID::internal::ArenaStringPtr::InternalSwap(
745         &email_, lhs_arena,
746         &other->email_, rhs_arena
747     );
748     swap(id_, other->id_);
749 }
750

```

(continues on next page)

(continued from previous page)

```

751 ::PROTOBUF_NAMESPACE_ID::Metadata Person::GetMetadata() const {
752     return ::_pbi::AssignDescriptors(
753         &descriptor_table_hello_2eproto_getter, &descriptor_table_hello_2eproto_once,
754         file_level_metadata_hello_2eproto[1]);
755 }
756
757 // =====
758
759 class AddressBook::_Internal {
760 public:
761 };
762
763 AddressBook::AddressBook(::PROTOBUF_NAMESPACE_ID::Arena* arena,
764                          bool is_message_owned)
765   : ::PROTOBUF_NAMESPACE_ID::Message(arena, is_message_owned),
766     people_(arena) {
767     SharedCtor();
768     // @@protoc_insertion_point(arena_constructor:tutorial.AddressBook)
769 }
770 AddressBook::AddressBook(const AddressBook& from)
771   : ::PROTOBUF_NAMESPACE_ID::Message(),
772     people_(from.people_) {
773     _internal_metadata_.MergeFrom<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
774 ↪ metadata_);
775     // @@protoc_insertion_point(copy_constructor:tutorial.AddressBook)
776 }
777
778 inline void AddressBook::SharedCtor() {
779 }
780
781 AddressBook::~AddressBook() {
782     // @@protoc_insertion_point(destructor:tutorial.AddressBook)
783     if (auto *arena = _internal_metadata_.DeleteReturnArena<::PROTOBUF_NAMESPACE_
784 ↪ ID::UnknownFieldSet>()) {
785         (void)arena;
786         return;
787     }
788     SharedDtor();
789 }
790
791 inline void AddressBook::SharedDtor() {
792     GOOGLE_DCHECK(GetArenaForAllocation() == nullptr);
793 }
794
795 void AddressBook::SetCachedSize(int size) const {
796     _cached_size_.Set(size);
797 }
798
799 void AddressBook::Clear() {
800     // @@protoc_insertion_point(message_clear_start:tutorial.AddressBook)
801     uint32_t cached_has_bits = 0;
802     // Prevent compiler warnings about cached_has_bits being unused

```

(continues on next page)

(continued from previous page)

```

801 (void) cached_has_bits;
802
803 people_.Clear();
804 _internal_metadata_.Clear<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>();
805 }
806
807 const char* AddressBook::_InternalParse(const char* ptr, ::_pbi::ParseContext* ctx) {
808 #define CHK_(x) if (PROTOBUF_PREDICT_FALSE(!(x))) goto failure
809 while (!ctx->Done(&ptr)) {
810     uint32_t tag;
811     ptr = ::_pbi::ReadTag(ptr, &tag);
812     switch (tag >> 3) {
813         // repeated .tutorial.Person people = 1;
814     case 1:
815         if (PROTOBUF_PREDICT_TRUE(static_cast<uint8_t>(tag) == 10)) {
816             ptr -= 1;
817             do {
818                 ptr += 1;
819                 ptr = ctx->ParseMessage(_internal_add_people(), ptr);
820                 CHK_(ptr);
821                 if (!ctx->DataAvailable(ptr)) break;
822             } while (::PROTOBUF_NAMESPACE_ID::internal::ExpectTag<10>(ptr));
823         } else
824             goto handle_unusual;
825         continue;
826     default:
827         goto handle_unusual;
828     } // switch
829 handle_unusual:
830     if ((tag == 0) || ((tag & 7) == 4)) {
831         CHK_(ptr);
832         ctx->SetLastTag(tag);
833         goto message_done;
834     }
835     ptr = UnknownFieldParse(
836         tag,
837         _internal_metadata_.mutable_unknown_fields<:PROTOBUF_NAMESPACE_
838 ↪ ID::UnknownFieldSet>(),
839         ptr, ctx);
840     CHK_(ptr != nullptr);
841 } // while
842 message_done:
843     return ptr;
844 failure:
845     ptr = nullptr;
846     goto message_done;
847 #undef CHK_
848 }
849
850 uint8_t* AddressBook::_InternalSerialize(
851     uint8_t* target, ::PROTOBUF_NAMESPACE_ID::io::EpsCopyOutputStream* stream) const {
852     // @@protoc_insertion_point(serialize_to_array_start:tutorial.AddressBook)

```

(continues on next page)

(continued from previous page)

```

852 uint32_t cached_has_bits = 0;
853 (void) cached_has_bits;
854
855 // repeated .tutorial.Person people = 1;
856 for (unsigned i = 0,
857      n = static_cast<unsigned>(this->_internal_people_size()); i < n; i++) {
858     const auto& repfield = this->_internal_people(i);
859     target = ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::
860         InternalWriteMessage(1, repfield, repfield.GetCachedSize(), target, stream);
861 }
862
863 if (PROTOBUF_PREDICT_FALSE(_internal_metadata_.have_unknown_fields())) {
864     target = ::_pbi::WireFormat::InternalSerializeUnknownFieldsToArray(
865         _internal_metadata_.unknown_fields<::PROTOBUF_NAMESPACE_ID::UnknownFieldSet>
866         ↪ (::PROTOBUF_NAMESPACE_ID::UnknownFieldSet::default_instance), target, stream);
867 }
868 // @@protoc_insertion_point(serialize_to_array_end:tutorial.AddressBook)
869 return target;
870 }
871
872 size_t AddressBook::ByteSizeLong() const {
873     // @@protoc_insertion_point(message_byte_size_start:tutorial.AddressBook)
874     size_t total_size = 0;
875
876     uint32_t cached_has_bits = 0;
877     // Prevent compiler warnings about cached_has_bits being unused
878     (void) cached_has_bits;
879
880     // repeated .tutorial.Person people = 1;
881     total_size += 1UL * this->_internal_people_size();
882     for (const auto& msg : this->people_) {
883         total_size +=
884             ::PROTOBUF_NAMESPACE_ID::internal::WireFormatLite::MessageSize(msg);
885     }
886
887     return MaybeComputeUnknownFieldsSize(total_size, &_cached_size_);
888 }
889
890 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData AddressBook::_class_data_ = {
891     ::PROTOBUF_NAMESPACE_ID::Message::CopyWithSizeCheck,
892     AddressBook::MergeImpl
893 };
894
895 const ::PROTOBUF_NAMESPACE_ID::Message::ClassData*AddressBook::GetClassData() const {
896     ↪return &_class_data_; }
897
898 void AddressBook::MergeImpl(::PROTOBUF_NAMESPACE_ID::Message* to,
899                             const ::PROTOBUF_NAMESPACE_ID::Message& from) {
900     static_cast<AddressBook*>(to)->MergeFrom(
901         static_cast<const AddressBook&>(from));
902 }

```

(continues on next page)

(continued from previous page)

```

902 void AddressBook::MergeFrom(const AddressBook& from) {
903 // @@protoc_insertion_point(class_specific_merge_from_start:tutorial.AddressBook)
904     GOOGLE_DCHECK_NE(&from, this);
905     uint32_t cached_has_bits = 0;
906     (void) cached_has_bits;
907
908     people_.MergeFrom(from.people_);
909     _internal_metadata_.MergeFrom<:PROTOBUF_NAMESPACE_ID::UnknownFieldSet>(from._internal_
910     metadata_);
911 }
912
913 void AddressBook::CopyFrom(const AddressBook& from) {
914 // @@protoc_insertion_point(class_specific_copy_from_start:tutorial.AddressBook)
915     if (&from == this) return;
916     Clear();
917     MergeFrom(from);
918 }
919
920 bool AddressBook::IsInitialized() const {
921     return true;
922 }
923
924 void AddressBook::InternalSwap(AddressBook* other) {
925     using std::swap;
926     _internal_metadata_.InternalSwap(&other->_internal_metadata_);
927     people_.InternalSwap(&other->people_);
928 }
929
930 :PROTOBUF_NAMESPACE_ID::Metadata AddressBook::GetMetadata() const {
931     return ::_pbi::AssignDescriptors(
932         &descriptor_table_hello_2eproto_getter, &descriptor_table_hello_2eproto_once,
933         file_level_metadata_hello_2eproto[2]);
934 }
935
936 // @@protoc_insertion_point(namespace_scope)
937 } // namespace tutorial
938 PROTOBUF_NAMESPACE_OPEN
939 template<> PROTOBUF_NOINLINE ::tutorial::Person_PhoneNumber*
940 Arena::CreateMaybeMessage< ::tutorial::Person_PhoneNumber >(Arena* arena) {
941     return Arena::CreateMessageInternal< ::tutorial::Person_PhoneNumber >(arena);
942 }
943 template<> PROTOBUF_NOINLINE ::tutorial::Person*
944 Arena::CreateMaybeMessage< ::tutorial::Person >(Arena* arena) {
945     return Arena::CreateMessageInternal< ::tutorial::Person >(arena);
946 }
947 template<> PROTOBUF_NOINLINE ::tutorial::AddressBook*
948 Arena::CreateMaybeMessage< ::tutorial::AddressBook >(Arena* arena) {
949     return Arena::CreateMessageInternal< ::tutorial::AddressBook >(arena);
950 }
951 PROTOBUF_NAMESPACE_CLOSE
952 // @@protoc_insertion_point(global_scope)

```

(continues on next page)

(continued from previous page)

953 `#include <google/protobuf/port_undef.inc>`

17.1 Install

See <https://grpc.io/docs/languages/cpp/quickstart/>

```
git clone --recurse-submodules -b v1.46.3 --depth 1 --shallow-submodules https://github.
↳com/grpc/grpc
mkdir build
cd build
cmake -Dgrpc_INSTALL=ON -Dgrpc_BUILD_TESTS=OFF -DCMAKE_INSTALL_PREFIX=/ceph-fj/fangjun/
↳software/grpc-1.46.3 .. 2>&1 | tee cmake-configure-1.log
make -j20 2>&1 | tee make-1.log
make install 2>&1 | tee make-2.log
```


18.1 TODOs

- Striking gold in binutils
<https://lwn.net/Articles/274859/>
- A ToC of the 20 part linker essay
<https://lwn.net/Articles/276782/>

There are other resources for linkers and loaders, see

- Executables linking and loading reading
<http://research.tedneward.com/reading/software/linking-loading/index.html>
- Optimizing real-world applications with GCC Link Time Optimization
<https://pdfs.semanticscholar.org/6adf/872e3533f40a607f39cdeaf264585efde9af.pdf>
by Honza Hubicka, whose scholar page is <https://scholar.google.cz/citations?user=vhXJ0JEAAAAJ&hl=en>

LINKER AND LOADER

19.1 References

- A ToC of the 20 part linker essay
<https://lwn.net/Articles/276782/>, which is written by Ian Lance Taylor
 1. Introduction, personal history, first half of what's-a-linker
 2. What's-a-linker: Dynamic linking, linker data types, linker operation
 3. Address spaces, Object file formats
 4. Shared Libraries
 5. More Shared Libraries -- specifically, linker implementation; ELF Symbols
- <https://www.ucw.cz/~hubicka/>, author of the gold linker
- Rod Evans: Surfing With a Linker Alien <http://www.linker-aliens.org/blogs/rie/>
 1. Hello there
- Michael Walker's Weblog <http://www.linker-aliens.org/blogs/msw/>
 1. Hello World
 2. How to build a Shared Library
 3. Library Bindings - let's be a little bit more precise shall we

Note: It shows the usage of LD_DEBUG, pldd, ldd, pgrep elfdump.

- Solaris Linking Blogs (Combined Index) <http://www.linker-aliens.org/blogs/>
- LD_LIBRARY_PATH - just say no
http://www.linker-aliens.org/blogs/rie/entry/tt_ld_library_path_tt/
- <https://github.com/berkus/odin/blob/master/tools/sjofn/sjofn.c>
An ELF linker. Read its source code!

19.2 Questions

1. How to view PLT?
2. How to view the relocation information? How many types of relocation are there?
3. What PIC code and non-PIC code look like?
4. What is lazy binding and how to use `LD_BIND_NOW`?
5. What is PLT and GOT?

20.1 aishell

20.1.1 AM training

The first one was added on 2019-02-01.

`asr_train.py` is in `espnet/bin/asr_train.py`, which invokes `espnet.asr.pytorch_backend.asr.train`.

The model is from `espnet.nets.pytorch_backend.e2e_asr.E2E`.

The encoder type *vggblstm*, 3 layers, hidden dim, 1024, proj dim 1024, subsampling 1_2_2_1_1.

Command is:

```
asr_train.py \  
  --config conf/train.yaml \  
  --preprocess-conf \  
  --ngpu 1 \  
  --backend pytorch \  
  --outdir exp/xxx \  
  --debugmode 1 \  
  --dict data/lang_char/train_sp_units.txt \  
  --minibatches 0 \  
  --verbose 0 \  
  --resume \  
  --train-json xxx/data.json \  
  --valid-json yyy/data.json
```


CMAKE

21.1 Tutorials

- <https://cmake.org/cmake/help/latest/guide/tutorial/index.html>

21.2 Install

Go to <https://github.com/Kitware/CMake/releases> for download.

```
wget https://github.com/Kitware/CMake/releases/download/v3.10.3/cmake-3.10.3-Linux-x86_
↪64.sh
chmod +x ./cmake-3.10.3-Linux-x86_64.sh
./cmake-3.10.3-Linux-x86_64.sh --help
mkdir /path/to/software/cmake-3.10.3
./cmake-3.10.3-Linux-x86_64.sh --prefix=/path/to/software/cmake-3.10.3 --skip-license
export PATH=/path/to/software/cmake-3.10.3/bin:$PATH
```


HUGGINGFACE

22.1 spaces

22.1.1 Install client API

```
pip install huggingface_hub
```

```
(py38) kuangfangjun:t$ python3
Python 3.8.0 (default, Oct 28 2019, 16:14:01)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from huggingface_hub import hf_hub_download
>>> hf_hub_download(repo_id="google/pegasus-xsum", filename="config.json")
Downloading: 100%|-----| 1.39k/1.39k [00:00<00:00, 1.12MB/s]
'/root/fangjun/.cache/huggingface/hub/models--google--pegasus-xsum/snapshots/
a0aa5531c00f59a32a167b75130805098b046f9c/config.json'
>>>
```

22.1.2 gradio

- <https://huggingface.co/spaces/alphacep/asr>
- <https://huggingface.co/spaces/jonatasgrosman/asr>
- <https://github.com/gradio-app/gradio/issues/1359>
- <https://huggingface.co/spaces/Gradio-Blocks/neon-tts-plugin-coqui/blob/main/app.py>, css styles

EECS E6870 SPEECH RECOGNITION

23.1 Notes

- <https://www.ee.columbia.edu/~stanchen/spring16/e6870/outline.html>
 - username: speech
 - password: pythonrules

24.1 Hello

```
git clone https://github.com/tencent/ncnn
cd ncnn
git checkout 7b4e77671a4457a414b60cee5425758212e725cf
mkdir build
cd build
cmake -DCMAKE_PREFIX_PATH=/ceph-fj/fangjun/software/protobuf-3.20.1-cmake ..
```

We have to make the following changes:

Listing 1: ./code/hello/7b4e77.diff

```
1 diff --git a/tools/CMakeLists.txt b/tools/CMakeLists.txt
2 index 0b710050..e1a5b3d0 100644
3 --- a/tools/CMakeLists.txt
4 +++ b/tools/CMakeLists.txt
5 @@ -8,6 +8,7 @@ include_directories(${CMAKE_CURRENT_BINARY_DIR})
6   protobuf_generate_cpp(CAFFE_PROTO_SRCS CAFFE_PROTO_HDRS caffe.proto)
7
8   add_executable(caffe2ncnn caffe2ncnn.cpp ${CAFFE_PROTO_SRCS} ${CAFFE_PROTO_HDRS})
9 +include_directories(${Protobuf_INCLUDE_DIR})
10
11   target_link_libraries(caffe2ncnn ${PROTOBUF_LIBRARIES})
12
13 diff --git a/tools/caffe2ncnn.cpp b/tools/caffe2ncnn.cpp
14 index 0eff756a..229cc653 100644
15 --- a/tools/caffe2ncnn.cpp
16 +++ b/tools/caffe2ncnn.cpp
17 @@ -193,7 +193,7 @@ static bool read_proto_from_binary(const char* filepath,
18   ↪google::protobuf::Message
19     google::protobuf::io::InputStream input(&fs);
20     google::protobuf::io::CodedInputStream codedstr(&input);
21
22 -    codedstr.SetTotalBytesLimit(INT_MAX, INT_MAX / 2);
23 +    codedstr.SetTotalBytesLimit(INT_MAX);
24
25     bool success = message->ParseFromCodedStream(&codedstr);
```

To install the Python package:

```

cd ncnn
mkdir build
cd build
cmake ..
make -j
cd ..
pip install .

```

24.1.1 test-net

Listing 2: ./code/hello/test-net.cc

```

#include "datareader.h"
#include "layer/relu.h"
#include "layer_type.h"
#include "net.h"
#include <fstream>
#include <iostream>

#ifdef MY_CHECK
#define MY_CHECK(x, y) \
    do { \
        if (x != y) { \
            NCNN_LOGE(#x " != " #y); \
            exit(EXIT_FAILURE); \
        } \
    } while (0)
#endif

class DataReaderFromEmpty : public ncnn::DataReader {
public:
    virtual int scan(const char *format, void *p) const { return 0; }
    virtual size_t read(void *buf, size_t size) const {
        memset(buf, 0, size);
        return size;
    }
};

ncnn::Layer *MyLayerCreator(void * /*userdata*/) { return new ncnn::ReLU(); }
void MyLayerDestoryer(ncnn::Layer *layer, void * /*userdata*/) { delete layer; }

static void TestCustomLayer() {
    static const char *s = R"(
7767517
2 2
Input      data      0 1 data
MyLayer    my_layer  1 1 data out
)";
    std::ofstream of("filename.param");
    of << s;
    of.close();
}

```

(continues on next page)

(continued from previous page)

```

ncnn::Net net;
net.register_custom_layer("MyLayer", MyLayerCreator, MyLayerDestoryer,
                           nullptr);
// This is the first custom layer, so its index is 0
MY_CHECK(net.custom_layer_to_index("MyLayer"), 0);

DataReaderFromEmpty dr;

// return 0 on success
// What does load_param() do?
// If we need to set options, we have to set it before calling load_param()
// (1) Read the magic number
// (2) Allocate space for layers and blobs
auto ret = net.load_param("filename.param");
MY_CHECK(ret, 0);

// load_model() is optional if there are no parameters to read
// ret = net.load_model(dr);
// MY_CHECK(ret, 0);

ncnn::Mat m(2, 3);
float *p = m;
for (int32_t i = 0; i != m.total(); ++i) {
    p[i] = i - float(m.total()) / 2;
}

for (int32_t i = 0; i != m.total(); ++i) {
    std::cout << p[i] << ", ";
}
std::cout << "\n";

ncnn::Extractor ex = net.create_extractor();
ex.input("data", m);
ncnn::Mat out;
ex.extract("out", out);
p = out;
for (int32_t i = 0; i != out.total(); ++i) {
    std::cout << p[i] << ", ";
}
std::cout << "\n";
}

/*
ncnn::Net uses pimpl

std::vector<Layer*> layers;
std::vector<Blob> blobs;
*/
void TestNet() {
    TestCustomLayer();
    NCNN_LOGE("Test net\n");
}

```

(continues on next page)

(continued from previous page)

```

/*
77675157 -> magic number
3 3 -> layer_count, blob_count

Input          data          0 1 data
->
layer type: Input
Layer name: data
bottom count: 0
top count: 1
It uses ncnn::create_layer(layer_type) to create a layer

If create_layer() returns a nullptr, it invokes
create_custom_layer(layer_type)

note: create_layer() is a global function, while create_custom_layer()
is a method of `ncnn::Net`.

It will also invoke layer->load_param(paramdict);

Input layer has index 0 since it is the first row
0 1 data -> data is the output blob of Input layer. It has index 0 since
it is the first blob

*/
static const char *s = R"(
77675157
3 3
Input          data          0 1 data
Convolution    conv0_fwd    1 1 data conv0_fwd 0=3 1=3 11=3 2=1
↪12=1 3=1 13=1 4=0 14=0 5=1 6=81
InnerProduct   dense0_fwd   1 1 conv0_fwd output 0=1 1=1 2=151875
)";
std::ofstream of("filename.param");
of << s;
of.close();

ncnn::Net net;
DataReaderFromEmpty dr;

// return 0 on success
// What does load_param() do?
// If we need to set options, we have to set it before calling load_param()
// (1) Read the magic number
// (2) Allocate space for layers and blobs
auto ret = net.load_param("filename.param");
MY_CHECK(ret, 0);

ret = net.load_model(dr);
MY_CHECK(ret, 0);

```

(continues on next page)

(continued from previous page)

```

MY_CHECK(net.blobs().size(), 3);
MY_CHECK(net.layers().size(), 3);

// test layer 0
// every layer has a type
MY_CHECK(net.layers()[0]->type, "Input");
MY_CHECK(net.layers()[0]->typeindex, ncnn::LayerType::Input);

// every layer has a name
MY_CHECK(net.layers()[0]->name, "data");

// every layer has an input vector
MY_CHECK(net.layers()[0]->bottoms.size(), 0);

// every layer has an output vector
MY_CHECK(net.layers()[0]->tops.size(), 1);
// indexes in tops are indexes into the global blobs vector
MY_CHECK(net.layers()[0]->tops[0], 0);

// every blob has a name
MY_CHECK(net.blobs()[0].name, "data");

// every blob has a producer (layer index)
MY_CHECK(net.blobs()[0].producer, 0); // producer is layer 0

// every blob has a consumer (layer index)
MY_CHECK(net.blobs()[0].consumer, 1); // consumer is layer 1
}

```

24.1.2 test-param-dict

Listing 3: ./code/hello/test-param-dict.cc

```

#include "paramdict.h"
#ifdef MY_CHECK
#define MY_CHECK(x, y) \
    do { \
        if (x != y) { \
            NCNN_LOGE("#x " != " #y"); \
            exit(EXIT_FAILURE); \
        } \
    } while (0)
#endif

/*
It has an array of struct.
The size of the array is 32, so the maximum number of parameters is 32
*/
void TestParamDict() {
    NCNN_LOGE("Test param dict\n");
}

```

(continues on next page)

(continued from previous page)

```

ncnn::ParamDict pd;

// int 2
pd.set(0, 10);
MY_CHECK(pd.type(0), 2);
MY_CHECK(pd.get(0, 0), 10);

// float 3
pd.set(1, 10.5f);
MY_CHECK(pd.type(1), 3);
// Caution: The type of the default value (the second argument) is very
// important. It determines which overload to invoke
MY_CHECK(pd.get(1, 0.f), 10.5);

// mat 4
ncnn::Mat m(1);
m[0] = 10.25;
pd.set(2, m);
MY_CHECK(pd.type(2), 4);
MY_CHECK(pd.get(2, ncnn::Mat())[0], 10.25);
}

```

24.1.3 test-data-reader

Listing 4: ./code/hello/test-data-reader.cc

```

#include "datareader.h"
#include "paramdict.h"
#include <iostream>

#ifdef MY_CHECK
#define MY_CHECK(x, y) \
do { \
    if (x != y) { \
        NCNN_LOGE("#x " != " #y"); \
        exit(EXIT_FAILURE); \
    } \
} while (0)
#endif

class ParamDict2 : public ncnn::ParamDict {
public:
    int load_param2(const ncnn::DataReader &dr) { return load_param(dr); }

    int load_param_bin2(const ncnn::DataReader &dr) { return load_param_bin(dr); }
};

void TestDataReader() {
    NCNN_LOGE("Test data reader\n");
    int32_t a = 4;
}

```

(continues on next page)

(continued from previous page)

```

FILE *fp = fopen("a.bin", "wb");
fwrite(&a, sizeof(a), 1, fp);

float f = 1.25;
fwrite(&f, sizeof(f), 1, fp);

fclose(fp);

fp = fopen("a.bin", "rb");

ncnn::DataReaderFromStdio dr(fp);
int32_t p;
float q;
dr.read(&p, sizeof(p));
dr.read(&q, sizeof(q));
MY_CHECK(p, a);
MY_CHECK(q, f);
fclose(fp);

// datareader with param dict
fp = fopen("a.txt", "w");
fprintf(fp, "0=100 1=1.250000 -23303=5,0.125,0.25,0.50,0.625,1.0");
fclose(fp);

fp = fopen("a.txt", "r");
ncnn::DataReaderFromStdio dr2(fp);
ParamDict2 pd;
pd.load_param2(dr2);
MY_CHECK(pd.get(0, 0), 100);
MY_CHECK(pd.get(1, 0.f), 1.25);
ncnn::Mat m = pd.get(3, ncnn::Mat());
MY_CHECK(m.w, 5);
MY_CHECK(m[0], 0.125);
MY_CHECK(m[1], 0.25);
MY_CHECK(m[2], 0.50);
MY_CHECK(m[3], 0.625);
MY_CHECK(m[4], 1.0);

fclose(fp);

// for binary
//
//     binary 0
//     binary 100
//     binary 1
//     binary 1.250000
//     binary 3 | array_bit
//     binary 5
//     binary 0.1
//     binary 0.2
//     binary 0.4
//     binary 0.8

```

(continues on next page)

(continued from previous page)

```
//      binary 1.0
//      binary -233(EOP)
fp = fopen("a.bin", "wb");
int32_t id = 0;
fwrite(&id, sizeof(id), 1, fp);
int32_t val = 100;
fwrite(&val, sizeof(val), 1, fp);

id = 1;
float val2 = 1.25;
fwrite(&id, sizeof(id), 1, fp);
fwrite(&val, sizeof(val2), 1, fp);

id = -23303;
fwrite(&id, sizeof(id), 1, fp);
int32_t len = m.w;
fwrite(&len, sizeof(len), 1, fp);
fwrite(m.data, sizeof(float), m.w, fp);

int32_t eop = -233;
fwrite(&eop, sizeof(eop), 1, fp);
fclose(fp);

fp = fopen("a.bin", "rb");

ncnn::DataReaderFromStdio dr3(fp);
ParamDict2 pd3;
pd3.load_param_bin2(dr3);
fclose(fp);
MY_CHECK(pd.get(0, 0), 100);

// ncnn does not know that param 1 is a float. The user has to specify that!
MY_CHECK(pd.get(1, 0.f), 1.25);

ncnn::Mat m2 = pd.get(3, ncnn::Mat());
MY_CHECK(m2.w, 5);
MY_CHECK(m2[0], 0.125);
MY_CHECK(m2[1], 0.25);
MY_CHECK(m2[2], 0.50);
MY_CHECK(m2[3], 0.625);
MY_CHECK(m2[4], 1.0);
}
```


25.1 Installation

Refer to <https://llvm.org/docs/GettingStarted.html#requirements>

<https://llvm.org/docs/GettingStarted.html#getting-a-modern-host-c-toolchain> describes how to install GCC from source.

Use of a user provided GCC:

```
mkdir build
cd build
CC=$HOME/toolchains/bin/gcc CXX=$HOME/toolchains/bin/g++ \
  cmake .. -DCMAKE_CXX_LINK_FLAGS="-Wl,-rpath,$HOME/toolchains/lib64 -L$HOME/toolchains/
  ↳lib64"
```

Useful tools that can be found in *build/bin*:

```
$ llvm-config --cxxflags
-I/ceph-fj/fangjun/open-source-2/llvm-project/llvm/include -I/ceph-fj/fangjun/open-
↳source-2/llvm-project/build/include -std=c++14 -fno-exceptions -fno-rtti -D_GNU_
↳SOURCE -D__STDC_CONSTANT_MACROS -D__STDC_FORMAT_MACROS -D__STDC_LIMIT_MACROS

$ llvm-config --libdir
/ceph-fj/fangjun/open-source-2/llvm-project/build/lib

$ llvm-config --cflags
-I/ceph-fj/fangjun/open-source-2/llvm-project/llvm/include -I/ceph-fj/fangjun/open-
↳source-2/llvm-project/build/include -D_GNU_SOURCE -D__STDC_CONSTANT_MACROS -D__STDC_
↳FORMAT_MACROS -D__STDC_LIMIT_MACROS

$ llvm-config --ldflags
-L/ceph-fj/fangjun/open-source-2/llvm-project/build/lib

$ llvm-config --src-root
/ceph-fj/fangjun/open-source-2/llvm-project/llvm

$ llvm-config --obj-root
/ceph-fj/fangjun/open-source-2/llvm-project/build

$ llvm-config --version
15.0.0git
```

(continues on next page)

(continued from previous page)

```
$ llvm-config --bindir  
/ceph-fj/fangjun/open-source-2/llvm-project/build/bin
```

25.2 ninja

```
pip install ninja
```

```
ninja
```

will look for the file *\$PWD/build.ninja*

```
ninja --help  
ninja -C build -j 20  
ninja -t targets  
ninja -t clean  
ninja -t cleandead  
  
ninja -v # be verbose while compiling files  
  
# suppose hello is a target  
ninja -v hello  
ninja -v -t clean hello  
  
ninja -n -v hello # dry run
```

25.3 Documentation

- Coding standard
<https://llvm.org/docs/CodingStandards.html>
- LLVM Developer Policy
<https://llvm.org/docs/DeveloperPolicy.html>
- doxygen doc
<https://llvm.org/doxygen/>
- <http://www.aosabook.org/en/llvm.html>

25.4 Intermediate representation

```
clang --help
```

```
-emit-llvm  Use the LLVM representation for assembler and object files
-S          Only run preprocess and compilation steps
```

- `clang -S -emit-llvm ex.c` generates a text file `ex.ll`.
- `clang -c -emit-llvm ex.c` generates a binary file `ex.bc`.
- `llvm-dis ex.bc` generates a file `ex.ll`, which is identical with the file generated using `clang -S -emit-llvm ex.c`.`
- `llvm-as ex.ll` generates a file `ex.bc`, which is identical with the file generated using `clang -c -emit-llvm ex.c`.
- `llc ex.ll` generates the assembly file `ex.s`
- `lli ex.ll` can run this file. Use `echo $?` to see the return value.

See https://llvm.org/devmtg/2019-04/slides/Tutorial-Bridgers-LLVM_IR_tutorial.pdf.

25.5 Install GCC

```
tar xvf gcc-12.2.0.tar.gz
cd gcc-12.2.0
mkdir build
cd build

unset C_INCLUDE_PATH
unset CPLUS_INCLUDE_PATH
unset LD_LIBRARY_PATH
unset LIBRARY_PATH

../configure --prefix=/ceph-fj/fangjun/software/gcc-12.2.0
make -j 5
make install
```

```
gcc_dir=/ceph-fj/fangjun/software/gcc-12.2.0
export PATH=$gcc_dir/bin:$PATH
export CC=$gcc_dir/bin/gcc
export CXX=$gcc_dir/bin/g++
export LIBRARY_PATH=$gcc_dir/lib64:$LIBRARY_PATH
export LD_LIBRARY_PATH=$gcc_dir/lib64:$LD_LIBRARY_PATH
export C_INCLUDE_PATH=$gcc_dir/include
export CPLUS_INCLUDE_PATH=$gcc_dir/include
```

25.5.1 Using conda

```
conda install -c conda-forge gcc==9.5.0  
conda install -c conda-forge gxx==9.5.0
```

26.1 kotlin

26.1.1 Install

Kotlin to java online converter: <https://www.codeconvert.ai/kotlin-to-java-converter>.

See <https://kotlinlang.org/docs/command-line.html#run-scripts>

```
wget https://github.com/JetBrains/kotlin/releases/download/v1.7.22/kotlin-compiler-1.7.22.zip
cd $HOME/software
mkdir kotlin
cd kotlin
unzip /path/to/kotlin-compiler-1.7.22.zip
mv kotlinc 1.7.22
export PATH=$HOME/software/kotlin/1.7.22/bin:$PATH
```

Hello world

Create a file `hello.kt`:

Listing 1: `./code/install/hello.kt`

```
fun main() {
    println("hello world")
}
```

Usage 1

```
kotlinc hello.kt
```

It will generate a binary file `HelloKt.class` and a folder `META-INF`.

```
$ javap HelloKt.class
```

The above command prints the following:

```
Compiled from "hello.kt"
public final class HelloKt {
    public static final void main();
    public static void main(java.lang.String[]);
}
```

We can use:

```
kotlin HelloKt.class
```

to execute it.

Usage 2

```
kotlinc hello.kt -d hello.jar
# or
kotlinc hello.kt -include-runtime -d hello.jar

# It can be run using:
java -jar hello.jar
kotlin -classpath ./hello.jar HelloKt
```

Note that `hello.jar` is actually a zip file.

kotlinc-jvm

To run kotlin inside an interactive shell, we can run the command:

```
kotlinc-jvm
```

or the command:

```
kotlinc
```

Naming convention

- class: `ClassName`
- function: `funcName`
- variable: `variableName`
- filename extension: `kt`

26.1.2 Basic types

See <https://kotlinlang.org/docs/basic-types.html>

Note `String.length` and `Array.size`. Both of them have `count()`, i.e., `String.count()` and `Array.count()`.
`String.isNotBlank()` and `String.isNotEmpty()`.

`String.isBlank()` and `String.isEmpty()` returns true for an empty string "".

Differences between `Array<Float>` and `FloatArray`:

- `Array<Float>` stores objects
- `FloatArray` is a primitive array
- `Array<Float>` to `FloatArray`: `.toFloatArray()`
- `FloatArray` to `Array<Float>`: `.toTypedArray()`

```
kotlinc -include-runtime -d hello.jar hello.kt
```

Listing 2: `./code/basic-types/hello.kt`

```
fun testInt() {  
    val a = 1  
    check(a is Int)  
    check(a == 1)  
  
    val b = 1L  
    check(b is Long)  
    check(b == 1L)  
  
    val c :Byte = -1  
    check(c is Byte)  
    check(c == (-1).toByte())  
  
    val d: Short = 1  
    check(d is Short)  
    check(d == 1.toShort())  
  
    val g = 0x0a  
    check(g is Int)  
    check(g == 10)  
  
    val f = 0b000001000  
    check(f is Int)  
    check(f == 8)  
  
    val h = 0b0000_1000  
    check(h is Int)  
    check(h == 8)  
  
    val k = 1_000  
    check(k is Int)  
    check(k == 1000)
```

(continues on next page)

(continued from previous page)

```
val m = 1 / 2
check(m is Int)
check(m == 0)

val n = 1 / 2.toFloat()
check(n is Float)
check(n == 0.5f)
}

fun testFloat() {
    val a = 1.0
    check(a is Double)
    check(a == 1.0)

    val b = 2.0F
    check(b is Float)
    check(b == 2.0F)
    check(b == (2.0).toFloat())

    // either f or F is ok
    val c = 2.0f
    check(c is Float)
    check(c == 2.0F)

    val d = (2.2F).toInt()
    check(d is Int)
    check(d == 2)
}

fun testBitwise() {
    // support only Int and Long
    val a = 0b01
    val b = 0b10

    val c = a or b // in C++: c = a | b

    check(c == 3)

    val d = a and b // in C++: d = a & b
    check(d == 0)

    val f = a shl 3 // in C++: a << 3
    check(f == 8)

    val g = b shr 1 // in C++: g = b >> 1
    check(g == 1)

    val h = a xor b // in C++: h = a ^ b
    check(h == 3)

    val k :Byte = 0b0100_0001
```

(continues on next page)

(continued from previous page)

```
// test inv()
}

fun testJavaClass() {
    val a = 1
    println(a.javaClass) // int
}

fun testBoolean() {
    val a = true
    val b = false
    val c: Boolean = true
    check(a)
    check(!b)
    check(c)

    val d = a && b
    check(d == false)

    val e = a || b
    check(e)
}

fun testChar() {
    // a char is 2-byte, utf-16
    val a = 'a'
    check(a is Char)
    check(a.isDigit() == false)

    val b = '\uff00'
    check(b is Char)

    val c = '9'
    check(c.isDigit() == true)

    check(Char.SIZE_BITS == 16)
    check('0'.code == 0x30)
    check('A'.lowercaseChar() == 'a')
    check('a'.uppercaseChar() == 'A')
}

// note String is immutable, like Python.
fun testString() {
    // a sequence of Char, utf16
    // immutable
    val a = "Abc"
    check(a + 1 == "Abc1")
    check(a.count() == 3)
    check(a.length == 3)
    check(a.lastIndex == a.length - 1)
    check(a.toUpperCase() == "ABC")
    check(a.toLowerCase() == "abc")
}
```

(continues on next page)

(continued from previous page)

```
// remove the first two characters
check(a.drop(2) == "c")

// remove the last two characters
check(a.dropLast(2) == "A")

check(a.indexOf("bc") == 1)

// multi-line strings, like python
val b = """
line 1
line 2
"""
println(b)
/*

line 1
line 2

*/

val c = """
|line 1
|line 2
""".trimMargin()
println(c)
/*
line 1
line 2
*/

val d = """
>line 1
>line 2
""".trimMargin(">")
println(d)
/*
line 1
line 2
*/

// format
val k = String.format("%x %02x", 10, 10)
check(k == "a 0a")
}

fun testArray() {
    val a = Array(3) {it}
    check(a[0] == 0)
    check(a[1] == 1)
    check(a[2] == 2)
}
```

(continues on next page)

(continued from previous page)

```

// val b = arrayOf(0, 1, 2)
val b = arrayOf<Int>(0, 1, 2)
check(a.contentEquals(b)) // we cannot use == since they save references

a[0] = 10 // assign a value
check(a[0] == 10)

check(a.sum() == 10 + 1 + 2)

check(a.joinToString() == "10, 1, 2")
}

fun shortArrayToByteArray() {
    val a: Array<Short> = arrayOf(0x0102, 0x0304)
    val b: Array<Byte> = Array(a.size * 2) {0}
    println(a.contentToString()) // [258, 772]
    println(b.contentToString()) // [0, 0, 0, 0]

    for (i in a.indices) {
        b[2*i] = (a[i].toInt() and 0xff).toByte()
        b[2*i+1] = ((a[i].toInt() shr 8) and 0xff).toByte()
    }
    println(b.contentToString()) // [2, 1, 4, 3]
}

fun byteArrayToShortArray() {
    val a: Array<Byte> = arrayOf(0x01, 0x02, 0x03, 0x04)
    val b: Array<Short> = Array(a.size / 2) {0}

    println(a.contentToString()) // [258, 772]
    println(b.contentToString()) // [0, 0, 0, 0]

    for (i in b.indices) {
        b[i] = (a[2*i].toInt() + (a[2*i+1].toInt() shl 8)).toShort()
    }
    check(b[0].toInt() == 0x0201)
    check(b[1].toInt() == 0x0403)
}

fun main() {
    testInt()
    testFloat()
    testBitwise()
        testJavaClass()
        testBoolean()
        testChar()
        testString()
    testArray()
    shortArrayToByteArray()
    byteArrayToShortArray()
}

```

CharSequence

See <https://github.com/JetBrains/kotlin/blob/0938b46726b9c6938df309098316ce741815bb55/core/builtins/native/kotlin/CharSequence>

It is an interface.

26.1.3 Collections

See:

- <https://kotlinlang.org/docs/kotlin-tour-collections.html>
- <https://kotlinlang.org/docs/collections-overview.html>

```
fun testList() {
    val a = mutableListOf(1, 2, 3)
    println(a.javaClass) // class java.util.ArrayList

    /*
    for((i, v) in a.withIndex()) {
        check(i + 1 == v)
    }
    */

    val b: List<Int> = a // view of a, no data is copied
    a[0] = 10 // b is also changed since b is a view of a
    check(b[0] == 10)

    check(a.first() == a[0])
    check(a.last() == a[a.size - 1])

    check(a.count() == a.size)
    check(10 in a)
    check(11 !in a)

    // append
    a.add(20)
    check(a.size == 4) // now there are 4 elements
    check(b.size == 4) // size of b is also changed
    check(a.last() == 20)

    a.add(1, 100) // at index 1, insert 100
    check(a[1] == 100)
    check(a[2] == 2)

    a.add(2)
    a.remove(2) // remove the first occurrence of 2
    check(a.last() == 2)
}

fun testSet() {
    val a = mutableSetOf(1, 0, 2)
    println(a.javaClass) // class java.util.LinkedHashSet
}
```

(continues on next page)

(continued from previous page)

```

println(a)
check(0 in a)
check(3 !in a)

check(a.count() == 3)
check(a.size == 3)

// remove a element
a.remove(0)
check(0 !in a)

a.add(0)
check(0 in a)
}

fun testMap() {
    val a: MutableMap<Int, String> = mutableMapOf(0 to "zero", 1 to "one", 2 to "two")
    println(a.javaClass) // class java.util.LinkedHashMap

    check(a.count() == 3)
    check(a[0] == "zero")
    check(a.containsKey(0))

    a.remove(0)
    check(0 !in a)
    check(a.containsKey(0) == false)

    a.put(0, "zero") // equivalent to a[0] = "zero"
    check(0 in a)
    check(a.containsKey(0))
}

fun main() {
    testList()
    testSet()
    testMap()
}

```

26.1.4 jni

To get the method signature, we first get the .class file of the kotlin or java code, and then use:

```
javap -s /path/to/xxx.class
```

26.1.5 Hello jni

References:

- <https://matt-moore.medium.com/kotlin-jni-for-native-code-835e93af7ddf>
- <https://docs.oracle.com/javase/7/docs/technotes/guides/jni/spec/functions.html>
- <https://developer.android.com/training/articles/perf-jni.html#primitive-arrays>

Listing 3: ./code/jni/hello/Makefile

```
CXXFLAGS := -I $(JAVA_HOME)/include
CXXFLAGS += -I $(JAVA_HOME)/include/linux

all: NativeSample.jar libhello.so

NativeSample.jar: NativeSample.kt Main.kt
    kotlinc-jvm -include-runtime -d NativeSample.jar NativeSample.kt Main.kt

libhello.so: hello.cc hello.h
    $(CXX) -o $@ -shared -fPIC $(CXXFLAGS) hello.cc

run: all
    java -jar NativeSample.jar

clean:
    $(RM) *.jar *.so
```

Note: We have to disable name mangling here.

<https://docs.oracle.com/javase/7/docs/technotes/guides/jni/spec/design.html> describe the specification:

- `Java_NativeSample_sayHello`: The prefix is fixed with `Java`. `NativeSample` is the class name in kotlin. `sayHello` is the method in kotlin.

Listing 4: ./code/jni/hello/hello.h

```
#ifndef HELLO_H
#define HELLO_H
#include "jni.h"

#ifdef __cplusplus
extern "C" {
#endif

JNIEXPORT void JNICALL Java_NativeSample_sayHello(JNIEnv *env, jobject obj);

JNIEXPORT jfloat JNICALL Java_NativeSample_sum(JNIEnv *env, jobject obj,
                                                jfloatArray array);

#ifdef __cplusplus
}
#endif

#endif // HELLO_H
```

The first argument must be `JNIEnv *env`. Since it is a non-static kotlin method, the second argument is a reference to the object. Otherwise, it is a reference to the class.

Listing 5: `./code/jni/hello/hello.cc`

```
#include "hello.h"

JNIEXPORT void JNICALL Java_NativeSample_sayHello(JNIEnv *env, jobject obj) {
    std::cout << "hello world\n";
}

JNIEXPORT jfloat JNICALL Java_NativeSample_sum(JNIEnv *env, jobject obj,
                                              jfloatArray array) {
    // the last arg is set to nullptr, which means we are not interested
    // in whether the returned pointer points to a copied region or not.
    jfloat *p = env->GetFloatArrayElements(array, nullptr);
    jsize n = env->GetArrayLength(array);

    jfloat s = 0;
    for (int32_t i = 0; i != n; ++i) {
        s += p[i];
    }
    // See
    // https://docs.oracle.com/javase/7/docs/technotes/guides/jni/spec/functions.html
    // JNI_ABORT: means to free the buffer pointed by p but don't copy the
    // buffer to array
    env->ReleaseFloatArrayElements(array, p, JNI_ABORT);

    return s;
}
```

`System.loadLibrary("hello"):`

- On Linux, it looks for `libhello.so`
- On Windows, it looks for `hello.dll`

`external fun sayHello():`

- It will look for the function with name `Java_NativeSample_sayHello`.

Listing 6: `./code/jni/hello/NativeSample.kt`

```
class NativeSample {
    companion object {
        init {
            System.loadLibrary("hello")
        }
    }

    external fun sayHello()
    external fun sum(array: FloatArray): Float
}
```

Listing 7: ./code/jni/hello/Main.kt

```

fun main() {
    var s = NativeSample()
    s.sayHello()
    var a : FloatArray = listOf(1.2f, 3.0f).toFloatArray()
    println(s.sum(a)) // 4.2

    a = floatArrayOf(3.0f, 4.0f, 5.0f)
    println(s.sum(a)) // 12.0

    a = FloatArray(100) {it.toFloat()}
    println(s.sum(a)) // 4950.0
}

```

26.1.6 Binding kaldi-native-fbank for kotlin

This note shows how to bind <https://github.com/csukuangfj/kaldi-native-fbank> for kotlin.

Listing 8: ./code/jni/feat-extractor/Makefile

```

SHERPA_NCNN_INSTALL_DIR := /ceph-fj/fangjun/open-source/sherpa-ncnn/build/install

CXXFLAGS := -I $(JAVA_HOME)/include
CXXFLAGS += -I $(JAVA_HOME)/include/linux
CXXFLAGS += -I $(SHERPA_NCNN_INSTALL_DIR)/include
CXXFLAGS += -I $(SHERPA_NCNN_INSTALL_DIR)/include/ncnn
CXXFLAGS += -Wall

LDFLAGS := -L $(SHERPA_NCNN_INSTALL_DIR)/lib -lkaldi-native-fbank-core -lsherpa-ncnn-
↳ core -lncnn
LDFLAGS += -Wl,-rpath,$(SHERPA_NCNN_INSTALL_DIR)/lib

all: main.jar libsherpa-ncnn.so

main.jar: Main.kt OnlineFeature.kt WaveReader.kt Model.kt
    kotlinc-jvm -include-runtime -d main.jar Main.kt OnlineFeature.kt WaveReader.kt_
↳ Model.kt

libsherpa-ncnn.so: online-feature.cc sherpa-ncnn.cc online-feature.h sherpa-ncnn.h
    $(CXX) -o $@ -shared -fPIC $(CXXFLAGS) online-feature.cc sherpa-ncnn.cc
↳ $(LDFLAGS)

run: all
    java -jar main.jar

clean:
    $(RM) main.jar libsherpa-ncnn.so

```


Listing 9: ./code/jni/feat-extractor/online-feature.h

```
#ifndef ONLINE_FEATURE_H_
#define ONLINE_FEATURE_H_
#include "jni.h"

#ifdef __cplusplus
extern "C" {
#endif

JNIEXPORT jlong JNICALL Java_OnlineFbank_new(JNIEnv *env, jobject obj,
                                              jobject opts);

JNIEXPORT void JNICALL Java_OnlineFbank_delete(JNIEnv *env, jobject obj,
                                              jlong ptr);

JNIEXPORT jint JNICALL Java_OnlineFbank_dim(JNIEnv *env, jobject obj,
                                              jlong ptr);

JNIEXPORT jfloat JNICALL Java_OnlineFbank_frameShiftInSeconds(JNIEnv *env,
                                                                jobject obj,
                                                                jlong ptr);

JNIEXPORT jint JNICALL Java_OnlineFbank_numFramesReady(JNIEnv *env, jobject obj,
                                                         jlong ptr);

JNIEXPORT jboolean JNICALL Java_OnlineFbank_isLastFrame(JNIEnv *env,
                                                         jobject obj, jlong ptr,
                                                         jint i);

JNIEXPORT void JNICALL Java_OnlineFbank_inputFinished(JNIEnv *env, jobject obj,
                                                         jlong ptr);

JNIEXPORT void JNICALL Java_OnlineFbank_acceptWaveform(JNIEnv *env, jobject obj,
                                                         jlong ptr,
                                                         jfloatArray samples,
                                                         jfloat sample_rate);

JNIEXPORT jfloatArray JNICALL Java_OnlineFbank_getFrame(JNIEnv *env,
                                                         jobject obj, jlong ptr,
                                                         jint i);

JNIEXPORT jfloatArray JNICALL Java_OnlineFbank_getFrames(JNIEnv *env,
                                                         jobject /*obj*/,
                                                         jlong ptr, jint start,
                                                         jint n);

#ifdef __cplusplus
}
#endif

#endif // ONLINE_FEATURE_H_
```

Listing 10: ./code/jni/feat-extractor/online-feature.cc

```

#include "online-feature.h"
#include "kaldi-native-fbank/csrc/online-feature.h"

JNIEXPORT jlong JNICALL Java_OnlineFbank_new(JNIEnv *env, jobject /*obj*/,
                                              jobject opts) {
    jclass cls = env->GetObjectClass(opts);
    jfieldID fid;

    // https://docs.oracle.com/javase/7/docs/technotes/guides/jni/spec/types.html
    // https://courses.cs.washington.edu/courses/cse341/99wi/java/tutorial/native1.1/
    ↪ implementing/field.html

    knf::FbankOptions fbank_opts;

    fid = env->GetFieldID(cls, "use_energy", "Z");
    fbank_opts.use_energy = env->GetBooleanField(opts, fid);

    fid = env->GetFieldID(cls, "energy_floor", "F");
    fbank_opts.energy_floor = env->GetFloatField(opts, fid);

    fid = env->GetFieldID(cls, "raw_energy", "Z");
    fbank_opts.raw_energy = env->GetBooleanField(opts, fid);

    fid = env->GetFieldID(cls, "htk_compat", "Z");
    fbank_opts.htk_compat = env->GetBooleanField(opts, fid);

    fid = env->GetFieldID(cls, "use_log_fbank", "Z");
    fbank_opts.use_log_fbank = env->GetBooleanField(opts, fid);

    fid = env->GetFieldID(cls, "use_power", "Z");
    fbank_opts.use_power = env->GetBooleanField(opts, fid);

    fid = env->GetFieldID(cls, "frame_opts", "LFrameExtractionOptions;");

    jobject frame_opts = env->GetObjectField(opts, fid);
    jclass frame_opts_cls = env->GetObjectClass(frame_opts);

    fid = env->GetFieldID(frame_opts_cls, "samp_freq", "F");
    fbank_opts.frame_opts.samp_freq = env->GetFloatField(frame_opts, fid);

    fid = env->GetFieldID(frame_opts_cls, "frame_shift_ms", "F");
    fbank_opts.frame_opts.frame_shift_ms = env->GetFloatField(frame_opts, fid);

    fid = env->GetFieldID(frame_opts_cls, "frame_length_ms", "F");
    fbank_opts.frame_opts.frame_length_ms = env->GetFloatField(frame_opts, fid);

    fid = env->GetFieldID(frame_opts_cls, "dither", "F");
    fbank_opts.frame_opts.dither = env->GetFloatField(frame_opts, fid);

    fid = env->GetFieldID(frame_opts_cls, "preemph_coeff", "F");

```

(continues on next page)

(continued from previous page)

```

fbank_opts.frame_opts.preemph_coeff = env->GetFloatField(frame_opts, fid);

fid = env->GetFieldID(frame_opts_cls, "remove_dc_offset", "Z");
fbank_opts.frame_opts.remove_dc_offset =
    env->GetBooleanField(frame_opts, fid);

fid = env->GetFieldID(frame_opts_cls, "window_type", "Ljava/lang/String;");
jstring window_type = (jstring)env->GetObjectField(frame_opts, fid);
const char *p_window_type = env->GetStringUTFChars(window_type, nullptr);
fbank_opts.frame_opts.window_type = p_window_type;
env->ReleaseStringUTFChars(window_type, p_window_type);

fid = env->GetFieldID(frame_opts_cls, "round_to_power_of_two", "Z");
fbank_opts.frame_opts.round_to_power_of_two =
    env->GetBooleanField(frame_opts, fid);

fid = env->GetFieldID(frame_opts_cls, "blackman_coeff", "F");
fbank_opts.frame_opts.blackman_coeff = env->GetFloatField(frame_opts, fid);

fid = env->GetFieldID(frame_opts_cls, "snip_edges", "Z");
fbank_opts.frame_opts.snip_edges = env->GetBooleanField(frame_opts, fid);

fid = env->GetFieldID(frame_opts_cls, "max_feature_vectors", "I");
fbank_opts.frame_opts.max_feature_vectors = env->GetIntField(frame_opts, fid);

fid = env->GetFieldID(cls, "mel_opts", "LMelBanksOptions;");
jobject mel_opts = env->GetObjectField(opts, fid);
jclass mel_opts_cls = env->GetObjectClass(mel_opts);

fid = env->GetFieldID(mel_opts_cls, "num_bins", "I");
fbank_opts.mel_opts.num_bins = env->GetIntField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "low_freq", "F");
fbank_opts.mel_opts.low_freq = env->GetFloatField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "high_freq", "F");
fbank_opts.mel_opts.high_freq = env->GetFloatField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "vtln_low", "F");
fbank_opts.mel_opts.vtln_low = env->GetFloatField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "vtln_high", "F");
fbank_opts.mel_opts.vtln_high = env->GetFloatField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "debug_mel", "Z");
fbank_opts.mel_opts.debug_mel = env->GetBooleanField(mel_opts, fid);

fid = env->GetFieldID(mel_opts_cls, "htk_mode", "Z");
fbank_opts.mel_opts.htk_mode = env->GetBooleanField(mel_opts, fid);

auto online_fbank = new knf::OnlineFbank(fbank_opts);

```

(continues on next page)

(continued from previous page)

```

    return (jlong)online_fbank;
}

JNIEXPORT void JNICALL Java_OnlineFbank_delete(JNIEnv *env, jobject obj,
                                                jlong ptr) {
    delete reinterpret_cast<knf::OnlineFbank *>(ptr);
}

JNIEXPORT jint JNICALL Java_OnlineFbank_dim(JNIEnv *env, jobject obj,
                                             jlong ptr) {
    return reinterpret_cast<const knf::OnlineFbank *>(ptr)->Dim();
}

JNIEXPORT jfloat JNICALL Java_OnlineFbank_frameShiftInSeconds(JNIEnv *env,
                                                                jobject obj,
                                                                jlong ptr) {
    return reinterpret_cast<const knf::OnlineFbank *>(ptr)->FrameShiftInSeconds();
}

JNIEXPORT jint JNICALL Java_OnlineFbank_numFramesReady(JNIEnv *env, jobject obj,
                                                         jlong ptr) {
    return reinterpret_cast<const knf::OnlineFbank *>(ptr)->NumFramesReady();
}

JNIEXPORT jboolean JNICALL Java_OnlineFbank_isLastFrame(JNIEnv *env,
                                                         jobject obj, jlong ptr,
                                                         jint i) {
    return reinterpret_cast<const knf::OnlineFbank *>(ptr)->IsLastFrame(i);
}

JNIEXPORT void JNICALL Java_OnlineFbank_inputFinished(JNIEnv *env, jobject obj,
                                                        jlong ptr) {
    reinterpret_cast<knf::OnlineFbank *>(ptr)->InputFinished();
}

JNIEXPORT void JNICALL Java_OnlineFbank_acceptWaveform(JNIEnv *env, jobject obj,
                                                         jlong ptr,
                                                         jfloatArray samples,
                                                         jfloat sample_rate) {
    jfloat *p = env->GetFloatArrayElements(samples, nullptr);
    jsize n = env->GetArrayLength(samples);

    reinterpret_cast<knf::OnlineFbank *>(ptr)->AcceptWaveform(sample_rate, p, n);

    env->ReleaseFloatArrayElements(samples, p, JNI_ABORT);
}

JNIEXPORT jfloatArray JNICALL Java_OnlineFbank_getFrame(JNIEnv *env,
                                                         jobject obj, jlong ptr,
                                                         jint i) {
    auto online_fbank = reinterpret_cast<const knf::OnlineFbank *>(ptr);
    auto frame = online_fbank->GetFrame(i);

```

(continues on next page)

(continued from previous page)

```

    auto dim = online_fbank->Dim();

    jfloatArray ans = env->NewFloatArray(dim);
    env->SetFloatArrayRegion(ans, 0, dim, frame);

    return ans;
}

JNIEXPORT jfloatArray JNICALL Java_OnlineFbank_getFrames(JNIEnv *env,
                                                         jobject /*obj*/,
                                                         jlong ptr, jint start,
                                                         jint n) {
    auto online_fbank = reinterpret_cast<const knf::OnlineFbank *>(ptr);
    auto dim = online_fbank->Dim();

    if (start + n > online_fbank->NumFramesReady()) {
        return nullptr;
    }

    jfloatArray ans = env->NewFloatArray(n * dim);
    for (int32_t i = 0; i != n; ++i) {
        auto frame = online_fbank->GetFrame(start + i);
        env->SetFloatArrayRegion(ans, i * dim, dim, frame);
    }

    return ans;
}

```

Listing 11: ./code/jni/feat-extractor/sherpa-ncnn.h

```

#ifndef SHERPA_NCNN_H
#define SHERPA_NCNN_H
#include "jni.h"

#ifdef __cplusplus
extern "C" {
#endif

JNIEXPORT jfloatArray JNICALL Java_WaveReader_00024Companion_readWave(
    JNIEnv *env, jclass cls, jstring filename, jfloat expected_sample_rate);

JNIEXPORT jlong JNICALL Java_Model_new(JNIEnv *env, jobject /*obj*/,
                                       jobject config);

JNIEXPORT void JNICALL Java_Model_delete(JNIEnv *env, jobject /*obj*/,
                                         jlong ptr);

JNIEXPORT jint JNICALL Java_Model_segment(JNIEnv *env, jobject /*obj*/,
                                          jlong ptr);

JNIEXPORT jint JNICALL Java_Model_offset(JNIEnv *env, jobject /*obj*/,
                                          jlong ptr);

```

(continues on next page)

(continued from previous page)

```

#ifdef __cplusplus
}
#endif

#endif // SHERPA_NCNN_H

```

Listing 12: ./code/jni/feat-extractor/sherpa-ncnn.cc

```

#include "sherpa-ncnn.h"
#include "sherpa-ncnn/csrc/model.h"
#include "sherpa-ncnn/csrc/wave-reader.h"
#include <iostream>

JNIEXPORT jfloatArray JNICALL Java_WaveReader_00024Companion_readWave(
    JNIEnv *env, jclass cls, jstring filename, jfloat expected_sample_rate) {

    const char *p_filename = env->GetStringUTFChars(filename, nullptr);

    bool is_ok = false;
    std::vector<float> samples =
        sherpa_ncnn::ReadWave(p_filename, expected_sample_rate, &is_ok);
    env->ReleaseStringUTFChars(filename, p_filename);

    if (!is_ok) {
        return nullptr;
    }

    jfloatArray ans = env->NewFloatArray(samples.size());
    env->SetFloatArrayRegion(ans, 0, samples.size(), samples.data());
    return ans;
}

JNIEXPORT jlong JNICALL Java_Model_new(JNIEnv *env, jobject /*obj*/,
                                       jobject config) {
    sherpa_ncnn::ModelConfig model_config;

    jclass cls = env->GetObjectClass(config);

    jfieldID fid = env->GetFieldID(cls, "encoderParam", "Ljava/lang/String;");
    jstring s = (jstring)env->GetObjectField(config, fid);
    const char *p = env->GetStringUTFChars(s, nullptr);
    model_config.encoder_param = p;
    env->ReleaseStringUTFChars(s, p);

    fid = env->GetFieldID(cls, "encoderBin", "Ljava/lang/String;");
    s = (jstring)env->GetObjectField(config, fid);
    p = env->GetStringUTFChars(s, nullptr);
    model_config.encoder_bin = p;
    env->ReleaseStringUTFChars(s, p);

    fid = env->GetFieldID(cls, "decoderParam", "Ljava/lang/String;");

```

(continues on next page)

(continued from previous page)

```

s = (jstring)env->GetObjectField(config, fid);
p = env->GetStringUTFChars(s, nullptr);
model_config.decoder_param = p;
env->ReleaseStringUTFChars(s, p);

fid = env->GetFieldID(cls, "decoderBin", "Ljava/lang/String;");
s = (jstring)env->GetObjectField(config, fid);
p = env->GetStringUTFChars(s, nullptr);
model_config.decoder_bin = p;
env->ReleaseStringUTFChars(s, p);

fid = env->GetFieldID(cls, "joinerParam", "Ljava/lang/String;");
s = (jstring)env->GetObjectField(config, fid);
p = env->GetStringUTFChars(s, nullptr);
model_config.joiner_param = p;
env->ReleaseStringUTFChars(s, p);

fid = env->GetFieldID(cls, "joinerBin", "Ljava/lang/String;");
s = (jstring)env->GetObjectField(config, fid);
p = env->GetStringUTFChars(s, nullptr);
model_config.joiner_bin = p;
env->ReleaseStringUTFChars(s, p);

fid = env->GetFieldID(cls, "numThreads", "I");
model_config.num_threads = env->GetIntField(config, fid);
std::cout << model_config.ToString() << "\n";

auto model = sherpa_ncnn::Model::Create(model_config);

return (jlong)(model.release());
}

JNIEXPORT void JNICALL Java_Model_delete(JNIEnv *env, jobject /*obj*/,
                                           jlong ptr) {
    delete reinterpret_cast<sherpa_ncnn::Model *>(ptr);
}

JNIEXPORT jint JNICALL Java_Model_segment(JNIEnv *env, jobject /*obj*/,
                                           jlong ptr) {
    return reinterpret_cast<const sherpa_ncnn::Model *>(ptr)->Segment();
}

JNIEXPORT jint JNICALL Java_Model_offset(JNIEnv *env, jobject /*obj*/,
                                           jlong ptr) {
    return reinterpret_cast<const sherpa_ncnn::Model *>(ptr)->Offset();
}

```

Listing 13: ./code/jni/feat-extractor/Main.kt

```

fun main() {
    var fbank_opts = FbankOptions()
    fbank_opts.mel_opts.num_bins = 80
}

```

(continues on next page)

(continued from previous page)

```

    var online_fbank = OnlineFbank(fbank_opts)

    var samples = WaveReader.readWave("./1089-134686-0001.wav", 16000.0f)
    if(samples != null) {
        online_fbank.acceptWaveform(samples)
    }
    online_fbank.inputFinished()

    var modelConfig = ModelConfig(
        encoderParam="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪encoder_jit_trace-epoch-30-avg-10-pnnx.ncnn.param",
        encoderBin="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪encoder_jit_trace-epoch-30-avg-10-pnnx.ncnn.bin",
        decoderParam="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪decoder_jit_trace-epoch-30-avg-10-pnnx.ncnn.param",
        decoderBin="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪decoder_jit_trace-epoch-30-avg-10-pnnx.ncnn.bin",
        joinerParam="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪joiner_jit_trace-epoch-30-avg-10-pnnx.ncnn.param",
        joinerBin="./sherpa-ncnn-conv-emformer-transducer-2022-12-04/
↪joiner_jit_trace-epoch-30-avg-10-pnnx.ncnn.bin",
        numThreads=4,
    )
    var model = Model(modelConfig)
    println("segment: ${model.segment}")
    println("offset: ${model.offset}")
}

```

Listing 14: ./code/jni/feat-extractor/OnlineFeature.kt

```

data class FrameExtractionOptions(
    var samp_freq: Float = 16000.0f,
    var frame_shift_ms: Float = 10.0f,
    var frame_length_ms: Float = 25.0f,
    var dither: Float = 0.0f,
    var preemph_coeff: Float = 0.97f,
    var remove_dc_offset: Boolean = true,
    var window_type: String = "povey",
    var round_to_power_of_two: Boolean = true,
    var blackman_coeff: Float = 0.42f,
    var snip_edges: Boolean = true,
    var max_feature_vectors: Int = -1
)

data class MelBanksOptions(
    var num_bins : Int = 25,
    var low_freq : Float = 20.0f,
    var high_freq : Float = 0.0f,
    var vtln_low : Float = 100.0f,
    var vtln_high : Float = -500.0f,
    var debug_mel : Boolean = false,

```

(continues on next page)

(continued from previous page)

```

    var htk_mode : Boolean = false,
)

data class FbankOptions(
    var frame_opts: FrameExtractionOptions = FrameExtractionOptions(),
    var mel_opts: MelBanksOptions = MelBanksOptions(),
    var use_energy: Boolean = false,
    var energy_floor: Float = 0.0f,
    var raw_energy: Boolean = true,
    var htk_compat: Boolean = false,
    var use_log_fbank: Boolean = true,
    var use_power: Boolean = true,
)

class OnlineFbank(var opts: FbankOptions) {
    private var ptr: Long

    init {
        ptr = new(opts)
    }

    protected fun finalize() {
        delete(ptr)
    }

    val dim: Int
        get() = dim(ptr)

    val frameShiftInSeconds: Float
        get() = frameShiftInSeconds(ptr)

    val numFramesReady: Int
        get() = numFramesReady(ptr)

    fun isLastFrame(i: Int) :Boolean = isLastFrame(ptr, i)
    fun inputFinished() = inputFinished(ptr)
    fun acceptWaveform(samples: FloatArray) = acceptWaveform(ptr, samples, opts.frame_opts.
    ← samp_freq)
    fun getFrame(i: Int): FloatArray = getFrame(ptr, i)
    fun getFrames(start: Int, n: Int): FloatArray = getFrames(ptr, start, n)

    private external fun new(opts: FbankOptions): Long
    private external fun delete(ptr: Long)
    private external fun dim(ptr: Long): Int
    private external fun frameShiftInSeconds(ptr: Long): Float
    private external fun numFramesReady(ptr: Long): Int
    private external fun isLastFrame(ptr: Long, i: Int): Boolean
    private external fun inputFinished(ptr: Long)
    private external fun acceptWaveform(ptr: Long, samples: FloatArray, sample_rate: Float)
    private external fun getFrame(ptr: Long, i: Int): FloatArray
    private external fun getFrames(ptr: Long, start: Int, n: Int): FloatArray

```

(continues on next page)

(continued from previous page)

```

companion object {
    init {
        System.loadLibrary("sherpa-ncnn")
    }
}

```

Listing 15: ./code/jni/feat-extractor/WaveReader.kt

```

class WaveReader {

    companion object {
        // Read a mono wave file.
        // No resampling is made.
        external fun readWave(filename: String, expected_sample_rate: Float = 16000.0f) :
        ↪FloatArray?

        init {
            System.loadLibrary("sherpa-ncnn")
        }
    }
}

```

Listing 16: ./code/jni/feat-extractor/Model.kt

```

data class ModelConfig(
    var encoderParam: String,
    var encoderBin: String,
    var decoderParam: String,
    var decoderBin: String,
    var joinerParam: String,
    var joinerBin: String,
    var numThreads: Int = 4,
)

class Model(var config: ModelConfig) {
    private var ptr: Long

    init {
        ptr = new(config)
    }

    protected fun finalize() {
        delete(ptr)
    }

    val segment: Int
        get() = segment(ptr)

    val offset: Int
        get() = offset(ptr)
}

```

(continues on next page)

(continued from previous page)

```
private external fun new(config: ModelConfig): Long
private external fun delete(ptr: Long)
private external fun segment(ptr: Long): Int
private external fun offset(ptr: Long): Int

}
```

26.1.7 References

- <https://github.com/JetBrains/kotlin>
Source code of Kotlin.
- <https://kotlin-quick-reference.com/>
An online book

26.1.8 Control flows

See <https://kotlinlang.org/docs/control-flow.html>

```
kotlinc -include-runtime -d hello.jar hello.kt
```

Listing 17: ./code/control-flows/hello.kt

```
fun testIf() {
    val a = 1
    val b = 2
    var max = -1
    if (a > b) {
        max = a
    } else {
        max = b
    }
    check(max == b)

    // if can be used in an expression
    val min = if (a < b) {a} else {b}
    check(min == a)

    //
    if (true && true || false) {
        check(true)
    }
}

fun testWhen() {
    val a = 1
    var c: Int
    when(a) {
```

(continues on next page)

(continued from previous page)

```

    1 -> c = 10
    2 -> c = 20
    else -> c = 30
}
check(c == 10)

val d = when {
    a < 0 -> 3
    a == 0 -> 10
    a == 1 -> 30
    else -> 5
}
check(d == 30)

val k: Any = 1
val b = when (k) {
    is Int -> 10
    is Float -> 20
    else -> 30
}
check(b == 10)
}

@OptIn(kotlin.ExperimentalStdlibApi::class)
fun testRange() {
    check(5 in 0..5) // [0,5]
    check(5 !in 0..<5) // [0,5)

    check(4 in 4 downTo 1) // [4,3,2,1]
    check(1 in 4 downTo 1) // [4,3,2,1]

    check(0 in 0..5 step 3)
    check(1 !in 0..5 step 3)
    check(2 !in 0..5 step 3)
    check(3 in 0..5 step 3)
    check(4 !in 0..5 step 3)
    check(5 !in 0..5 step 3)

    check(5 in 5 downTo 0 step 3)
    check(4 !in 5 downTo 0 step 3)
    check(3 !in 5 downTo 0 step 3)
    check(2 in 5 downTo 0 step 3)
    check(1 !in 5 downTo 0 step 3)
    check(0 !in 5 downTo 0 step 3)

    val c = 'a'
    check(c in 'a' .. 'z' || c in 'A' .. 'Z')
    check(c !in '0' .. '9')

    val d = '0'
    check(d in '0' .. '9')
    check(d !in 'a'..'z' && d !in 'A'..'Z')

```

(continues on next page)

(continued from previous page)

```
}

fun testFor() {
    var a: Array<Int> = Array(5) {0}

    for (i in 0..4) {
        a[i] = i
    }

    check(a[0] == 0)
    check(a[1] == 1)
    check(a[2] == 2)
    check(a[3] == 3)

    for(i in 4 downTo 0) {
        a[i] = 4 - i
    }

    check(a[4] == 0);
    check(a[3] == 1);
    check(a[2] == 2);
    check(a[1] == 3);
    check(a[0] == 4);

    for (i in a.indices) {
        when (i) {
            0 -> check(a[i] == 4)
            1 -> check(a[i] == 3)
            2 -> check(a[i] == 2)
            3 -> check(a[i] == 1)
            4 -> check(a[i] == 0)
        }
    }
}

/*
for ((i, value) in a.withIndexes()) {
    when (i) {
        0 -> check(value == 4)
        1 -> check(value == 3)
        2 -> check(value == 2)
        3 -> check(value == 1)
        4 -> check(value == 0)
    }
}
*/

fun main() {
    testIf()
    testWhen()
    testRange()
    testFor()
}
```

26.1.9 Functions

See <https://kotlinlang.org/docs/functions.html>

typealias:

```
// https://kotlinlang.org/docs/type-aliases.html
typealias ClickHandler = (Button, ClickEvent) -> Unit
```

```
kotlinc -include-runtime -d hello.jar hello.kt
```

Listing 18: ./code/functions/hello.kt

```
fun sum1(x: Int, y: Int): Int {
    return x + y
}

// ignore the {} and return
fun sum2(x: Int, y: Int): Int = x + y

// ignore the return type hint
fun sum3(x: Int, y: Int) = x + y

fun testSum() {
    check(sum1(1, 2) == 3)
    check(sum2(1, 2) == 3)
    check(sum3(1, 2) == 3)
}

fun double(x: Double): Double {
    return x * 2
}

// we can use trailing comma in the function parameters
//
@OptIn(kotlin.ExperimentalStdlibApi::class)
fun powerOf(
    x: Int,
    exponent: Int,
): Int {
    var n = 1
    for (i in 0 ..< exponent) {
        n *= x
    }
    return n
}

fun default(a: Int, b: Int = 2): Int {
    return a + b
}

// we can call a.size in setting the default value for b!!!
fun default2(a: Array<Float>, b: Int = a.size): Float {
```

(continues on next page)

(continued from previous page)

```
    return a[0] + b
}

fun testDouble() {
    val x = double(2.5)
    check(x == 5.0)
}

fun testPowerOf() {
    val x = powerOf(2, 3)
    check(x == 8)
}

fun testDefault() {
    val x = default(1)
    check(x == 3)

    val y = default(1, 3)
    check(y == 4)
}

fun testDefault2() {
    val a: Array<Float> = arrayOf(1.5f, 2.0f, 3.0f)
    val b = default2(a)
    check(b == 1.5f+3)

    val c = default2(a, 10)
    check(c == 1.5f+10)
}

fun testLambda() {
    val makeUppercase = {x: String -> x.uppercase()}
    check(makeUppercase("ab") == "AB")
}

fun main() {
    testSum()
    testDouble()
    testPowerOf()
    testDefault()
    testDefault2()
    testLambda()
}
```

26.1.10 Package

26.1.11 for

```
val numbers = arrayOf(1, 2, 3)
for (n in numbers) {
    println(n)
}

for (i in numbers.indices) {
    println(numbers[i])
}

for ((i, value) in numbers.withIndex()) {
    check(numbers[i] == value)
}

// forEach
val a = arrayOf(1, 2, 3)
a.forEach(::println)
```

26.1.12 enum

```
enum class Weekday {
    Monday,
    Tuesday,
    Wednesday,
    Thursday,
}

for(day in Weekday.values()) {
    // day.ordinal: 0, 1, 2, 3, 4
    // day.name: Monday, Tuesday, Wednesday, Thursday
}

Weekday.valueOf("Monday") == Weekday.Monday
```

26.2 Basics

There are two tools: NDK and SDK. Each tool has a version.

Android has a concept of `Android native API level`.

There are three environment variables to set:

- `ANDROID_NDK_ROOT`
- `ANDROID_SDK_ROOT`
- `ANDROID_HOME`

The name of the app is determined by `<application android:label` in `AndroidManifest.xml`. We can use `@string/app_name`.

The title of the nav bar is determined by `<activity android:label` in `AndroidManifest.xml`.

Button click:

```
val button: Button = findViewById(R.id.button1)
button.setOnClickListener {
    Toast.makeText(this, "Clicked me!", Toast.LengthShort).show()
}
```

The menu is added in `res/menu`. The directory `res/menu` is created by us manually. After creating the directory, we can right click it, and select menu resource file. If we name it `main`, it will create a file `main.xml` inside `res/menu`. Inside `main.xml`, each menu item contains something like below:

```
<item android:id="@+id/add_item" android:title="Add" />
```

After creating `main.xml`, click the menu Code -> Override Method, select `onCreateOptionsMenu`:

```
menuInflater.inflate(R.menu.main, menu)
return true
```

To define a function when a menu item is clicked, click the menu Code->Override Method, select `onOptionsItemSelected`.

```
when (item.itemId) {
    R.id.add_item -> { xxx }
    R.id.remove_item -> { xxx }
}
return true
```

To destroy an activity, call `finish()`.

26.2.1 TextView

It is a label in other GUI.

```
<TextView
    android:id="@+id/textView"
    android:text="hello world"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
/>
```

To change the alignment, use `android:gravity`.

```
android:gravity="center"
// top, bottom, center, start, end
// center_vertical|center_horizontal
```

To change the color, use `android:textColor`

```
android:textColor="#00ff00"
```

To change the font size, use `android:textSize="24sp"`

26.2.2 Button

```
<Button
    android:id="@+id/button"
    android:text="Button"
    android:textAllCaps="false"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
/>
```

26.2.3 EditText

```
<EditText
    android:id="@+id/editText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="this is some hint"
    android:maxLines="10"
/>
```

To get the text of EditText in the kotlin code, use:

```
val inputText = editText.text.toString()
```

26.2.4 LinearLayout

```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
/>
```

orientation can also be horizontal.

26.3 Installation

26.3.1 Install NDK on Linux (not recommended)

(Use the following SDK to install NDK)

See <https://developer.android.com/ndk/downloads/index.html>

See [https://www.cryptopp.com/wiki/Android_Setup_\(Command_Line\)](https://www.cryptopp.com/wiki/Android_Setup_(Command_Line)) for details.

```
wget https://dl.google.com/android/repository/android-ndk-r25-linux.zip
unzip -d /ceph-fj/fangjun/software android-ndk-r25-linux.zip
# It will create /ceph-fj/fangjun/software/android-ndk-r25
```

Other versions can be downloaded from <https://github.com/android/ndk/wiki/Unsupported-Downloads>

```
wget https://dl.google.com/android/repository/android-ndk-r24-linux.zip
unzip -d /ceph-fj/fangjun/software android-ndk-r24-linux.zip

# It will create /ceph-fj/fangjun/software/android-ndk-r24
```

Now create a symlink in /ceph-fj/fangjun/software:

```
cd /ceph-fj/fangjun/software
ln -s android-ndk-r25 android-ndk
```

Set the following environment variable:

```
export ANDROID_NDK_ROOT=/ceph-fj/fangjun/software/android-ndk
export PATH=$ANDROID_NDK_ROOT:$PATH
```

26.3.2 Install SDK on Linux

Download the commandline tools only from <https://developer.android.com/studio#downloads>

```
wget https://dl.google.com/android/repository/commandlinetools-linux-8512546_latest.zip
unzip -d /ceph-fj/fangjun/software/android-sdk ./commandlinetools-linux-8512546_latest.
  ↪ zip
# Everything is inside /ceph-fj/fangjun/software/android-sdk/cmdline-tools/
cd /ceph-fj/fangjun/software/android-sdk
mv cmdline-tools latest
mkdir cmdline-tools
mv latest ./cmdline-tools/
```

If we don't run `mv cmdline-tools latest`, it will throw the following error

```
$ /ceph-fj/fangjun/software/android-sdk/cmdline-tools/bin/sdkmanager --list
```

```
Error: Could not determine SDK root.
Error: Either specify it explicitly with --sdk_root= or move this package into its
  ↪ expected location: <sdk>/cmdline-tools/latest/
```

```
sdkmanager --update
sdkmanager --list
```

```
# Install the build tools
sdkmanager "platforms;android-28" "build-tools;28.0.3"
# It will create the following directories inside /ceph-fj/fangjun/software/android-sdk/
#
# build-tools, emulator, licenses, patcher, platform-tools, platforms, tools
```

```
$ sdkmanager --list_installed
Installed packages:=====] 100% Fetch remote repository...
  Path                | Version | Description                | Location
  -----            | -
  build-tools;28.0.3   | 28.0.3  | Android SDK Build-Tools 28.0.3 | build-tools/28.0.3
  emulator             | 31.3.10 | Android Emulator            | emulator
```

(continues on next page)

(continued from previous page)

patcher;v4	1	SDK Patch Applier v4	patcher/v4
platform-tools	33.0.2	Android SDK Platform-Tools	platform-tools
platforms;android-28	6	Android SDK Platform 28	platforms/android-28

```

sdkmanager --help
yes | sdkmanager --licenses # to accept or licenses, the decision is saved in a cache
                             ↪ file.

```

Set the following environment variables:

```

export ANDROID_SDK_ROOT=/ceph-fj/fangjun/software/android-sdk
export PATH=$ANDROID_SDK_ROOT/cmdline-tools/latest/bin:$PATH

export ANDROID_HOME=/ceph-fj/fangjun/software/android-sdk

# for emulator
export PATH=$ANDROID_SDK_ROOT/emulator:$PATH

# for adb
export PATH=$ANDROID_SDK_ROOT/platform-tools:$PATH

# We installed 28.0.3 before
export PATH=$ANDROID_SDK_ROOT/build-tools/28.0.3:$PATH # change it for different versions

```

Now install NDK using sdkmanager:

```

# sdkmanager --list | grep ndk
sdkmanager "ndk;21.0.6113669"
# it will download android-ndk-r21 and will generate
# android-sdk/ndk/21.0.6113669

export ANDROID_NDK_HOME=$ANDROID_SDK_ROOT/ndk/21.0.6113669
export PATH=$ANDROID_NDK_ROOT:$PATH

```

26.3.3 emulator

```

find /Users/fangjun/software/my-android -name emulator
/Users/fangjun/software/my-android/emulator -list-avds
/Users/fangjun/software/my-android/emulator -avd Pixel_4_API_31 -dns-server 8.8.8.8

```

Refer to <https://developer.android.com/studio/run/emulator-networking#proxy> to setup the proxy.

26.4 cmake

See

- <https://cmake.org/cmake/help/latest/manual/cmake-toolchains.7.html>
- <https://developer.android.com/ndk/guides/cmake#command-line>
- `/ceph-fj/fangjun/software/android-ndk/build/cmake/android.toolchain.cmake`

User provided:

- `ANDROID_NDK`: Set to the path of `android-ndk`
- `ANDROID_ABI`: `armeabi-v7a`, `arm64-v8a`, `x86`, `x86_64`,
- `ANDROID_PLATFORM`
- `ANDROID_NATIVE_API_LEVEL`
- `ANDROID_TOOLCHAIN`
- `ANDROID`: `TRUE`
- `CMAKE_SYSTEM_NAME`: `Anroid`
- `ANDROID_STL`
- `ANDROID_HOST_TAG`

Auto generated:

- `ANDROID_NDK_MAJOR` - see `android-ndk/source.properties`
- `ANDROID_NDK_MINOR` - see `android-ndk/source.properties`
- `ANDROID_NDK_BUILD` - see `android-ndk/source.properties`
- `ANDROID_NDK_REVISION` - see `android-ndk/source.properties`
- `ANDROID_TOOLCHAIN_ROOT`
- `ANDROID_C_COMPILER`

26.5 hello

In this note, we describe how to build an executable, how to create an emulator, and how to run the executable in the emulator via `adb push`, `adb shell`.

Listing 19: `./code/hello/hello.cc`

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "hello world\n";
5     return 0;
6 }
```

Listing 20: ./code/hello/CMakeLists

```

1 cmake_minimum_required(VERSION 3.8)
2
3 project(hello)
4
5 add_executable(hello hello.cc)

```

26.5.1 x86

```

export ANDROID_NDK_ROOT=/ceph-fj/fangjun/software/android-ndk
cmake -DCMAKE_TOOLCHAIN_FILE=$ANDROID_NDK_ROOT/build/cmake/android.toolchain.cmake -
↳DANDROID_ABI=x86 ..

```

Other values:

- -DANDROID_ABI="arm64-v8a"
- -DANDROID_ABI="armeabi-v7a"
- -DANDROID_ARM_NEON=ON
- -DANDROID_PLATFORM=android-21
- -DANDROID_PLATFORM=android-24

```

$ sdkmanager --list | grep system-images | grep x86 | grep android-28
$ sdkmanager "system-images;android-28;default;x86"

$ which avdmanager
/ceph-fj/fangjun/software/android-sdk/cmdline-tools/latest/bin/avdmanage

$ avdmanager create avd --help
$ -k --package : Package path of the system image for this AVD
#                (e.g., 'system-images;android-19;google_apis;x86').
$ -n --name      : Name of the new AVD [required]
$ -b --abi       : The ABI to use for the AVD. The default is to auto-select
#                the ABI if the platform has only one ABI for its system images
$ -g --tag       : The sys-img tag to use for the AVD. The default is to
#                auto-select if the platform has only one tag for its system
#                images
$ avdmanager create avd -k "system-images;android-28;default;x86" -n hello -b x86 -g
↳default
# Use the default option [no] when it prompts:
# Do you wish to create a custom hardware profile? [no]
#
$ avdmanager delete avd -n hello # to delete it

```

```

$ avdmanager list avd
Available Android Virtual Devices:
  Name: hello
  Path: /root/fangjun/.android/avd/hello.avd
  Target: Default Android System Image
         Based on: Android 9.0 (Pie) Tag/ABI: default/x86
  Sdcard: 512 MB

```

```
emulator -avd hello -no-window -no-accel # then, open a new terminal
```

```
$ adb devices
```

```
* daemon not running; starting now at tcp:5037
* daemon started successfully
List of devices attached
emulator-5554    offline
```

```
# push the binary from code/hello/build/hello
```

```
adb push ./code/hello/build/hello /sdcard # not able to use chmod +x in it
adb push ./code/hello/build/hello /data/local
adb shell /data/local/hello
adb shell
generic_x86:/ #
```

```
generic_x86:/ # ./data/local/hello
hello world
```

26.6 Android.mk

See https://developer.android.com/ndk/guides/android_mk

26.6.1 hello

Listing 21: ./code/android_mk/hello/jni/foo.cc

```
1 #include <iostream>
2 int main() { std::cout << "hello world\n"; }
```

Listing 22: ./code/android_mk/hello/jni/Android.mk

```
1 LOCAL_PATH := $(call my-dir)
2 include $(CLEAR_VARS)
3 LOCAL_MODULE := foo
4 LOCAL_SRC_FILES := foo.cc
5 # include $(BUILD_SHARED_LIBRARY)
6 include $(BUILD_EXECUTABLE)
```

Listing 23: ./code/android_mk/hello/jni/Application.mk

```
1 APP_ABI := x86
2 APP_STL := c++_shared
```

APP_STL := c++_shared is to fix the following errors:

```
ld: error: undefined symbol: std::__ndk1::cout
```

To compile:

```
cd code/android_mk/hello
ndk-build
```

It will generate two directories in hello: libs and obj..

```
adb push libs/x86/foo /data/local
adb push libs/x86/libc++_shared.so /data/local
adb shell
cd /data/local
export LD_LIBRARY_PATH=.
./foo
```

26.7 adb

26.7.1 install on macos

```
wget https://dl.google.com/android/repository/platform-tools-latest-darwin.zip
# unzip it and you will find the binary `adb`
```

26.7.2 install on windows

Go to <https://adbshell.com/downloads> to download it.

26.7.3 install on Linux

```
wget https://dl.google.com/android/repository/platform-tools-latest-linux.zip
# unzip it
```

26.7.4 run binaries

```
adb push ./hello-world /data/local/tmp/
```

26.7.5 paste text to android emulator

```
adb shell "input text 'hello world'"
```


26.8 tts

Install Android Accessibility Suite via Google Play.

- <https://github.com/Miserlou/Android-SDK-Samples/tree/master/TtsEngine>
- <https://github.com/benjaminwan/ChineseTtsTflite>

26.9 jetpack

See also <https://github.com/android/codelab-android-compose>

26.9.1 material design

```
MaterialTheme(
    colorScheme = MyAppsColorScheme,
    typography = MyAppsTypography,
    shapes = MyAppsShapes,
) {
}
```

26.9.2 state

```
// var selectedAnswer: MutableState<Answer?> = mutableStateOf(null)
var selectedAnswer: MutableState<Answer?> = remember { mutableStateOf(null) }
var selectedAnswer: MutableState<Answer?> = rememberSaveable { mutableStateOf(null) }

// note that we use the member ``value`` here
isSelected = (selectedAnswer.value == answer)
```

```
var selectedAnswer: Answer? by rememberSaveable { mutableStateOf(null) }

isSelected = (selectedAnswer == answer)
```

```
val mutableState = remember { mutableStateOf(default) }

var value by remember { mutableStateOf(default) }

val (value, setValue) = remember { mutableStateOf(default) }
```

26.9.3 bom

<https://developer.android.com/jetpack/compose/bom/bom-mapping>

26.9.4 Arrangement

<https://vitor-ramos.medium.com/understand-arrangement-and-alignment-in-jetpack-compose-7633f2ed5b39>

- `Arrangement.Start`, `Arrangement.Top`
- `Arrangement.End`, `Arrangement.Bottom`
- `Arrangement.Center`
- `Arrangement.SpaceBetween`
- `Arrangement.SpaceAround`
- `Arrangement.SpaceEvenly`
- `Arrangement.spacedBy(8.dp)`

26.9.5 Alignment

- `Alignment.Start`
- `Alignment.End`
- `Alignment.Center`

26.9.6 Surface

```
Surface(  
    modifier = Modifier.fillMaxSize(),  
    color = MaterialTheme.colorscheme.background  
)  
  
Surface(  
    color = MaterialTheme.colorscheme.primary  
)  
  
Surface(  
    color = MaterialTheme.colorscheme.surface,  
    shape = RoundedCornerShape(8.dp),  
    border = BorderStroke(2.dp, MaterialTheme.colorscheme.surfaceVariant),  
    shadowElevation = 8.dp,  
    tonalElevation = 8.dp,  
)
```

26.9.7 Preview

```
@Preview(showBackground=true, showSystemUi=true, name="My name")

// dark mode
@Preview(
    showBackground = true,
    widthDp = 320,
    uiMode = UI_MODE_NIGHT_YES,
    name = "GreetingPreviewDark"
)
```

26.9.8 Column

```
Column(
    verticalArrangement = Arrangement.Center,
    horizontalAlignment = Alignment.CenterHorizontally,
    modifier = modifier.padding(8.dp)
)
```

26.9.9 LazyColumn

```
import androidx.compose.foundation.lazy.LazyColumn
import androidx.compose.foundation.lazy.items

@Composable
private fun Greetings(
    modifier: Modifier = Modifier,
    names: List<String> = List(1000) { "$it" }
) {
    LazyColumn(modifier = modifier.padding(vertical = 4.dp)) {
        items(items = names) { name ->
            Greeting(name = name)
        }
    }
}
```

26.9.10 Row

See [https://developer.android.com/reference/kotlin/androidx/compose/foundation/layout/package-summary#Row\(androidx.compose.ui.Modifier\)](https://developer.android.com/reference/kotlin/androidx/compose/foundation/layout/package-summary#Row(androidx.compose.ui.Modifier))

```
Row(modifier = Modifier.padding(all = 8.dp))

Row(
    modifier = Modifier.fillMaxSize(),
    horizontalArrangement = Arrangement.Center,
    verticalAlignment = Alignment.Top
)
```

(continues on next page)

(continued from previous page)

```
) {  
}  
  
Row(modifier = Modifier.padding(all = 8.dp)) {  
    Column(modifier = Modifier.weight(1f)) {  
        Text(text = "Hello")  
        Text(text = "World")  
    }  
  
    ElevatedButton(  
        onClick = {}  
    ) {  
        Text("Show more")  
    }  
}  
  
Row(  
    modifier = Modifier  
        .fillMaxSize()  
        .border(width = 2.dp, color = Color.Green)  
) {
```

26.9.11 LazyRow

```
LazyRow(  
    horizontalArrangement = Arrangement.spacedBy(8.dp)  
)  
  
@Composable  
fun AlignYourBodyRow(  
    modifier: Modifier = Modifier  
) {  
    LazyRow(  
        horizontalArrangement = Arrangement.spacedBy(8.dp),  
        contentPadding = PaddingValues(horizontal = 16.dp),  
        modifier = modifier  
    ) {  
        items(alignYourBodyData) { item ->  
            AlignYourBodyElement(item.drawable, item.text)  
        }  
    }  
}
```

26.9.12 Spacer

```
// within a row
Spacer(modifier = Modifier.width(8.dp))

// within a column
Spacer(modifier = Modifier.height(4.dp))

Spacer(Modifier.height(16.dp))

Spacer(
    modifier = Modifier
        .matchParentSize()
        .background(color = Color.Gray.copy(alpha = .7f))
)
```

26.9.13 Text

```
Text(
    text = "Happy Birthday Sam!",
    fontSize = 100.sp,
    lineHeight = 116.sp
    textAlign = TextAlign.Center
)

Text(
    text = "From Emma",
    modifier = Modifier.padding(16.dp).align(alignment = Alignment.End)
)

Text(text = name, style = MaterialTheme.typography.headlineMedium)

Text(
    text = name,
    style = MaterialTheme.typography.headlineMedium.copy(
        fontWeight = FontWeight.ExtraBold
    )
)

Text(text = stringResource(R.string.ab1_inversions))

Text(
    "Hello compose",
    Modifier.background(Color.Magenta)
        .size(200.dp, 30.dp)
        .padding(5.dp)
        .alpha(0.5f)
        .align(Alignment.BottomEnd)
        .clickable {}
)
```

26.9.14 Button

```
Button(  
    onclick = {}  
) {  
    Text("Show less")  
}
```

26.9.15 IconButton

```
IconButton(onClick = {...}) {  
    Icon(Icons.Filled.Close, contentDescription = "Close")  
}
```

26.9.16 Checkbox

```
Checkbox(  
    checked = true,  
    onCheckedChange = {it->}  
)
```

26.9.17 Modifier

See <https://developer.android.com/jetpack/compose/modifiers>

```
modifier = Modifier.fillMaxSize()  
  
modifier = modifier.padding(8.dp)  
  
modifier = modifier.padding(vertical = 4.dp, horizontal = 8.dp)  
  
modifier = Modifier.padding(start = 4.dp, end = 8.dp)  
  
modifier = modifier.padding(bottom = 4.dp)  
  
modifier = modifier.size(40.dp)  
  
modifier = modifier.weight(1f)  
  
modifier = modifier.heightIn(min = 56.dp)  
  
modifier = Modifier.align(Arrangement.CenterVertically)  
  
modifier.clip(CircleShape)  
  
modifier.clickable {  
    // some code  
}
```

(continues on next page)

(continued from previous page)

```
// For a function, it'd better to set the last argument as modifier
@Composable
fun MyApp(modifier: Modifier = Modifier) {}
```

26.9.18 Box

```
Box(
    modifier = Modifier.fillMaxSize(),
    contentAlignment = Alignment.Center,
)

Box(
    modifier = Modifier
        .padding(all = 2.dp)
        .size(56.dp)
        .background(color = Color.Magenta)
) {
}
```

26.9.19 Scaffold

```
Scaffold(
    topBar = {
        InfoHeader(...)
    },
    bottomBar = {
        Button(
            onClick = {...}
        ) {
            Text(...)
        }
    }
) {
    //
}
```

26.9.20 TextField

See [https://developer.android.com/reference/kotlin/androidx/compose/material3/package-summary#TextField\(androidx.compose.ui.text.Text\)](https://developer.android.com/reference/kotlin/androidx/compose/material3/package-summary#TextField(androidx.compose.ui.text.Text))

```
var text by rememberSaveable(stateSaver = TextFieldValue.Saver) {
    mutableStateOf(TextFieldValue("example", TextRange(0, 7)))
}

TextField(
    value = text,
```

(continues on next page)

(continued from previous page)

```

        onChange = { text = it },
        label = { Text("Label") }
    )

    TextField(
        value = "",
        onChange = {},
        modifier = modifier
            .fillMaxWidth()
            .heightIn(min = 56.dp)
    )

```

26.9.21 remember

```

// Note we use val here with "remember"
val expanded = remember { mutableStateOf(false) }

expanded.value = !expanded.value

val extraPadding = if(expanded.value) 48.dp else 0.dp

```

```

// Note we use var here with "by remember"
var expanded by remember { mutableStateOf(false) }

expanded = !expanded

val extraPadding = if(expanded) 48.dp else 0.dp

```

```

var expanded by rememberSaveable { mutableStateOf(false) }
val extraPadding by animateDpAsState {
    if (expanded) 48.dp else 0.dp
}

```

26.9.22 Example 1

See <https://developer.android.com/codelabs/jetpack-compose-basics?continue=https%3A%2F%2Fdeveloper.android.com%2Fcourses%2Fcompose-basics>

Note how Surface is used.

```

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            BasicsCodelabTheme {
                MyApp(modifier = Modifier.fillMaxSize())
            }
        }
    }
}

```

(continues on next page)

(continued from previous page)

```

@Composable
fun MyApp(modifier: Modifier = Modifier) {
    Surface(
        modifier = modifier,
        color = MaterialTheme.colorScheme.background
    ) {
        Greeting("Android")
    }
}

@Composable
fun Greeting(name: String, modifier: Modifier = Modifier) {
    Surface(color = MaterialTheme.colorScheme.primary) {
        Text(
            text = "Hello $name!",
            modifier = modifier.padding(24.dp)
        )
    }
}

@Preview(showBackground = true)
@Composable
fun GreetingPreview() {
    BasicsCodelabTheme {
        MyApp()
    }
}

```

26.9.23 Example 2

See <https://developer.android.com/codelabs/jetpack-compose-basics?continue=https%3A%2F%2Fdeveloper.android.com%2Fcourses%2Fcompose-basics>

Note that we pass a lambda function `onContinueClicked` to change the state.

```

@Composable
fun MyApp(modifier: Modifier = Modifier) {

    var shouldShowOnboarding by remember { mutableStateOf(true) }

    Surface(modifier) {
        if (shouldShowOnboarding) {
            OnboardingScreen(onContinueClicked = { shouldShowOnboarding = false })
        } else {
            Greetings()
        }
    }
}

@Composable
fun OnboardingScreen(

```

(continues on next page)

(continued from previous page)

```

onContinueClicked: () -> Unit,
modifier: Modifier = Modifier
) {

    Column(
        modifier = modifier.fillMaxSize(),
        verticalArrangement = Arrangement.Center,
        horizontalAlignment = Alignment.CenterHorizontally
    ) {
        Text("Welcome to the Basics Codelab!")
        Button(
            modifier = Modifier
                .padding(vertical = 24.dp),
            onClick = onContinueClicked
        ) {
            Text("Continue")
        }
    }
}

@Preview(showBackground = true, widthDp = 320, heightDp = 320)
@Composable
fun OnboardingPreview() {
    BasicsCodelabTheme {
        OnboardingScreen(onContinueClicked = {}) // Do nothing on click.
    }
}

```

26.9.24 Example 3

See <https://developer.android.com/codelabs/jetpack-compose-basics?continue=https%3A%2F%2Fdeveloper.android.com%2Fcourses%2Fcompose-basics>

Change the colors.

```

// Color.kt

val Navy = Color(0xFF073042)
val Blue = Color(0xFF4285F4)
val LightBlue = Color(0xFFD7E9FE)
val Chartreuse = Color(0xFFE9F7CF)

```

```

// Theme.kt

private val LightColorScheme = lightColorScheme(
    surface = Blue,
    onSurface = Color.White,
    primary = LightBlue,
    onPrimary = Navy
)

```

(continues on next page)

(continued from previous page)

```
private val DarkColorScheme = darkColorScheme(
    surface = Blue,
    onSurface = Navy,
    primary = Navy,
    onPrimary = Chartreuse
)
```

26.9.25 Example 4

```
@Composable
fun FavoriteCollectionCard(
    @DrawableRes drawable: Int,
    @StringRes text: Int,
    modifier: Modifier = Modifier
) {
    Surface(
        shape = MaterialTheme.shapes.medium,
        color = MaterialTheme.colorScheme.surfaceVariant,
        modifier = modifier
    ) {
        Row(
            verticalAlignment = Alignment.CenterVertically,
            modifier = Modifier.width(255.dp)
        ) {
            Image(
                painter = painterResource(drawable),
                contentDescription = null,
                contentScale = ContentScale.Crop,
                modifier = Modifier.size(80.dp)
            )
            Text(
                text = stringResource(text),
                style = MaterialTheme.typography.titleMedium,
                modifier = Modifier.padding(horizontal = 16.dp)
            )
        }
    }
}

@Preview(showBackground = true, backgroundColor = 0xFFFFF0EE)
@Composable
fun FavoriteCollectionCardPreview() {
    MySootheTheme {
        FavoriteCollectionCard(
            text = R.string.fc2_nature_meditations,
            drawable = R.drawable.fc2_nature_meditations,
            modifier = Modifier.padding(8.dp)
        )
    }
}
```

26.9.26 Example 5

scroll

```
@Composable
fun HomeScreen(modifier: Modifier = Modifier) {
    Column(
        modifier
            .verticalScroll(rememberScrollState())
    ) {
        Spacer(Modifier.height(16.dp))
        SearchBar(Modifier.padding(horizontal = 16.dp))
        HomeSection(title = R.string.align_your_body) {
            AlignYourBodyRow()
        }
        HomeSection(title = R.string.favorite_collections) {
            FavoriteCollectionsGrid()
        }
        Spacer(Modifier.height(16.dp))
    }
}
```

26.9.27 Example 6

```
@Composable
fun SingleChoiceQuestion(answers: List<Answer>) {
    Column {
        if(answers.isEmpty()) {
            Text("...")
        } else {
            answers.forEach {answer ->
                Text("...")
            }
        }
    }
}
```

26.9.28 Example 7

```
@Composable
fun MyAppTopAppBar(topAppBarText: String, onBackPressed: () -> Unit) {
    TopAppBar(
        title = {
            Text(
                text = topAppBarText,
                textAlign = TextAlign.Center,
                modifier = Modifier
                    .fillMaxSize()
                    .wrapContentSize(Alignment.Center)
            )
        }
    )
}
```

(continues on next page)

(continued from previous page)

```
    },
    navigationIcon = {
        IconButton(onClick = onBackPressed) {
            Icon(
                Icons.Filled.ArrowBack,
                contentDescription = localizedString
            )
        }
    },
    // ...
)
}
```

26.10 Samples

- <https://developer.android.com/courses/android-basics-compose/course>
 - <https://developer.android.com/courses/android-basics-compose/unit-2>
- <https://developer.android.com/modern-android-development>
- <https://developer.android.com/courses/pathways/android-basics-compose-unit-1-pathway-3>
- <https://developer.android.com/courses/pathways/jetpack-compose-for-android-developers-1>
- <https://github.com/android/compose-samples>
- <https://developer.android.com/courses/android-basics-compose/course>

27.1 Install

```
git clone --depth 1 https://github.com/qemu/qemu
cd qemu
```

27.1.1 qemu-arm

Refer to <https://github.com/Tencent/ncnn/blob/master/.github/workflows/linux-arm-cpu-gcc.yml>

```
# To generate only qemu-arm
./configure --prefix=/ceph-fj/fangjun/software/qemu/ --target-list=arm-linux-user,
↪ aarch64-linux-user --disable-system
make -j10

# It generates the executable: ./build/qemu-arm and ./build/qemu-aarch64

# If we run `make install`, it will generate
#
# /ceph-fj/fangjun/software/qemu/bin/qemu-arm
# /ceph-fj/fangjun/software/qemu/bin/qemu-aarch64
#
# Add it to PATH and use it!
```

To run it, we have to download some cross-compile toolchain, e.g,

Go to <https://developer.arm.com/tools-and-software/open-source-software/developer-tools/gnu-toolchain/gnu-a/downloads/8-3-2019-0> to download the toolchain.

```
mkdir /ceph-fj/fangjun/software
cd /ceph-fj/fangjun/software
tar xvf /path/to/gcc-arm-8.3-2019.03-x86_64-arm-linux-gnueabi.tar.xz

export PATH=/ceph-fj/fangjun/software/gcc-arm-8.3-2019.03-x86_64-arm-linux-gnueabi/bin:
↪ $PATH
```

If we have built an executable using the above toolchain, we can run it with qemu-arm:

```
./build/qemu-arm /path/to/sherpa-ncnn
```

It throws the following error:

```
qemu-arm: Unable to reserve 0xffff0000 bytes of virtual address space at
0x1000 (Success) for use as guest address space (check your virtual memory
ulimit setting, min_mmap_addr or reserve less using -R option)
```

We can use

```
./build/qemu-arm -B 0x100000000 /path/to/sherpa-ncnn
```

which throws the following new error:

```
(py38) kuangfangjun:qemu$ find /ceph-fj/fangjun/software/gcc-arm-8.3-2019.03-x86_64-arm-
↳ linux-gnueabi/lib/ -name "ld-linux-armhf.so.3"
/ceph-fj/fangjun/software/gcc-arm-8.3-2019.03-x86_64-arm-linux-gnueabi/lib/arm-linux-
↳ gnueabi/libc/lib/ld-linux-armhf.so.3
(py38) kuangfangjun:qemu$ export QEMU_LD_PREFIX=/ceph-fj/fangjun/software/gcc-arm-8.3-
↳ 2019.03-x86_64-arm-linux-gnueabi/libc
```

Now we can restart:

```
./build/qemu-arm -B 0x100000000 /path/to/sherpa-ncnn
```

```
export PATH=/ceph-fj/fangjun/software/qemu/bin:$PATH
export QEMU_LD_PREFIX=/ceph-fj/fangjun/software/gcc-arm-8.3-2019.03-x86_64-arm-linux-
↳ gnueabi/libc/arm-linux-gnueabi/libc

qemu-arm -B 0x100000000 ./build-arm-linux-gnueabi/bin/sherpa-ncnn \
./sherpa-ncnn-2022-09-05/tokens.txt \
./sherpa-ncnn-2022-09-05/bar/encoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \
./sherpa-ncnn-2022-09-05/bar/encoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \
./sherpa-ncnn-2022-09-05/bar/decoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \
./sherpa-ncnn-2022-09-05/bar/decoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \
./sherpa-ncnn-2022-09-05/bar/joiner_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \
./sherpa-ncnn-2022-09-05/bar/joiner_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \
./test_wavs/1221-135766-0002.wav
```

27.1.2 qemu-aarch64

```
wget https://releases.linaro.org/components/toolchain/binaries/latest-7/aarch64-linux-
↳ gnu/gcc-linaro-7.5.0-2019.12-x86_64_aarch64-linux-gnu.tar.xz
cd /ceph-fj/fangjun/software
tar xvf /path/to/gcc-linaro-7.5.0-2019.12-x86_64_aarch64-linux-gnu.tar.xz
```

```
export PATH=/ceph-fj/fangjun/software/qemu/bin:$PATH
export QEMU_LD_PREFIX=/ceph-fj/fangjun/software/gcc-linaro-7.5.0-2019.12-x86_64_aarch64-
↳ linux-gnu/aarch64-linux-gnu/libc

qemu-aarch64 -B 0x100000000 ./build-aarch64-linux-gnu/bin/sherpa-ncnn \
./sherpa-ncnn-2022-09-05/tokens.txt \
./sherpa-ncnn-2022-09-05/bar/encoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \
./sherpa-ncnn-2022-09-05/bar/encoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \
./sherpa-ncnn-2022-09-05/bar/decoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \
```

(continues on next page)

(continued from previous page)

```
./sherpa-ncnn-2022-09-05/bar/decoder_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \  
./sherpa-ncnn-2022-09-05/bar/joiner_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.param \  
./sherpa-ncnn-2022-09-05/bar/joiner_jit_trace-v2-iter-468000-avg-16-pnnx.ncnn.bin \  
./test_wavs/1221-135766-0002.wav
```


28.1 Basics

28.1.1 Extract part of a wave

```
# offset 2 seconds, length 0.195 seconds  
sox in.mp3 out.mp3 trim 2 0.195
```

28.1.2 Fix broken waves

```
sox --ignore-length corrupted.wav fixed.wav
```

28.1.3 pad

```
# add 2 seconds of silence at the begining and 3 seconds of silence  
# at the end.  
sox in.wav out.wav pad 2 3
```

28.1.4 raw

```
sox -t raw -r 16000 -e signed -b 16 -c 1 test.raw -r 48000 out.wav
```

28.1.5 Generate

```
sox -r 16000 -n -b 16 -c 1 a.wav synth 10 sin 1000
```

28.1.6 remix

Split multi-channel wave to single channels.

```
sox stereo_inputfile.wav leftchannel.wav remix 1
sox stereo_inputfile.wav rightchannel.wav remix 2
```

28.2 base64

```
import base64
encode_string = base64.b64encode(open("audio.wav", "rb").read())
wav_file = open("temp.wav", "wb")
decode_string = base64.b64decode(encode_string)
wav_file.write(decode_string)
```

29.1 Install

Refer to <https://mnn-docs.readthedocs.io/en/latest/compile/engine.html>

```
./schema/generate.sh
mkdir build
cd build
cmake -DMNN_BUILD_CONVERTER=ON -DMNN_SUPPORT_DEPRECATED_OP=OFF -DMNN_BUILD_TORCH=ON ..
make -j 10
```

Note: Use CPLUS_INCLUDE_PATH and C_INCLUDE_PATH to add additional paths for searching.

29.1.1 Python

See <https://mnn-docs.readthedocs.io/en/latest/compile/pymnn.html>

To build a python package, use:

```
cd pymnn/pip_package/
python3 ./build_deps.py
python3 setup.py bdist_wheel --version 2.1.1
# We can get the version from include/MNN/MNNDefine.h
# or we can give it an arbitrary version string.
pip install ./dist/MNN-2.1.1-cp38-cp38-linux_x86_64.whl
cd $HOME
python3 -c "import MNN; print(dir(MNN))"
```

29.2 Hello

Create a simple torchscript model:

Listing 1: ./code/hello/ex1.py

```
1 #!/usr/bin/env python3
2
3 import torch
```

(continues on next page)

(continued from previous page)

```

4
5
6 class Foo(torch.nn.Module):
7     def forward(self, x):
8         return torch.nn.functional.relu(x)
9
10
11 f = Foo()
12 x = torch.rand(2)
13 m = torch.jit.trace(f, x)
14 m.save("ex1.pt")
15 print(m.graph)

```

```
MNNConvert -f TORCH --modelFile ./ex1.pt --MNNModel ex1.mnn --bizCode MNN
```

It prints:

```

Start to Convert Other Model Format To MNN Model...
Start to Optimize the MNN Net...
inputTensors : [ x.1, ]
outputTensors: [ 2, ]
Converted Success!

```

We can use:

```
netron ex1.mnn --port 6006
```

to view it:

```
Serving 'ex1.mnn' at http://localhost:6006
```

To convert the model to json, use:

```
MNNDump2Json ./ex1.mnn ./ex1.json
```

which generates:

Listing 2: ./code/hello/ex1.json

```

1 { "bizCode": "MNN", "extraInfo":
2   { "version": "2.1.1" }
3   , "oplists":
4   [
5     { "main_type": "Input", "main":
6       { "dtype": "DT_FLOAT", "dformat": "NCHW" }
7       , "name": "x.1", "outputIndexes":
8       [ 0 ]
9       , "type": "Input", "defaultDimentionFormat": "NHWC" }
10    ,
11    { "inputIndexes":
12      [ 0 ]
13      , "main_type": "Relu", "main":
14      { "slope": 0.0 }

```

(continues on next page)

(continued from previous page)

```

15 , "name": "2", "outputIndexes":
16 [ 1 ]
17 , "type": "ReLU", "defaultDimentionFormat": "NCHW" }
18 ]
19 , "outputName":
20 [ "2" ]
21 , "preferForwardType": "CPU", "sourceType": "TORCH", "tensorName":
22 [ "x.1", "2" ]
23 , "tensorNumber": 0, "usage": "INFERENCE", "mnn_uuid": "d24b26b1-14c8-42cb-9ce1-
  ↪ 2a9b867714e9" }

```

To convert ex1.json back to a .mnn file, use:

```
MNNRevert2Buffer ex1.json ex1.mnn
```

To show the information of the model:

```
MNNConvert --framework MNN --modelFile ./ex1.mnn --info
```

It prints:

```

Model default dimensionFormat is NCHW
Model Inputs:
[ x.1 ]: dimensionFormat: NCHW, size: [ ], type is float
Model Outputs:
[ 2 ]
Model Version: 2.1.1

```

To run it with MNN in Python:

Listing 3: ./code/hello/test-ex1-mnn.py

```

1  #!/usr/bin/env python3
2
3  import numpy as np
4  import MNN
5  import torch
6
7
8  def main():
9      interpreter = MNN.Interpreter("ex1.mnn")
10
11      config = {}
12      config["precision"] = "low" # low, high, normal
13      config["backend"] = "CPU"
14      # config["thread"] = 1
15      session = interpreter.createSession(config)
16
17      # a = torch.tensor([1, -2.5, 3.2], dtype=torch.float32)
18      # b = a.numpy()
19
20      input_tensor = interpreter.getSessionInput(session)
21      interpreter.resizeTensor(input_tensor, (3,))

```

(continues on next page)

(continued from previous page)

```
22 interpreter.resizeSession(session)
23 print("input_tensor.getShape()", input_tensor.getShape())
24
25 input_data = MNN.expr.const(
26     np.array([1, -2.5, 3.2], dtype=np.float32),
27     (3,),
28     MNN.expr.NCHW,
29     MNN.expr.float,
30 )
31 input_tensor.copyFrom(MNN.Tensor(input_data))
32 print("input_tensor.getNumpyData()", input_tensor.getNumpyData())
33
34 interpreter.runSession(session)
35
36 output_tensor = interpreter.getSessionOutput(session)
37 print("output_tensor.getNumpyData()", output_tensor.getNumpyData())
38
39 output_data = MNN.Tensor(
40     output_tensor.getShape(), MNN.Halide_Type_Float, MNN.Tensor_DimensionType_Caffe
41 )
42 output_tensor.copyToHostTensor(output_data)
43 print("output_data.getNumpyData()", output_data.getNumpyData())
44
45
46 if __name__ == "__main__":
47     main()
```

It prints:

```
input_tensor.getShape() (3,)
input_tensor.getNumpyData() [ 1. -2.5  3.2]
output_tensor.getNumpyData() [1.  0.  3.2]
output_data.getNumpyData() [1.  0.  3.2]
```


30.1 References

- <https://www.intel.com/content/www/us/en/docs/intrinsics-guide/index.html>
All APIs for intrinsics with examples
- 15-418/15-618: Parallel Computer Architecture and Programming, Spring 2018: Schedule
<https://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15418-s18/www/schedule.html>
- How to Write Fast Code 18-645 (CMU, ECE)
<https://users.ece.cmu.edu/~pueschel/teaching/18-645-CMU-spring08/course.html>
- <http://spcl.inf.ethz.ch/Teaching/2018-dphpc/lectures/lecture8-simd.pdf>

SSE:

- 128-bit
- `__m128`, `__m128d`
- `_mm_load_ps`, `_mm_add_pd`

AVX:

- 256-bit
- `__m256`, `__m256d`
- `_mm256_load_ps`, `_mm256_add_pd`

AVX512:

- 512-bit

30.2 Headers

- `mmintrin.h` MMX
- `xmmintrin.h` SSE
- `emmintrin.h`, SSE2
- `pmmintrin.h`, SSE3
- `tmmintrin.h`, SSSE3
- `smmintrin.h`, SSE4.1

- `nmmintrin.h`, SSE4.2
- `ammintrin.h`, SSE4A
- `wmmintrin.h`, AES
- `immintrin.h`, AVX

30.3 SSE

The header file is `xmmintrin.h`, it is in `/usr/lib/gcc/x86_64-linux-gnu/7/include`. There is also a file `mmmintrin.h`.

30.4 avx

SSE4 data types:

- `__m128`, 4 floats
- `__m128d`, 2 doubles
- `__m128i`, it depends, can be 16 8-bit, 8 16-bit, 4 32-bit, 2 64-bit

AVX2 data types:

- `__m256`, 8 floats
- `__m256d`, 4 doubles
- `__m256i`, 32 8-bit, 16 16-bit, 8 32-bit, 4 64-bit

Listing 1: `./code/avx/main.cc`

```

1  #include <cassert>
2  #include <immintrin.h>
3
4  // ps means packed single precision
5  static void TestLoadStore() {
6      alignas(16) float a[4] = {1, 2, 3, 4};
7      alignas(16) float b[4];
8      __m128 f = _mm_load_ps(a);
9      // f = _mm_loadu_ps(a); // if a not aligned
10     _mm_store_ps(b, f);
11     // _mm_storeu_ps(b, f); // if b is not aligned
12     assert(b[0] == a[0]);
13     assert(b[1] == a[1]);
14     assert(b[2] == a[2]);
15     assert(b[3] == a[3]);
16
17     // set manually
18     f = _mm_set_ps(a[3], a[2], a[1], a[0]);
19     _mm_store_ps(b, f);
20     assert(b[0] == a[0]);
21     assert(b[1] == a[1]);
22     assert(b[2] == a[2]);

```

(continues on next page)

(continued from previous page)

```

23  assert(b[3] == a[3]);
24
25  // for double
26  alignas(32) double k[4] = {1, 2, 3, 4};
27  __m256d d = _mm256_load_pd(k);
28  // d = _mm256_loadu_pd(k); // if k is not aligned
29  alignas(32) double m[4];
30  _mm256_store_pd(m, d);
31  // _mm256_storeu_pd(m, d); // if m is not aligned
32  assert(m[0] == k[0]);
33  assert(m[1] == k[1]);
34  assert(m[2] == k[2]);
35  assert(m[3] == k[3]);
36
37  d = _mm256_set_pd(k[3], k[2], k[1], k[0]);
38  _mm256_store_pd(m, d);
39  assert(m[0] == k[0]);
40  assert(m[1] == k[1]);
41  assert(m[2] == k[2]);
42  assert(m[3] == k[3]);
43 }
44
45 static void TestLoadStore1() {
46     float a = 10;
47     float b[4];
48     __m128 f = _mm_load_ps1(&a);
49     _mm_store_ps(b, f);
50     assert(b[0] == a);
51     assert(b[1] == a);
52     assert(b[2] == a);
53     assert(b[3] == a);
54 }
55
56 static void TestAdd() {
57     float a[4] = {1, 2, 3, 4};
58     float b[4] = {10, 20, 30, 40};
59     __m128 f = _mm_load_ps(a);
60     __m128 g = _mm_load_ps(b);
61     __m128 h = _mm_add_ps(f, g);
62     float c[4];
63     _mm_store_ps(c, h);
64     assert(c[0] == a[0] + b[0]);
65     assert(c[1] == a[1] + b[1]);
66     assert(c[2] == a[2] + b[2]);
67     assert(c[3] == a[3] + b[3]);
68 }
69
70 static void AddIndex1(double *x, int32_t n) {
71     for (int32_t i = 0; i < n; ++i) {
72         x[i] = x[i] + i;
73     }
74 }

```

(continues on next page)

(continued from previous page)

```

75
76 // assume n % 4 == 0
77 static void AddIndex2(double *x, int32_t n) {
78     assert(n % 4 == 0);
79     __m256d index, x_vec;
80     for (int32_t i = 0; i < n; i += 4) {
81         x_vec = _mm256_load_pd(x + i);
82         // x_vec[0] = x[i]
83         // x_vec[1] = x[i+1]
84         // x_vec[2] = x[i+2]
85         // x_vec[3] = x[i+3]
86
87         index = _mm256_set_pd(i + 3, i + 2, i + 1, i);
88         // index[0] = i
89         // index[1] = i+1
90         // index[2] = i+2
91         // index[3] = i+3
92
93         x_vec = _mm256_add_pd(x_vec, index);
94         // x_vec[0] = x_vec[0] + index[0]
95         // x_vec[1] = x_vec[1] + index[1]
96         // x_vec[2] = x_vec[2] + index[2]
97         // x_vec[3] = x_vec[3] + index[3]
98
99         _mm256_store_pd(x + i, x_vec);
100        // (x+i)[0] = x_vec[0]
101        // (x+i)[1] = x_vec[1]
102        // (x+i)[2] = x_vec[2]
103        // (x+i)[3] = x_vec[3]
104    }
105 }
106
107 static void TestAddIndex() {
108     alignas(32) double a[64];
109     alignas(32) double b[64];
110     for (int32_t i = 0; i != 64; ++i) {
111         a[i] = b[i] = i;
112     }
113     AddIndex1(a, 64);
114     AddIndex2(b, 64);
115     for (int32_t i = 0; i != 64; ++i) {
116         assert(a[i] == b[i]);
117     }
118 }
119
120 int main() {
121     TestLoadStore();
122     TestLoadStore1();
123     TestAdd();
124     TestAddIndex();
125     return 0;
126 }

```

31.1 Install

```
git clone https://github.com/chriskohlhoff/asio/  
cd asio/asio  
./autogen.sh  
./configure --prefix=/ceph-fj/fangjun/software/asio  
make -j 10  
make install
```

It will create include and lib/pkgconfig/asio.pc inside /ceph-fj/fangjun/software/asio. Note that it is a header only library.

31.2 Hello

<https://think-async.com/Asio/asio-1.24.0/doc/asio/tutorial/tuttimer1.html>

Listing 1: ./code/Makefile

```
1 CXXFLAGS := -I /ceph-fj/fangjun/software/asio/include -pthread  
2  
3 srcs := $(shell find . -type f -name "*.cc" -printf "%f ")  
4 targets :=$(srcs:%.cc=%)  
5  
6 all: $(targets)  
7  
8 %: %.cc  
9     $(CXX) $(CXXFLAGS) -o $@ $<  
10  
11  
12 .PHONY: clean  
13 clean:  
14     $(RM) $(targets)
```

Listing 2: ./code/1-timer-synchronous.cc

```
1 #include "asio.hpp"  
2 #include <iostream>  
3
```

(continues on next page)

(continued from previous page)

```

4 int main() {
5     asio::io_context io;
6     std::cout << "sleep for 1 second\n";
7     asio::steady_timer t(io, asio::chrono::seconds(1));
8     t.wait();
9     std::cout << "hello world\n";
10    return 0;
11 }

```

Listing 3: ./code/2-timer-asynchronous.cc

```

1 #include "asio.hpp"
2 #include <iostream>
3
4 void print(const asio::error_code &) { std::cout << "done\n"; }
5
6 int main() {
7     asio::io_context io;
8     std::cout << "sleep for 1 seconds\n";
9     asio::steady_timer t(io, asio::chrono::seconds(1));
10
11     t.async_wait(&print);
12     t.async_wait(&print);
13     io.run();
14     return 0;
15 }

```

Listing 4: ./code/3-timer-async-bind-argument.cc

```

1 #include "asio.hpp"
2 #include <iostream>
3
4 void print(const asio::error_code &ec, asio::steady_timer *t, int *counter) {
5     if (*counter < 3) {
6         std::cout << *counter << "\n";
7         *counter += 1;
8         t->expires_at(t->expiry() + asio::chrono::seconds(1));
9         t->async_wait(
10             [t, counter](const asio::error_code &e) { print(e, t, counter); });
11     }
12 }
13
14 int main() {
15     asio::io_context io;
16     asio::steady_timer t(io, asio::chrono::seconds(1));
17     int counter = 0;
18
19     // use a lambda to pass extra parameters
20     t.async_wait(
21         [&t, &counter](const asio::error_code &ec) { print(ec, &t, &counter); });
22     io.run();
23     return 0;

```

(continues on next page)

(continued from previous page)

24 }

Listing 5: ./code/4-timer-async-member-function.cc

```

1  #include "asio.hpp"
2  #include <iostream>
3
4  class printer {
5  public:
6      printer(asio::io_context &io)
7          : t_(io, asio::chrono::seconds(1)), counter_(0) {
8          t_.async_wait([this](const asio::error_code &ec) { print(ec); });
9      }
10
11     void print(const asio::error_code &ec) {
12         if (counter_ < 6) {
13             std::cout << counter_ << "\n";
14             counter_ += 1;
15             t_.async_wait([this](const asio::error_code &e) { this->print(e); });
16         }
17     }
18
19 private:
20     asio::steady_timer t_;
21     int counter_;
22 };
23 int main() {
24     asio::io_context io;
25     printer p(io);
26     io.run();
27     return 0;
28 }

```

Listing 6: ./code/5-timer-async-thread-non-synchronization.cc

```

1  #include "asio.hpp"
2  #include <iostream>
3  #include <thread>
4
5  class printer {
6  public:
7      printer(asio::io_context &io)
8          : t1_(io, asio::chrono::seconds(1)), t2_(io, asio::chrono::seconds(1)) {
9          t1_.async_wait([this](const asio::error_code &e) { print1(e); });
10         t2_.async_wait([this](const asio::error_code &e) { print2(e); });
11     }
12
13     void print1(const asio::error_code &ec) {
14         if (counter_ < 8) {
15             std::cout << "print1 id: " << std::this_thread::get_id() << ", counter "
16                 << counter_ << "\n";
17             counter_ += 1;

```

(continues on next page)

(continued from previous page)

```

18     t1_.expires_at(t1_.expiry() + asio::chrono::seconds(1));
19     t1_.async_wait([this](const asio::error_code &e) { print1(e); });
20 }
21 }
22
23 void print2(const asio::error_code &ec) {
24     if (counter_ < 8) {
25         std::cout << "print2 id: " << std::this_thread::get_id() << ", counter "
26             << counter_ << "\n";
27         counter_ += 1;
28         t2_.expires_at(t2_.expiry() + asio::chrono::seconds(1));
29         t2_.async_wait([this](const asio::error_code &e) { print2(e); });
30     }
31 }
32
33 private:
34     asio::steady_timer t1_;
35     asio::steady_timer t2_;
36     int counter_ = 0;
37 };
38
39 int main() {
40     asio::io_context io;
41     printer p(io);
42     std::thread t([&io]() { io.run(); }); // NOTE: print1() and print2() can run
43                                         // either in t or in the main thread
44     io.run();
45     t.join();
46 }
47
48 #if 0
49 (py38) kuangfangjun:code$ ./5-timer-async-thread-non-synchronization
50 print1 id: print2 id: 140164266059584, counter 0
51 140164247860992, counter 1
52 print1 id: 140164266059584, counter 2
53 print2 id: 140164247860992, counter 3
54 print1 id: 140164266059584, counter 4
55 print2 id: 140164247860992, counter 5
56 print1 id: 140164266059584, counter 6
57 print2 id: 140164247860992, counter 7
58 (py38) kuangfangjun:code$ ./5-timer-async-thread-non-synchronization
59 print2 id: print1 id: 140191018206976, counter 140191036405568, counter 0
60 0
61 print1 id: 140191018206976, counter print2 id: 2
62 140191036405568, counter 3
63 print1 id: 140191018206976, counter 4
64 print2 id: 140191036405568, counter 5
65 print1 id: 140191018206976, counter 6
66 print2 id: 140191036405568, counter 7
67 #endif

```


Listing 7: ./code/6-timer-async-thread-with-synchronization.cc

```

1  #include "asio.hpp"
2  #include <iostream>
3  #include <thread>
4
5  class printer {
6  public:
7      printer(asio::io_context &io)
8          : strand_(asio::make_strand(io)), t1_(io, asio::chrono::seconds(1)),
9            t2_(io, asio::chrono::seconds(1)) {
10         t1_.async_wait(asio::bind_executor(
11             strand_, [this](const asio::error_code &e) { print1(e); }));
12         t2_.async_wait(asio::bind_executor(
13             strand_, [this](const asio::error_code &e) { print2(e); }));
14     }
15
16     void print1(const asio::error_code &ec) {
17         if (counter_ < 8) {
18             std::cout << "print1 id: " << std::this_thread::get_id() << ", counter "
19                 << counter_ << "\n";
20             counter_ += 1;
21             t1_.expires_at(t1_.expiry() + asio::chrono::seconds(1));
22             t1_.async_wait([this](const asio::error_code &e) { print1(e); });
23         }
24     }
25
26     void print2(const asio::error_code &ec) {
27         if (counter_ < 8) {
28             std::cout << "print2 id: " << std::this_thread::get_id() << ", counter "
29                 << counter_ << "\n";
30             counter_ += 1;
31             t2_.expires_at(t2_.expiry() + asio::chrono::seconds(1));
32             t2_.async_wait([this](const asio::error_code &e) { print2(e); });
33         }
34     }
35
36 private:
37     asio::strand<asio::io_context::executor_type> strand_;
38     asio::steady_timer t1_;
39     asio::steady_timer t2_;
40     int counter_ = 0;
41 };
42
43 int main() {
44     asio::io_context io;
45     printer p(io);
46     std::thread t([&io]() { io.run(); }); // NOTE: print1() and print2() can run
47                                         // either in t or in the main thread
48     io.run();
49     t.join();
50 }

```

(continues on next page)

(continued from previous page)

```

51 #if 0
52 (py38) kuangfangjun:code$ ./6-timer-async-thread-with-synchronization
53 print1 id: 140648598263616, counter 0
54 print2 id: 140648598263616, counter 1
55 print1 id: 140648580065024, counter 2
56 print2 id: 140648598263616, counter 3
57 print1 id: 140648580065024, counter 4
58 print2 id: 140648598263616, counter 5
59 print1 id: 140648580065024, counter 6
60 print2 id: 140648598263616, counter 7
61 (py38) kuangfangjun:code$ ./6-timer-async-thread-with-synchronization
62 print1 id: 139671509145408, counter 0
63 print2 id: 139671509145408, counter 1
64 print1 id: 139671490946816, counter 2
65 print2 id: 139671509145408, counter 3
66 print1 id: 139671490946816, counter 4
67 print2 id: 139671509145408, counter 5
68 print1 id: 139671490946816, counter 6
69 print2 id: 139671509145408, counter 7
70 #endif

```

Listing 8: ./code/7-daytime-sync-client.cc

```

1  #include "asio.hpp"
2  #include <array>
3  #include <iostream>
4
5  using asio::ip::tcp;
6  int main(int argc, char *argv[]) {
7      try {
8          if (argc != 2) {
9              std::cerr << "Usage: client <host>" << std::endl;
10             return 1;
11         }
12
13         asio::io_context io;
14         tcp::resolver resolver(io);
15
16         tcp::resolver::results_type endpoints =
17             resolver.resolve(argv[1], "daytime");
18
19         tcp::socket socket(io);
20         asio::connect(socket, endpoints);
21
22         for (;;) {
23             std::array<char, 128> buf;
24             asio::error_code ec;
25             size_t len = socket.read_some(asio::buffer(buf), ec);
26             if (ec == asio::error::eof) {
27                 break;
28             } else if (ec) {
29                 throw asio::system_error(ec);

```

(continues on next page)

(continued from previous page)

```

30     }
31     std::cout.write(buf.data(), len);
32 }
33 } catch (std::exception &e) {
34     std::cerr << "exception: " << e.what() << "\n";
35 }
36 return 0;
37 }

```

Listing 9: ./code/8-daytime-sync-server.cc

```

1  #include "asio.hpp"
2  #include <ctime>
3  #include <iostream>
4  #include <string>
5
6  using asio::ip::tcp;
7  std::string make_daytime_string() {
8      using namespace std;
9      time_t now = time(0);
10     return ctime(&now);
11 }
12
13 int main() {
14     try {
15         asio::io_context io;
16         // need sudo permission to bind port 13
17         tcp::acceptor acceptor(io, tcp::endpoint(tcp::v4(), 13));
18         std::cout << "started\n";
19         for (;;) {
20             tcp::socket socket(io);
21             acceptor.accept(socket);
22
23             auto message = make_daytime_string();
24             asio::error_code ec;
25             asio::write(socket, asio::buffer(message), ec);
26         }
27     } catch (std::exception &e) {
28         std::cout << "Exception: " << e.what() << "\n";
29     }
30     return 0;
31 }

```

Listing 10: ./code/9-daytime-async-server.cc

```

1  #include "asio.hpp"
2  #include <ctime>
3  #include <iostream>
4  #include <memory>
5
6  using asio::ip::tcp;
7

```

(continues on next page)

(continued from previous page)

```

8  std::string make_daytime_string() {
9      using namespace std;
10     time_t now = time(0);
11     return ctime(&now);
12 }
13
14 class tcp_connection : public std::enable_shared_from_this<tcp_connection> {
15 public:
16     using pointer = std::shared_ptr<tcp_connection>;
17
18     static pointer create(asio::io_context &io) {
19         return pointer(new tcp_connection(io));
20         // return std::make_shared<tcp_connection>(io);
21     }
22
23     tcp::socket &socket() { return socket_; }
24
25     void start() {
26         message_ = make_daytime_string();
27         auto p = shared_from_this();
28         asio::async_write(
29             socket_, asio::buffer(message_),
30             [p](const asio::error_code &ec, size_t bytes_transferred) {
31                 p->handle_write(ec, bytes_transferred);
32             });
33     }
34
35 private:
36     tcp_connection(asio::io_context &io) : socket_(io) {}
37
38     void handle_write(const asio::error_code & /*ec*/,
39                     size_t /*bytes_transferred*/) {}
40
41     tcp::socket socket_;
42     std::string message_;
43 };
44
45 class tcp_server {
46 public:
47     tcp_server(asio::io_context &io)
48         : io_(io), acceptor_(io, tcp::endpoint(tcp::v4(), 13)) {
49         start_accept();
50     }
51
52 private:
53     void start_accept() {
54         tcp_connection::pointer new_connection = tcp_connection::create(io_);
55         acceptor_.async_accept(new_connection->socket(),
56                               [this, new_connection](const asio::error_code &ec) {
57                                   this->handle_accept(new_connection, ec);
58                               });
59     }

```

(continues on next page)

(continued from previous page)

```
60 void handle_accept(tcp_connection::pointer new_connection,  
61                    const asio::error_code &ec) {  
62     if (!ec) {  
63         new_connection->start();  
64     }  
65     start_accept();  
66 }  
67  
68 asio::io_context io_;  
69 tcp::acceptor acceptor_;  
70 };  
71  
72 int main() {  
73     try {  
74         asio::io_context io;  
75         tcp_server server(io);  
76         io.run();  
77     } catch (std::exception &e) {  
78         std::cout << "Exception: " << e.what();  
79     }  
80     return 0;  
81 }
```


WEBSOCKETPP

32.1 Install

```
git clone https://github.com/zaphoyd/websocketpp
cd websocketpp
mkdir build
cmake -DCMAKE_INSTALL_PREFIX=/ceph-fj/fangjun/software/websocketpp ..
make -j install
```

32.2 hello

Listing 1: ./code/hello/Makefile

```
1 CXXFLAGS := -DASIO_STANDALONE
2 CXXFLAGS += -I/ceph-fj/fangjun/software/asio/include
3 CXXFLAGS += -I/ceph-fj/fangjun/software/websocketpp/include
4 CXXFLAGS += -std=c++14
5 LDFLAGS += -pthread
6
7 main: main.o uri.o
8     $(CXX) -o $@ $^ $(LDFLAGS)
9
10 %.o: %.cc
11     $(CXX) -c $(CXXFLAGS) -o $@ $<
12
13 clean:
14     $(RM) *.o main
```

Listing 2: ./code/hello/main.cc

```
1 #include "websocketpp/client.hpp"
2 #include "websocketpp/config/asio_no_tls_client.hpp"
3 #include <assert.h>
4 #include <iostream>
5 #include <memory>
6 #include <string>
7 #include <thread>
8
```

(continues on next page)

(continued from previous page)

```

9 using client = websocketpp::client<websocketpp::config::asio_client>;
10
11 class connection_metadata {
12 public:
13     using ptr = std::shared_ptr<connection_metadata>;
14     connection_metadata(int32_t id, websocketpp::connection_hdl hdl,
15                         std::string uri)
16         : m_id(id), m_hdl(hdl), m_status("Connecting"), m_uri(uri),
17           m_server("N/A") {}
18     void on_open(client *c, websocketpp::connection_hdl hdl) {
19         m_status = "Open";
20         client::connection_ptr con = c->get_con_from_hdl(hdl);
21         m_server = con->get_response_header("Server");
22
23         // the type of con->get_response() is defined in
24         // websocketpp/http/response.hpp
25         assert(con->get_response().get_version() == "HTTP/1.1");
26     #if 0
27         // header_list is a std::map<std::string, std::string>
28         auto header_list = con->get_response().get_headers();
29         for (const auto &it : header_list) {
30             std::cout << it.first << ": " << it.second << "\n";
31         }
32         /*
33         Connection: upgrade
34         Date: Wed, 05 Oct 2022 11:33:39 GMT
35         Sec-WebSocket-Accept: QvLQVaiZn5YWerU3s15SuDWypjo=
36         Server: nginx
37         Upgrade: websocket
38         */
39     #endif
40         assert(con->get_response().get_body() == "");
41     }
42
43     void on_fail(client *c, websocketpp::connection_hdl hdl) {
44         m_status = "Failed";
45         client::connection_ptr con = c->get_con_from_hdl(hdl);
46         m_server = con->get_response_header("Server");
47         m_error_reason = con->get_ec().message();
48     }
49
50     void on_close(client *c, websocketpp::connection_hdl hdl) {
51         m_status = "Closed";
52         client::connection_ptr con = c->get_con_from_hdl(hdl);
53         std::ostringstream os;
54         os << "close code: " << con->get_remote_close_code() << " ("
55           << websocketpp::close::status::get_string(con->get_remote_close_code())
56           << "), close reason: " << con->get_remote_close_reason();
57         m_error_reason = os.str();
58     }
59
60     websocketpp::connection_hdl get_hdl() const { return m_hdl; }

```

(continues on next page)

(continued from previous page)

```

61  int32_t get_id() const { return m_id; }
62  const std::string &get_status() const { return m_status; }
63
64  friend std::ostream &operator<<(std::ostream &out,
65                                  const connection_metadata &data);
66
67 private:
68  int32_t m_id;
69  websocketpp::connection_hdl m_hdl;
70  std::string m_status;
71  std::string m_uri;
72  std::string m_server;
73  std::string m_error_reason;
74 };
75
76 std::ostream &operator<<(std::ostream &out, const connection_metadata &data) {
77     out << "> URI: " << data.m_uri << "\n"
78     << "> Status: " << data.m_status << "> Remote Server: "
79     << (data.m_server.empty() ? "None Specified" : data.m_server) << "\n"
80     << "> Error/close reason: "
81     << (data.m_error_reason.empty() ? "N/A" : data.m_error_reason);
82     return out;
83 }
84
85 class websocket_endpoint {
86 public:
87     websocket_endpoint() : m_next_id(0) {
88         m_endpoint.clear_access_channels(websocketpp::log::alevel::all);
89         m_endpoint.clear_error_channels(websocketpp::log::elevel::all);
90
91         m_endpoint.init_asio();
92         m_endpoint.start_perpetual();
93
94         m_thread.reset(new std::thread(&client::run, &m_endpoint));
95     }
96
97     ~websocket_endpoint() {
98         m_endpoint.stop_perpetual();
99         for (auto it = m_connection_list.begin(); it != m_connection_list.end();
100             ++it) {
101             if (it->second->get_status() != "Open") {
102                 continue;
103             }
104
105             std::cout << "> Closing connection " << it->second->get_id() << std::endl;
106             websocketpp::lib::error_code ec;
107             m_endpoint.close(it->second->get_hdl(),
108                             websocketpp::close::status::going_away, "", ec);
109             if (ec) {
110                 std::cout << "> Error closing connection " << it->second->get_id()
111                 << ": " << ec.message() << std::endl;
112             }

```

(continues on next page)

(continued from previous page)

```

113     }
114     m_thread->join();
115 }
116
117 int32_t connect(std::string uri) {
118     websocketpp::lib::error_code ec;
119     client::connection_ptr con = m_endpoint.get_connection(uri, ec);
120     if (ec) {
121         std::cout << "> Connect initialization error: " << ec.message()
122                 << std::endl;
123         return -1;
124     }
125     int32_t new_id = m_next_id++;
126     connection_metadata::ptr metadata_ptr(
127         new connection_metadata(new_id, con->get_handle(), uri));
128     m_connection_list[new_id] = metadata_ptr;
129
130     // see websocketpp/connection.hpp
131     con->set_open_handler(
132         [metadata_ptr = metadata_ptr, this](websocketpp::connection_hdl hdl) {
133             metadata_ptr->on_open(&m_endpoint, hdl);
134         });
135
136     // see websocketpp/connection.hpp
137     con->set_fail_handler(
138         [metadata_ptr = metadata_ptr, this](websocketpp::connection_hdl hdl) {
139             metadata_ptr->on_fail(&m_endpoint, hdl);
140         });
141
142     con->set_close_handler(
143         [metadata_ptr = metadata_ptr, this](websocketpp::connection_hdl hdl) {
144             metadata_ptr->on_close(&m_endpoint, hdl);
145         });
146
147     m_endpoint.connect(con);
148
149     return new_id;
150 }
151
152 connection_metadata::ptr get_metadata(int32_t id) const {
153     auto metadata_it = m_connection_list.find(id);
154     if (metadata_it == m_connection_list.end()) {
155         return connection_metadata::ptr();
156     } else {
157         return metadata_it->second;
158     }
159 }
160
161 void close(int32_t id, websocketpp::close::status::value code,
162           const std::string &reason = "") {
163     websocketpp::lib::error_code ec;
164     auto metadata_it = m_connection_list.find(id);

```

(continues on next page)

(continued from previous page)

```

165     if (metadata_it == m_connection_list.end()) {
166         std::cout << "> No connection found with id " << id << std::endl;
167         return;
168     }
169
170     m_endpoint.close(metadata_it->second->get_hdl(), code, reason, ec);
171     if (ec) {
172         std::cout << "> Error initiating close: " << ec.message() << std::endl;
173     }
174 }
175
176 private:
177     client m_endpoint;
178     std::shared_ptr<std::thread> m_thread;
179     std::map<int32_t, connection_metadata::ptr> m_connection_list;
180     int32_t m_next_id;
181 };
182
183 void test_uri();
184
185 int main() {
186     test_uri();
187     // return 0;
188
189     bool done = false;
190     std::string input;
191     websocket_endpoint endpoint;
192
193     while (true) {
194         std::cout << "Enter command: ";
195         std::getline(std::cin, input);
196         if (input == "quit") {
197             done = true;
198         } else if (input == "help") {
199             std::cout << "\nCommand List:\n"
200                 << "connect <ws uri>\n"
201                 << "show <connection_id>\n"
202                 << "close <connection_id> <close_code> <close_reason>\n"
203                 << "help: Display the help and exit\n"
204                 << "quit: Exit the program\n"
205                 << "\n";
206         } else if (input.substr(0, 7) == "connect") {
207             int32_t id = endpoint.connect(input.substr(8));
208             if (id != -1) {
209                 std::cout << "Created connection with id " << id << "\n";
210             }
211         } else if (input.substr(0, 5) == "close") {
212             std::stringstream ss(input);
213             std::string cmd;
214             int32_t id;
215             int32_t close_code = websocketpp::close::status::normal;
216             std::string reason;

```

(continues on next page)

(continued from previous page)

```

217     ss >> cmd >> id >> close_code;
218     std::getline(ss, reason);
219     endponit.close(id, close_code, reason);
220 } else if (input.substr(0, 4) == "show") {
221     int32_t id = atoi(input.substr(5).c_str());
222     connection_metadata::ptr metadata = endponit.get_metadata(id);
223     if (metadata) {
224         std::cout << *metadata << std::endl;
225     } else {
226         std::cout << "Unknown connection id " << id << std::endl;
227     }
228 } else {
229     std::cout << "Unrecognized command\n";
230 }
231 if (done)
232     break;
233 }
234 return 0;
235 }
236 // connect ws://websocket-echo.com
237 // close 0 1001 example message

```

Listing 3: ./code/hello/uri.cc

```

1  #include "websocketpp/uri.hpp"
2  #include <assert.h>
3  #include <iostream>
4
5  static void check_valid_ipv4() {
6      std::string s = "10.192.168.1";
7      bool b = websocketpp::uri_helper::ipv4_literal(s.begin(), s.end());
8      assert(b == true);
9
10     s = "256.192.168.1";
11     b = websocketpp::uri_helper::ipv4_literal(s.begin(), s.end());
12     assert(b == false);
13 }
14
15 static void check_uri() {
16     websocketpp::uri uri("ws://localhost:81");
17     assert(uri.str() == "ws://localhost:81/");
18     assert(uri.get_valid() == true);
19     assert(uri.is_ipv6_literal() == false);
20     assert(uri.get_secure() == false);
21     assert(uri.get_scheme() == "ws");
22     assert(uri.get_host() == "localhost");
23     assert(uri.get_port() == 81);
24     assert(uri.get_port_str() == "81");
25     assert(uri.get_host_port() == "localhost:81");
26     assert(uri.get_authority() == "localhost:81");
27     assert(uri.get_resource() == "/");
28     assert(uri.get_query() == "");

```

(continues on next page)

(continued from previous page)

```
29 }
30
31 static void check_uri2() {
32     websocketpp::uri uri("wss://localhost/foo/bar?hello=12");
33     assert(uri.str() == "wss://localhost/foo/bar?hello=12");
34     assert(uri.get_valid() == true);
35     assert(uri.is_ipv6_literal() == false);
36     assert(uri.get_secure() == true);
37     assert(uri.get_scheme() == "wss");
38     assert(uri.get_host() == "localhost");
39     assert(uri.get_port() == 443);
40     assert(uri.get_port_str() == "443");
41     assert(uri.get_host_port() == "localhost");
42     assert(uri.get_authority() == "localhost:443");
43     assert(uri.get_resource() == "/foo/bar?hello=12");
44     assert(uri.get_query() == "hello=12");
45 }
46
47 void test_uri() {
48     check_valid_ipv4();
49     check_uri();
50     check_uri2();
51 }
```


OPERATING SYSTEMS

33.1 macos

Copy content to the clipboard:

```
cat result.txt | pbcopy
```

33.1.1 Port scan

```
brew install masscan  
sudo masscan -p 1-10000 137.184.92.35
```

33.2 lldb

- GDB to LLDB command map <https://lldb.llvm.org/use/map.html>

In gdb, we use `frame 1` to switch to frame 1, but in lldb, we use `frame select 1`.

34.1 C++ String

Note: `std::string` contains an array of bytes. Its length is the number of bytes in the array.

Listing 1: `./code/cpp_string.cc`

```
#include <iostream>
#include <string>
int main() {
    std::string s = "nihaohello";
    auto p = s.c_str();
    fprintf(stderr, "s.size: %d\n", (int)s.size());
    for (int i = 0; i != s.size(); ++i) {
        fprintf(stderr, "%d: %02x, %c\n", i, p[i], p[i]);
    }
    return 0;
    // UTF8 encoding of s is
    // \xe4\xbd\xa0\xe5\xa5\xbd\x68\x65\x6c\x6c\x6f
    //
    // UTF8 encoding of ni is 0xe4 0xbd 0xa0
    // UTF8 encoding of hao is 0xe5 0xa5 0xbd
    // UTF8 encoding of h is 0x68
    // UTF8 encoding of e is 0x65
    // UTF8 encoding of l is 0x6c
    // UTF8 encoding of o is 0x6f
}
/*
It prints:
s.size: 11

0: ffffffe4, x
1: ffffffbd, x
2: ffffffa0, x
3: ffffffe5, x
4: ffffffa5, x
5: ffffffbd, x
6: 68, h
7: 65, e
8: 6c, l
9: 6c, l
```

(continues on next page)

(continued from previous page)

```
10: 6f, 0
*/
```

34.2 kotlin String

Note: The length kotlin String equals to the number of characters, not number of bytes.

Listing 2: ./code/kotlin_string.kt

```
fun main() {
    var s = "nihaohello"
    println("s.length: ${s.length}")
    var bytes : ByteArray = "nihaohello".toByteArray()
    // for(b in bytes) {
    //     println(b.toUByte().toString(radix = 16))
    // }
    bytes.forEachIndexed{i, b ->
        println("$i, ${b.toUByte().toString(radix=16)}")
    }
}
/*
s.length: 7

0, e4
1, bd
2, a0
3, e5
4, a5
5, bd
6, 68
7, 65
8, 6c
9, 6c
10, 6f
*/
}
```

34.3 Basics

If the terminal displays chinese characters as ???, then use the following environment variable to fix it.

```
export LC_ALL=en_US.UTF-8
```

34.4 TODOs

- Unicode Characters – What Every Developer Must Know About Encoding

<https://www.freecodecamp.org/news/everything-you-need-to-know-about-encoding/>

35.1 Call cpp

1. Start xcode
2. Create a new project, macOS -> Command Line Tool
3. Product name: TestCpp
4. Language Swift
5. Edit main.swift, keep only `println("hello world")` and remove other lines
6. Product -> Run
7. Create a c++ shared library

Listing 1: ./code/call_cpp/hello.h

```
class A {  
public:  
    A(int);  
    int getInt() const;  
  
private:  
    int i_;  
};
```

Listing 2: ./code/call_cpp/hello.cc

```
#include "hello.h"  
  
A::A(int k) : i_(k) {}  
int A::getInt() const { return i_; }
```

Listing 3: ./code/call_cpp/Makefile

```
all: libhello.a  
  
libhello.a: hello.h hello.cc  
    g++ -c hello.cc -o hello.o  
    ar r libhello.a hello.o
```

(continues on next page)

(continued from previous page)

clean:`$(RM) libhello.a hello.o`

8. In xcode, project->build phases->frameworks and libraries, click +, and select `libhello.a`. Then, modify build settings to change the library search paths (in search paths)
9. Add `hello.h` to the same folder of `main.swift`. File->Add files to TestCpp.
10. Add a wrapper. File->New->File->C++ file, next, choose an arbitrary name, e.g., `wrapper.cc`. Uncheck Also create a header file. We only need the `.cc` file. In the popped-up dialog, select Create bridging header.
11. If we don't select Create bridging header, we have to go to build settings, swift compiler, objective-c bridging header, and enter a header name.
12. Content of `wrapper.cc`

```
#include "hello.h"
extern "C" int getIntFromCpp() {
    return A(10).getInt();
}
```

13. Content of the bridging header `TestCpp-Briding-Header.h`:

```
int getIntFromCpp();
```

14. In `main.swift`, use `print(getIntFromCpp())`

35.2 Swift

35.2.1 Basics

See

- A Swift Tour
https://docs.swift.org/swift-book/GuidedTour/GuidedTour.html#apple_ref/doc/uid/TP40014097-CH2-ID1
- Getting Started
<https://www.swift.org/getting-started>
- Size, Stride, Alignment
<https://swiftunboxed.com/internals/size-stride-alignment/>
- Swift And C: Everything You Need to Know on Types, Pointers and more
<https://www.uraimo.com/2016/04/07/swift-and-c-everything-you-need-to-know/>

```
swift --version
swift-driver version: 1.62.15 Apple Swift version 5.7.2 (swiftlang-5.7.2.135.5 clang-
↳ 1400.0.29.51)
Target: x86_64-apple-macosx13.0
```

To start the REPL commandline:

```
swift repl
```

Listing 4: ./code/1-hello.swift

```
// swiftc ./1-hello.swift

print("Hello, world!")
// Note:
// 1. It is not println but it still prints a new line
// 2. There is no semicolon
// 3. There is no main(), like Python
```

```
swiftc ./1-hello.swift
```

It will generate an executable ./1-hello.

```
otool -L ./1-hello

./1-hello:
/usr/lib/libobjc.A.dylib (compatibility version 1.0.0, current version 228.0.0)
/usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1319.0.0)
/usr/lib/swift/libswiftCore.dylib (compatibility version 1.0.0, current version 5.7.1)

ls -lh 1-hello

-rwxr-xr-x  1 fangjun  staff   33K Jan 21 11:55 1-hello
```

Listing 5: ./code/2-variables.swift

```
var myVariable = 42
myVariable = 50
let myConstant = 43
print("myVariable is \(myVariable)")
print("myConstant is \(myConstant)")

let implicitInteger = 1
let implicitDouble = 1.0
let explicitDouble: Double = 1

let label = "The width is "
let width = 100
var widthLabel = label + String(width)
widthLabel = "\(label)\(width)"
print(widthLabel)

// """
let s = """
abc
def
"""
// Note: there are leading spaces before the ending """
print(s)
```

Listing 6: ./code/3-array-dict.swift

```
var fruits = ["strawberries", "limes", "apples"]
fruits[1] = "grapes"

print(fruits) // ["strawberries", "grapes", "apples"]

fruits.append("blueberries")

var occupations = [
    "Tom": "Captain",
    "Jerry": "Mechanic",
]
print(occupations) // ["Tom": "Captain", "Jerry": "Mechanic"]
occupations["Tom"] = "Teacher"

let emptyArray: [String] = []
let emptyDict: [String: Float] = [:]
```

Listing 7: ./code/4-if.swift

```
let individualScores = [75, 43, 103, 87, 12]
var teamScore = 0
for score in individualScores {
    if score > 50 {
        teamScore += 3
    } else {
        teamScore += 1
    }
}
print(teamScore) // 11

var optionalString: String? = "Hello"
print(optionalString == nil) // false

var optionalName: String? = "Tom"
var greeting = "Hello!"
if let name = optionalName {
    greeting = "Hello, \(name)"
}
print(greeting) // Hello, Tom

let nickname: String? = nil
let fullName: String = "Tom Green"
let informalGreeting = "Hi, \(nickname ?? fullName)" // Hi, Tom Green
print(informalGreeting)

if let nickname {
    print("Hey, \(nickname)")
}
```


Listing 8: ./code/5-switch.swift

```
let vegetable = "red pepper"
switch vegetable {
    case "celery":
        print("celery")
    case "cucumber", "watercress":
        print("cucumber or watercress")
    case let x where x.hasSuffix("pepper"):
        print("x")
    default:
        print("Everything tastes good in soup.")
}
// 1. no need to use break
// 2. default is mandatory so that it is exhaustive
```

Listing 9: ./code/6-for.swift

```
let interestingNumbers = [
    "Prime": [2, 3, 5, 7, 11, 13],
    "Fibonacci": [1, 1, 2, 3, 5, 8],
    "Square": [1, 4, 9, 16, 25],
]
var largest = 0
for (_, numbers) in interestingNumbers {
    for number in numbers {
        if number > largest {
            largest = number
        }
    }
}
print(largest) // 25

var total = 0
for i in 0..<4 {
    total += i
}
print(total) // 6

total = 0
for i in 0...4 {
    total += i
}
print(total) // 10
```

Listing 10: ./code/7-while.swift

```
var n = 2
while n < 100 {
    n *= 2;
}
print(n) // 128
```

(continues on next page)

(continued from previous page)

```

var m = 2
repeat {
    m *= 2;
} while (m < 100)

print(m) // 128

```

Listing 11: ./code/8-func.swift

```

func greet(person: String, day: String) -> String {
    return "Hello \(person), today is \(day)"
}

print(greet(person: "Bob", day: "Tuesday"))
// print(greet("Bob", day: "Tuesday")) // error: missing argument label 'person:' in_
↳ call

func greet2(_ person: String, on day: String) -> String {
    return "Hello \(person), today is \(day)"
}
// print(greet2(person: "Bob", day: "Tuesday")) // error: incorrect argument labels_
↳ in call (have 'person:day:', expected '_:on:')
print(greet2("Bob", on: "Tuesday"))

func calculateStatistics(scores: [Int]) -> (min: Int, max: Int, sum: Int) {
    var min = scores[0]
    var max = scores[1]
    var sum = 0

    for score in scores {
        if score > max {
            max = score
        } else if score < min {
            min = score
        }

        sum += score
    }

    return (min, max, sum)
}

let statistics = calculateStatistics(scores: [5, 3, 100, 3, 9])
print(statistics.sum) // 120
print(statistics.2) // 120

func returnFifteen() -> Int {
    var y = 10
    func add() {
        y += 5
    }
    add()
}

```

(continues on next page)

(continued from previous page)

```

    return y
}
print(returnFifteen()) // 15

func makeIncrementer() -> ((Int) -> Int) {
    func addOne(number: Int) -> Int {
        return number + 1
    }

    return addOne
}

var increment = makeIncrementer()
print(increment(7)) // 8

func hasAnyElement(list: [Int], condition: (Int) -> Bool) -> Bool {
    for item in list {
        if condition(item) {
            return true
        }
    }
    return false
}

func lessThanTen(number: Int) -> Bool {
    return number < 10
}

var numbers = [20, 19, 7, 12]
print(hasAnyElement(list: numbers, condition: lessThanTen)) // true

// the closure has to be put in {}
print(hasAnyElement(list: numbers, condition: {(number: Int) -> Bool in number < 1})) // ↳
↳ false

// we can omit the input argument type and return type
print(hasAnyElement(list: numbers, condition: {number in number < 1})) // false

print(numbers.map({ (number: Int) -> Int in
    let result = 3 * number
    return result
})) // [60, 57, 21, 36]

let mappedNumber = numbers.map({number in 3 * number})
print(mappedNumber) // [60, 57, 21, 36]

var sortedNumbers = numbers.sorted {$0 > $1}
print(sortedNumbers) // [20, 19, 12, 7]

sortedNumbers = numbers.sorted {$0 < $1} // also ok
print(sortedNumbers) // [7, 12, 19, 20]

```

Listing 12: ./code/9-class.swift

```
class Shape {
    var numberOfSides = 0
    func simpleDescription() -> String {
        return "A shape with \(numberOfSides) sides."
    }
}

var shape = Shape()
print(shape.numberOfSides) // 0
shape.numberOfSides = 4
print(shape.numberOfSides) // 4
var simpleDescription = shape.simpleDescription()
print(simpleDescription) // A shape with 4 sides

// with constructor
class NamedShape {
    var numberOfSides = 0
    var name: String

    init(name: String, numberOfSides: Int) {
        self.name = name
        self.numberOfSides = numberOfSides
    }

    func simpleDescription() -> String {
        return "\(name): A shape with \(numberOfSides) sides"
    }
}

var namedShape = NamedShape(name: "Hello", numberOfSides: 10)
print(namedShape.simpleDescription())

class Square: NamedShape {
    var sideLength: Double

    init(sideLength: Double, name: String) {
        self.sideLength = sideLength
        super.init(name: name, numberOfSides: 4)
    }

    func area() -> Double {
        return sideLength * sideLength
    }

    override func simpleDescription() -> String {
        return "A square with side length \(sideLength)"
    }
}

var square = Square(sideLength: 10, name: "MySquare")
print(square.simpleDescription()) // A square with side length 10.0
```

(continues on next page)

(continued from previous page)

```

print(square.area()) // 100.0

// property getter/setter

class EquilateralTriangle: NamedShape {
    var sideLength: Double = 0.0

    init(sideLength: Double, name: String) {
        self.sideLength = sideLength
        super.init(name: name, numberOfSides: 3)
    }

    var perimeter: Double {
        get {
            return 3.0 * sideLength
        }
        set {
            // note: the new value has the implicit name newValue
            sideLength = newValue / 3.0
        }
    }

    override func simpleDescription() -> String {
        return "An equilateral triangle with side of length \(sideLength)."
    }
}

var triangle = EquilateralTriangle(sideLength: 10, name: "MyTriangle")
print(triangle.perimeter) // 30.0
triangle.perimeter = 15
print(triangle.perimeter) // 15.0
print(triangle.sideLength) // 5.0

```

Listing 13: ./code/10-enum.swift

```

enum Rank: Int {
    case ace = 1
    case two, three, four, five, six, seven, eight, nine, ten
    case jack, queen, king

    func simpleDescription() -> String {
        switch self {
            case .ace:
                return "ace"
            case .jack:
                return "jack"
            case .queen:
                return "queen"
            case .king:
                return "king"
            default:
                return String(self.rawValue)
        }
    }
}

```

(continues on next page)

(continued from previous page)

```

    }
}
let ace = Rank.ace
let aceRawValue = ace.rawValue
print(ace) // ace
print(aceRawValue) // 1

let two = Rank(rawValue: 2)
print(two ?? "zwei") // two

```

Listing 14: ./code/11-memory-layout.swift

```

assert(MemoryLayout<Bool>.size == 1)
assert(MemoryLayout<Bool>.stride == 1)
assert(MemoryLayout<Bool>.alignment == 1)

assert(MemoryLayout<Int>.size == 8)
assert(MemoryLayout<Int>.stride == 8)
assert(MemoryLayout<Int>.alignment == 8)

assert(MemoryLayout<Int32>.size == 4)
assert(MemoryLayout<Int32>.stride == 4)
assert(MemoryLayout<Int32>.alignment == 4)

// Like C/C++
struct Example {
    let foo: Int // 8
    let bar: Bool // 1
}
assert(MemoryLayout<Example>.size == 9)
assert(MemoryLayout<Example>.stride == 16)
assert(MemoryLayout<Example>.alignment == 8)

let ex = Example(foo: 10, bar: true)
assert(MemoryLayout.size(ofValue: ex) == 9)
assert(MemoryLayout.stride(ofValue: ex) == 16)
assert(MemoryLayout.alignment(ofValue: ex) == 8)

struct Example2 {
    let bar: Bool // 1
    let foo: Int // 8
}
assert(MemoryLayout<Example2>.size == 16)
assert(MemoryLayout<Example2>.stride == 16)
assert(MemoryLayout<Example2>.alignment == 8)

```

Listing 15: ./code/12-weak-reference.swift

```

class Author {
    let name: String
    // weak reference is required

```

(continues on next page)

(continued from previous page)

```

weak var post: Post?

init(name: String) {self.name = name}
deinit {print("Author deinit")}
}

class Post {
    let title: String

    var author: Author?
    init(title: String) {self.title = title}
    deinit {print("Post deinit")}
}

var author: Author? = Author(name: "John Snow")
var post: Post? = Post(title: "foo bar")
post?.author = author
author?.post = post

print(author?.post) // Optional(main.Post)
post = nil // Post deinit
print(author?.post) // nil
author = nil // Author deinit

```

- <https://developer.apple.com/documentation/swift/unsafeMutableRawPointer>

Listing 16: ./code/13-pointers.swift

```

// Unsafe[Mutable][Raw][Buffer]Pointer[Type]
//
// Mutable: means you can change the value
// Raw: means it points to a blob of bytes
// Buffer: means it works like a collection
// Type: means generic typed pointers
//
// UnsafePointer<T>
// UnsafeMutablePointer<T>
//
// UnsafeRawPointer
// UnsafeMutableRawPointer
//
// UnsafeRawBufferPointer
// UnsafeMutableRawBufferPointer
//
// UnsafeBufferPointer<T>
// UnsafeMutableBufferPointer<T>

let count = 2
let stride = MemoryLayout<Int>.stride
let alignment = MemoryLayout<Int>.alignment
let byteCount = count * stride

```

(continues on next page)

(continued from previous page)

```

let pointer = UnsafeMutableRawPointer.allocate(byteCount: byteCount, alignment:
↳alignment)

defer {
    pointer.deallocate()
}

pointer.storeBytes(of: 30, as: Int.self)
pointer.advanced(by: stride).storeBytes(of: 3, as: Int.self)
assert(pointer.load(as: Int.self) == 30)
assert(pointer.load(fromByteOffset: stride, as: Int.self) == 3)
assert(pointer.advanced(by: stride).load(as: Int.self) == 3)

let bufferPointer = UnsafeRawBufferPointer(start: pointer, count: byteCount)
for (index, byte) in bufferPointer.enumerated() {
    print("byte \(index) -> \(byte)")
}
/*
byte 0 -> 30
byte 1 -> 0
byte 2 -> 0
byte 3 -> 0
byte 4 -> 0
byte 5 -> 0
byte 6 -> 0
byte 7 -> 0
byte 8 -> 3
byte 9 -> 0
byte 10 -> 0
byte 11 -> 0
byte 12 -> 0
byte 13 -> 0
byte 14 -> 0
byte 15 -> 0
*/

```

- <https://developer.apple.com/documentation/swift/unsafemutablepointer>

Listing 17: ./code/14-typed-pointers.swift

```

let count = 2
let stride = MemoryLayout<Int>.stride
let pointer = UnsafeMutablePointer<Int>.allocate(capacity: count)
pointer.initialize(repeating: 0, count: count)

defer {
    pointer.deinitialize(count: count)
    pointer.deallocate()
}

pointer.pointee = 42
pointer.advanced(by: 1).pointee = 6

```

(continues on next page)

(continued from previous page)

```

let bufferPointer = UnsafeBufferPointer(start: pointer, count: count)

for (index, value) in bufferPointer.enumerated() {
    print("value \(index) -> \(value)")
}
/*
value 0 -> 42
value 1 -> 6
*/

```

35.2.2 Package

See

- Using the Package Manager
<https://www.swift.org/getting-started/#using-the-package-manager>
- Package Manager
<https://www.swift.org/package-manager/>
- How Swift imports C APIs
<https://github.com/apple/swift/blob/main/docs/HowSwiftImportsCAPIs.md>

hello

```

mkdir hello
cd hello
swift package init

```

```

.
├── Package.swift
├── README.md
├── Sources
│   └── hello
│       └── hello.swift
├── Tests
│   └── helloTests
│       └── helloTests.swift

```

4 directories, 4 files

Listing 18: `./code/package/hello/Package.swift`

```

// swift-tools-version: 5.7
// The swift-tools-version declares the minimum version of Swift required to build
↳ this package.

import PackageDescription

```

(continues on next page)

(continued from previous page)

```

let package = Package(
    name: "hello",
    products: [
        // Products define the executables and libraries a package produces, and
        ↪ make them visible to other packages.
        .library(
            name: "hello",
            targets: ["hello"]),
    ],
    dependencies: [
        // Dependencies declare other packages that this package depends on.
        // .package(url: /* package url */, from: "1.0.0"),
    ],
    targets: [
        // Targets are the basic building blocks of a package. A target can define
        ↪ a module or a test suite.
        // Targets can depend on other targets in this package, and on products in
        ↪ packages this package depends on.
        .target(
            name: "hello",
            dependencies: []),
        .testTarget(
            name: "helloTests",
            dependencies: ["hello"]),
    ]
)

```

Listing 19: ./code/package/hello/README

```

# hello

A description of this package.

```

Listing 20: ./code/package/hello/Sources/hello/hello.swift

```

public struct hello {
    public private(set) var text = "Hello, World!"

    public init() {
    }
}

```

Listing 21: ./code/package/hello/Tests/helloTests/helloTests.swift

```

import XCTest
@testable import hello

final class helloTests: XCTestCase {
    func testExample() throws {
        // This is an example of a functional test case.
        // Use XCTAssert and related functions to verify your tests produce the

```

(continues on next page)

(continued from previous page)

```

↪correct
    // results.
    XCTAssertEqual(hello().text, "Hello, World!")
}
}

```

hello2

```

mkdir hello2
cd hello2
swift package init --type executable

```

```

.
├── Package.swift
├── README.md
├── Sources
│   └── hello2
│       └── hello2.swift
├── Tests
│   └── hello2Tests
│       └── hello2Tests.swift

```

4 directories, 4 files

Listing 22: ./code/package/hello2/Package.swift

```

// swift-tools-version: 5.7
// The swift-tools-version declares the minimum version of Swift required to build
↪this package.

import PackageDescription

let package = Package(
    name: "hello2",
    dependencies: [
        // Dependencies declare other packages that this package depends on.
        // .package(url: /* package url */, from: "1.0.0"),
    ],
    targets: [
        // Targets are the basic building blocks of a package. A target can define
↪a module or a test suite.
        // Targets can depend on other targets in this package, and on products in
↪packages this package depends on.
        .executableTarget(
            name: "hello2",
            dependencies: []),
        .testTarget(
            name: "hello2Tests",
            dependencies: ["hello2"]),
    ]

```

(continues on next page)

(continued from previous page)

)

Listing 23: ./code/package/hello2/README

```
# hello2

A description of this package.
```

Listing 24: ./code/package/hello2/Sources/hello2/hello2.swift

```
@main
public struct hello2 {
    public private(set) var text = "Hello, World!"

    public static func main() {
        print(hello2().text)
    }
}
```

Listing 25: ./code/package/hello2/Tests/hello2Tests/hello2Tests.swift

```
import XCTest
@testable import hello2

final class hello2Tests: XCTestCase {
    func testExample() throws {
        // This is an example of a functional test case.
        // Use XCTAssert and related functions to verify your tests produce the
        ↪correct
        // results.
        XCTAssertEqual(hello2().text, "Hello, World!")
    }
}
```

```
swift run hello2
```

```
Building for debugging...
[3/3] Linking hello2
Build complete! (0.88s)
Hello, World!
```

35.3 Objective C

35.3.1 Basics

Where to find Foundation.h:

```
cd /Applications/Xcode.app/
find . -name "Foundation.h"
```

```
./Contents/Developer/Platforms/AppleTVOS.platform/Developer/SDKs/AppleTVOS.sdk/System/  
↳ Library/Frameworks/Foundation.framework/Headers/Foundation.h  
./Contents/Developer/Platforms/iPhoneOS.platform/Developer/SDKs/iPhoneOS.sdk/System/  
↳ Library/Frameworks/Foundation.framework/Headers/Foundation.h  
./Contents/Developer/Platforms/WatchOS.platform/Developer/SDKs/WatchOS.sdk/System/  
↳ Library/Frameworks/Foundation.framework/Headers/Foundation.h  
./Contents/Developer/Platforms/MacOSX.platform/Developer/SDKs/MacOSX.sdk/System/Library/  
↳ Frameworks/Foundation.framework/Versions/C/Headers/Foundation.h
```


EMBEDDED SYSTEMS

36.1 m3axpi

- https://wiki.sipeed.com/hardware/zh/maixIII/ax-pi/basic_usage.html

36.1.1 How to setup date

```
# If network is not available, then setup the http proxy
#
# export http_proxy=http://localhost:6009
# export HTTP_PROXY=http://localhost:6009
sudo date -s "$(wget --no-cache -S -O /dev/null google.com 2>&1 | \
sed -n -e '/ *Date: */ {' -e s///p -e q -e '})'"

# or use
sudo date -s "$(wget --no-cache -S -O /dev/null bing.com 2>&1 | \
sed -n -e '/ *Date: */ {' -e s///p -e q -e '})'"
```

36.1.2 Python proxy server

<https://github.com/csukuangfj/python-proxy-server>

36.2 Raspberry PI

36.2.1 How to connect it to Windows with direct ethernet connection

1. Change the firewall on Windows to allow port for DHCP. The DHCP server uses port 67, supporting both TCP and UDP
2. Download a DHCP server from <https://www.dhcpserver.de/cms/download/>. I have uploaded it to <https://huggingface.co/csukuangfj/dhcp-server-for-windows>
3. Change the IP address of the network interface on Windows to 192.168.1.2.
4. Download and install wireshark. Start it to capture the traffic of the network interface, on which we will run the DHCP server. We can use the filter dhcp to filter out packets.

5. Start the `dhcpcwiz.exe` to configure the DHCP server. We set the starting address to `192.168.1.100`. Note: Don't put `192.168.1.2` into the pool since it has already been occupied.
6. Use a network cable to connect the windows machine with Raspberry pi.
7. We can see the IP address of the Raspberry pi from the packets captured by Wireshark.

36.2.2 Default IP of Raspberry pi

```
pi@raspberrypi:~ $ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 169.254.49.182 netmask 255.255.0.0 broadcast 169.254.255.255
    inet6 fe80::11f:baaf:1325:9f1e prefixlen 64 scopeid 0x20<link>
    ether dc:a6:32:cc:76:1c txqueuelen 1000 (Ethernet)
    RX packets 275806 bytes 67146923 (64.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5359 bytes 454586 (443.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

We can use Wireshark to capture its network traffic and find its address. Then we can assign a static address, e.g., `169.254.49.181` to our own computer.

`169.254.0.0/16` IPv4 link local addresses.

36.2.3 Headless setup

1. Download the 64-bit image from <https://www.raspberrypi.com/software/operating-systems/>. The current version is <https://www.raspberrypi.com/software/operating-systems/#raspberrypi-os-64-bit>
2. After flashing to a SD card with BalenaEtcher, create two files in the drive `bootfs`
 - A empty file with name `ssh`, not it has no extension.
 - **userconf.txt** containing

```
pi:$6$45qY/WWAHw7hR5Fk
↪$30VJUl878J3iQhc2WczQhOtjmiqWijSgUE9oY92v1BIl4n9PiF2TPzxRjlf/wMshZZJ00ee/
↪cdDxYoQeJvtKR0
```

`pi`` is the user name while the second part is the encrypted password ``pi` generated by `openssl passwd -6`.

36.3 ALSA

Advanced Linux Sound Architecture (ALSA).

36.3.1 USB microphone for raspberry pi

```
pi@raspberrypi:~ $ arecord -l
**** List of CAPTURE Hardware Devices ****
card 3: UACDemoV10 [UACDemoV1.0], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

36.3.2 alsa-utils

See <https://github.com/alsa-project/alsa-utils>

```
sudo apt-get install alsa-utils
```

It contains source code for arecord, areplay, etc.

aplay

List playback devices:

```
aplay -l

**** List of PLAYBACK Hardware Devices ****
card 1: vc4hdmi0 [vc4-hdmi-0], device 0: MAI PCM i2s-hifi-0 [MAI PCM i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: vc4hdmi1 [vc4-hdmi-1], device 0: MAI PCM i2s-hifi-0 [MAI PCM i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

To play foo.wav on card 1, device 0, use:

```
aplay -D hw:1,0 foo.wav
```

arecord

```
cat /proc/asound/pcm
00-00: bcm2835 Headphones : bcm2835 Headphones : playback 8
01-00: MAI PCM i2s-hifi-0 : MAI PCM i2s-hifi-0 : playback 1
02-00: MAI PCM i2s-hifi-0 : MAI PCM i2s-hifi-0 : playback 1
03-00: USB Audio : USB Audio : capture 1
```

```
arecord -l
**** List of CAPTURE Hardware Devices ****
card 3: UACDemoV10 [UACDemoV1.0], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

```
cat /proc/asound/card3/stream0

Jieli Technology UACDemoV1.0 at usb-0000:01:00.0-1.4, full speed : USB Audio

Capture:
  Status: Stop
  Interface 1
    Altset 1
    Format: S16_LE
    Channels: 1
    Endpoint: 0x83 (3 IN) (ASYNC)
    Rates: 48000
    Bits: 16
    Channel map: MONO
```

36.3.3 Cross compiling alsa-lib

See <https://github.com/alsa-project/alsa-lib/blob/master/INSTALL>

```
git clone --depth 1 https://github.com/alsa-project/alsa-lib
cd alsa-lib
CC=aarch64-linux-gnu-gcc ./gitcompile --host=aarch64-linux-gnu
```

```
/bin/bash: plantuml: command not found: sudo apt-get install plantuml
```

```
alsa-lib$ find . -name "*.so"

./src/.libs/libasound.so
./src/topology/.libs/libatopology.so
```

```
alsa-lib$ find . -name asoundlib.h
./include/asoundlib.h
```

Then copy the include directory to raspberry pi and rename include to alsa. Also, copy libasound.so to raspberry pi.

We can pass `-I/path/to/alsa/parent` and `-lasound` to `g++` on raspberry pi.

36.3.4 Introduction to Sound Programming with ALSA

```
sudo apt-get install libasound2-dev libasound2
```

See

- <https://www.linuxjournal.com/article/6735>.
- <https://www.alsa-project.org/alsa-doc/alsa-lib/index.html>
- <https://vovkos.github.io/doxyrest/samples/alsa/index.html>, which has more detailed doc for APIs.
- A Tutorial on Using the ALSA Audio API
- PCM (digital audio) interface

- `snd_pcm_open`
- `snd_strerror`
- `snd_device_name_hint`
- `snd_pcm_hw_params_malloc`
- `snd_pcm_hw_params_free`
- `snd_pcm_hw_params_any`
- `snd_pcm_hw_params_set_access`
- `snd_pcm_hw_params_set_format`
- `snd_pcm_hw_params_set_rate_near`
- `snd_pcm_hw_params_set_channels`
- `snd_pcm_hw_params`
- `snd_async_add_pcm_handler`, which calls `snd_pcm_prepare` automatically.
- `snd_pcm_prepare`
- `snd_pcm_start`
- `snd_pcm_drop`
- `snd_pcm_drain`
- `snd_pcm_readi`
- `snd_pcm_readn`
- `snd_pcm_format_t`
- `snd_pcm_hw_params_alloca`

36.4 nano-pi

- user: pi, password pi
- user: root, password: fa

How to assign a static IP on Ubuntu 16.04:

```
ifconfig # it shows eth0  
  
sudo vim /etc/network/interfaces
```

The default content is:

```
auto lo  
iface lo inet loopback
```

Add the following:

```
# this is a comment line  
# iface eth0 inet dhcp  
iface eth0 inet static
```

(continues on next page)

(continued from previous page)

```
address 192.168.1.100
netmask 255.255.255.0
```

```
sudo /etc/init.d/networking restart
```

Recording:

```
arecord -l
arecord -Dhw:0,1 -f S16_LE -r 16000 -d 10 -c 2 -t wav test.wav
```

Install cmake from source:

```
./configure -- -DCMAKE_USE_OPENSSL=OFF -DBUILD_TESTING=OFF
```

```
sudo apt-get install cmake
sudo apt-get install libasound2-dev alsa-utils
```

ssh config on macos:

```
Host nano-pi
  Hostname 192.168.1.3
  User pi
  IdentityFile ~/.ssh/pi
  RemoteForward 8080 localhost:8080
```

```
sudo ifup eth0
```

36.5 star visionfive 2

boot mode settings:

- https://doc-en.rvspace.org/VisionFive2/Quick_Start_Guide/VisionFive2_SDK_QSG/boot_mode_settings.html
- RGPI0_1: low, RGPI_0: high for sd card

Note: The network interface name of the board is end1 and end0

- QR CODE -> end0
- FLY_CORE FC-11Q280 2307 -> end1, I have set static IP address for it: 192.168.1.100

Login:

- username: user
- password: starfive (the same password for root)

Install toolchains:

```
sudo apt-get install -y gcc-riscv64-linux-gnu
```

Setup proxy:

```
# on my macOS
Host starfive
  Hostname 192.168.1.100
  User user
  IdentityFile ~/.ssh/starfive
  RemoteForward 8080 localhost:8080
```

```
# on the board
export http_proxy=http://localhost:8080
export HTTP_PROXY=http://localhost:8080

sudo touch /etc/apt/apt.conf.d/proxy.conf
sudo vi /etc/apt/apt.conf.d/proxy.conf

# Add one line to proxy.conf
Acquire::http::Proxy "http://user:password@proxy.server:port/";
Acquire::https::Proxy "http://user:password@proxy.server:port/";
```

Python proxy server

<https://github.com/csukuangfj/python-proxy-server>

Free space:

```
df -h
cd /usr/share
sudo rm -rf help
sudo rm -rf doc
```

install usb:

```
wget https://snapshot.debian.org/archive/debian-ports/20221225T084846Z/pool-riscv64/main/
↳ u/usbutils/usbutils_014-1+b1_riscv64.deb

sudo apt-get install ./usbutils*.deb

wget https://snapshot.debian.org/archive/debian-ports/20221225T084846Z/pool-riscv64/main/
↳ n/net-tools/net-tools_2.10-0.1_riscv64.deb
sudo apt-get install ./net-tools*.deb
```

```
lsusb
cat /proc/asound/cards
lsmod
```

Compile the kernel by ourselves:

See <https://github.com/starfive-tech/VisionFive2>

```
git clone https://github.com/starfive-tech/VisionFive2

sudo apt-get install build-essential g++ git autoconf automake autotools-dev texinfo
↳ bison xxd curl flex gawk gdisk gperf libgmp-dev libmpfr-dev libmpc-dev libz-dev libssl-
↳ dev libncurses-dev libtool patchutils python3 screen texinfo unzip zlib1g-dev libyaml-
```

(continues on next page)

(continued from previous page)

```

↪dev wget cpio bc dosfstools mtools device-tree-compiler libglib2.0-dev libpixman-1-dev
↪kpartx

sudo apt-get install git-lfs

cd ~/VisionFive2/
git checkout JH7110_VisionFive2_devel
git submodule update --init --recursive
cd buildroot && git checkout --track origin/JH7110_VisionFive2_devel && cd ..
cd u-boot && git checkout --track origin/JH7110_VisionFive2_devel && cd ..
cd linux && git checkout --track origin/JH7110_VisionFive2_devel && cd ..
cd opensbi && git checkout master && cd ..
cd soft_3rdpart && git checkout JH7110_VisionFive2_devel && cd ..

cd ~/VisionFive2/soft_3rdpart/IMG_GPU/out
git lfs pull

cd ~/VisionFive2/
make -j$(nproc)
make buildroot_rootfs -j$(nproc)
make img

# The generated file is work/sdcard.img
# username: root
# password: starfive
# The inner network interface is eth1, i.e., the inerface with name FLY_CORE
# The outer network interface is eth0, i.e., the interface with a QR code

```

```

# after login with root
vi /etc/network/interfaces

# iface eth1 inet dhcp
#   pre-up /etc/network/nfs_check
#   wait-delay 15
#   hostname $(hostname)
iface eth1 inet static
    address 192.168.1.100
    netmask 255.255.255.0

```

```
/etc/init.d/S40network restart
```

To show the current directory in the terminal:

```

PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01;34m\]\
↪W\[\033[00m\]\$ '

```

36.6 Lichee Pi 4A

Refer to *Raspberry PI* to setup a DHCP server

After the board gets an IP, ssh to it.

```
/sbin/ifconfig
```

It shows two interfaces: `end0` and `end1`. We don't use `lo` and `wlan0`.

`end0` is for Ethernet0 and `end1` for Ethernet1.

Now we assign a static address to `end0`.

```
sudo vim /etc/network/interfaces
```

The default content is:

```
source /etc/network/interfaces.d/*
```

We append the following to `/etc/network/interfaces`:

```
# this is a comment line
# iface eth0 inet dhcp
auto end0
iface eth0 inet static
    address 192.168.1.8
    netmask 255.255.255.0
```

```
sudo /etc/init.d/networking restart
```

Power off the board and connect `end0` to my mac.

Create a user:

```
sudo adduser fangjun
sudo usermod -a -G sudo fangjun
```

36.6.1 openKylin

https://wiki.sipeed.com/hardware/zh/lichee/th1520/lpi4a/3_images.html#openKylin

Download address <https://www.openkylin.top/downloads/index-cn.html>. Select openKylin 2.0 alpha RISC-V for LiCheePi4A.

```
-rw-rw-r-- 1 fangjun fangjun 2.8G Mar 7 04:06 openKylin-2.0-alpha-licheepi4a.tar.xz
```

Power off the board, press the BOOT key and don't release it, power on, and then release it. In virtual box, select the USB device T-HEAD USB download gadget [0223].

scp `openKylin-2.0-alpha-licheepi4a.tar.xz` to my virtual machine.

```
scp -P 1234 openKylin-2.0-alpha-licheepi4a.tar.xz localhost:/mnt/sdb/shared
```

Refer to <https://docs.openkylin.top/zh/%E7%A4%BE%E5%8C%BA%E5%BC%80%E5%8F%91%E6%8C%87%E5%8D%97/riscv%E>

On virtual machine:

```
mkdir t
cd t
tar xf ../openKylin-2.0-alpha-licheepi4a.tar.xz
cd openkylin-2.0-alpha-licheepi4a/fastboot/linux/
```

```
$ lsusb
Bus 002 Device 003: ID 2345:7654 T-HEAD USB download gadget
```

```
sudo ./fastboot/linux/fastboot flash ram ./images/8G/u-boot-nonsec-2020.10-r0-noswap.bin

Warning: skip copying ram image avb footer (ram partition size: 0, ram image size:
↳957544).
Sending 'ram' (935 KB)                OKAY [ 0.384s]
Writing 'ram'                        OKAY [ 0.008s]
Finished. Total time: 0.444s
```

```
sudo ./fastboot/linux/fastboot reboot

Rebooting                            OKAY [ 0.031s]
Finished. Total time: 1.451s
```

```
sleep 1

sudo ./fastboot/linux/fastboot flash uboot ./images/8G/u-boot-nonsec-2020.10-r0-noswap.
↳bin

Warning: skip copying uboot image avb footer (uboot partition size: 0, uboot image
↳size: 957544).
Sending 'uboot' (935 KB)              OKAY [ 0.259s]
Writing 'uboot'                      OKAY [ 0.081s]
Finished. Total time: 0.403s
```

```
sudo ./fastboot/linux/fastboot flash boot ./images/8G/boot.ext4

Warning: skip copying boot image avb footer (boot partition size: 409600, boot image
↳size: 61440000).
Sending 'boot' (60000 KB)             OKAY [ 14.813s]
Writing 'boot'                      OKAY [ 0.462s]
Finished. Total time: 15.335s
```

```
sudo ./fastboot/linux/fastboot flash root ./images/openkylin-2.0-alpha-licheepi4a-
↳riscv64.ext4

Sending sparse 'root' 1/64 (113621 KB) OKAY [ 28.396s]
Writing 'root'                        OKAY [ 1.437s]
Sending sparse 'root' 2/64 (114573 KB) OKAY [ 28.678s]
Writing 'root'                        OKAY [ 1.119s]
Sending sparse 'root' 3/64 (114465 KB) OKAY [ 31.803s]
Writing 'root'                        OKAY [ 3.088s]
Sending sparse 'root' 4/64 (114687 KB) OKAY [ 29.581s]
Writing 'root'                        OKAY [ 1.332s]
```

(continues on next page)

(continued from previous page)

Sending sparse 'root' 5/64 (114686 KB)	OKAY [29.088s]
Writing 'root'	OKAY [1.243s]
Sending sparse 'root' 6/64 (114684 KB)	OKAY [28.922s]
Writing 'root'	OKAY [0.926s]
Sending sparse 'root' 7/64 (110724 KB)	OKAY [27.861s]
Writing 'root'	OKAY [0.868s]
Sending sparse 'root' 8/64 (106020 KB)	OKAY [26.523s]
Writing 'root'	OKAY [0.905s]
Sending sparse 'root' 9/64 (114685 KB)	OKAY [29.080s]
Writing 'root'	OKAY [0.990s]
Sending sparse 'root' 10/64 (113701 KB)	OKAY [28.919s]
Writing 'root'	OKAY [1.000s]
Sending sparse 'root' 11/64 (114152 KB)	OKAY [29.780s]
Writing 'root'	OKAY [1.916s]
Sending sparse 'root' 12/64 (113537 KB)	OKAY [29.624s]
Writing 'root'	OKAY [1.600s]
Sending sparse 'root' 13/64 (113881 KB)	OKAY [29.739s]
Writing 'root'	OKAY [1.634s]
Sending sparse 'root' 14/64 (113118 KB)	OKAY [28.989s]
Writing 'root'	OKAY [1.169s]
Sending sparse 'root' 15/64 (106670 KB)	OKAY [27.178s]
Writing 'root'	OKAY [1.157s]
Sending sparse 'root' 16/64 (114685 KB)	OKAY [29.728s]
Writing 'root'	OKAY [1.656s]
Sending sparse 'root' 17/64 (112204 KB)	OKAY [28.763s]
Writing 'root'	OKAY [0.990s]
Sending sparse 'root' 18/64 (114547 KB)	OKAY [30.126s]
Writing 'root'	OKAY [1.495s]
Sending sparse 'root' 19/64 (111451 KB)	OKAY [28.567s]
Writing 'root'	OKAY [1.276s]
Sending sparse 'root' 20/64 (114651 KB)	OKAY [29.283s]
Writing 'root'	OKAY [1.380s]
Sending sparse 'root' 21/64 (104981 KB)	OKAY [27.260s]
Writing 'root'	OKAY [1.565s]
Sending sparse 'root' 22/64 (108444 KB)	OKAY [28.772s]
Writing 'root'	OKAY [1.958s]
Sending sparse 'root' 23/64 (112916 KB)	OKAY [27.882s]
Writing 'root'	OKAY [0.964s]
Sending sparse 'root' 24/64 (110055 KB)	OKAY [28.402s]
Writing 'root'	OKAY [1.373s]
Sending sparse 'root' 25/64 (114686 KB)	OKAY [29.349s]
Writing 'root'	OKAY [1.308s]
Sending sparse 'root' 26/64 (109020 KB)	OKAY [28.892s]
Writing 'root'	OKAY [1.883s]
Sending sparse 'root' 27/64 (113813 KB)	OKAY [30.411s]
Writing 'root'	OKAY [2.091s]
Sending sparse 'root' 27/64 (113813 KB)	OKAY [30.411s]
Writing 'root'	OKAY [2.091s]
Sending sparse 'root' 28/64 (112541 KB)	OKAY [28.959s]
Writing 'root'	OKAY [1.048s]
Sending sparse 'root' 29/64 (108317 KB)	OKAY [27.795s]
Writing 'root'	OKAY [0.927s]

(continues on next page)

(continued from previous page)

Sending sparse 'root' 30/64 (103881 KB)	OKAY [26.063s]
Writing 'root'	OKAY [0.883s]
Sending sparse 'root' 31/64 (112380 KB)	OKAY [28.106s]
Writing 'root'	OKAY [0.923s]
Sending sparse 'root' 32/64 (113912 KB)	OKAY [28.863s]
Writing 'root'	OKAY [0.887s]
Sending sparse 'root' 33/64 (107545 KB)	OKAY [27.507s]
Writing 'root'	OKAY [0.901s]
Sending sparse 'root' 34/64 (108964 KB)	OKAY [27.396s]
Writing 'root'	OKAY [0.924s]
Sending sparse 'root' 35/64 (114685 KB)	OKAY [28.831s]
Writing 'root'	OKAY [1.052s]
Sending sparse 'root' 36/64 (114635 KB)	OKAY [29.927s]
Writing 'root'	OKAY [1.857s]
Sending sparse 'root' 37/64 (106582 KB)	OKAY [34.001s]
Writing 'root'	OKAY [4.941s]
Sending sparse 'root' 38/64 (112349 KB)	OKAY [28.087s]
Writing 'root'	OKAY [1.019s]
Sending sparse 'root' 39/64 (114597 KB)	OKAY [28.444s]
Writing 'root'	OKAY [1.003s]
Sending sparse 'root' 40/64 (113833 KB)	OKAY [28.063s]
Writing 'root'	OKAY [0.955s]
Sending sparse 'root' 41/64 (111932 KB)	OKAY [27.982s]
Writing 'root'	OKAY [0.852s]
Sending sparse 'root' 42/64 (114656 KB)	OKAY [29.274s]
Writing 'root'	OKAY [0.919s]
Sending sparse 'root' 43/64 (114409 KB)	OKAY [28.753s]
Writing 'root'	OKAY [1.011s]
Sending sparse 'root' 44/64 (112007 KB)	OKAY [28.708s]
Writing 'root'	OKAY [1.232s]
Sending sparse 'root' 45/64 (114687 KB)	OKAY [29.922s]
Writing 'root'	OKAY [1.978s]
Sending sparse 'root' 46/64 (114684 KB)	OKAY [28.482s]
Writing 'root'	OKAY [0.903s]
Sending sparse 'root' 47/64 (111932 KB)	OKAY [28.518s]
Writing 'root'	OKAY [0.917s]
Sending sparse 'root' 48/64 (106984 KB)	OKAY [26.597s]
Writing 'root'	OKAY [0.869s]
Sending sparse 'root' 49/64 (114684 KB)	OKAY [28.722s]
Writing 'root'	OKAY [0.888s]
Sending sparse 'root' 50/64 (113296 KB)	OKAY [27.990s]
Writing 'root'	OKAY [0.901s]
Sending sparse 'root' 51/64 (114685 KB)	OKAY [28.955s]
Writing 'root'	OKAY [1.035s]
Sending sparse 'root' 52/64 (114684 KB)	OKAY [28.549s]
Writing 'root'	OKAY [0.906s]
Sending sparse 'root' 53/64 (114685 KB)	OKAY [29.215s]
Writing 'root'	OKAY [1.025s]
Sending sparse 'root' 54/64 (102348 KB)	OKAY [24.596s]
Writing 'root'	OKAY [1.030s]
Sending sparse 'root' 55/64 (113644 KB)	OKAY [28.859s]
Writing 'root'	OKAY [0.899s]

(continues on next page)

(continued from previous page)

```

Sending sparse 'root' 56/64 (111648 KB)      OKAY [ 28.154s]
Writing 'root'                               OKAY [ 0.940s]
Sending sparse 'root' 57/64 (105900 KB)      OKAY [ 26.766s]
Writing 'root'                               OKAY [ 0.832s]
Sending sparse 'root' 58/64 (104752 KB)      OKAY [ 26.516s]
Writing 'root'                               OKAY [ 0.832s]
Sending sparse 'root' 59/64 (114684 KB)      OKAY [ 28.399s]
Writing 'root'                               OKAY [ 0.963s]
Sending sparse 'root' 60/64 (114684 KB)      OKAY [ 28.732s]
Writing 'root'                               OKAY [ 0.912s]
Sending sparse 'root' 61/64 (110325 KB)      OKAY [ 27.705s]
Writing 'root'                               OKAY [ 0.991s]
Sending sparse 'root' 62/64 (114684 KB)      OKAY [ 28.856s]
Writing 'root'                               OKAY [ 0.897s]
Sending sparse 'root' 63/64 (109388 KB)      OKAY [ 27.435s]
Writing 'root'                               OKAY [ 0.860s]
Sending sparse 'root' 64/64 (24856 KB)       OKAY [ 6.347s]
Writing 'root'                               OKAY [ 0.205s]
Finished. Total time: 1929.583s

```

```

username: openkylin
password: openkylin

```

The network interfaces on openKylin is eth0 and eth1. The original content of /etc/network/interfaces are:

```

auto lo iface lo inet loopback
auto eth0 iface eth0 inet dhcp

```

Change it to:

```

auto lo iface lo inet loopback
# auto eth0 iface eth0 inet dhcp
iface eth0 inet static
    address 192.168.1.8
    netmask 255.255.255.0

```


SSH

37.1 Basics

Use only password:

```
ssh -o PubkeyAuthentication=no -o PreferredAuthentications=password example.com
```

37.1.1 Remote forward

```
# Run it on host2
ssh -R 6006:localhost:22 foo@host1
# We can access the port 6006 on host1, which is forwarded to host2's localhost:22
```

```
# host2's ~/.ssh/config Host host1
```

```
    Hostname ip_of_host1 User user_at_host1 RemoteForward 6006 localhost:22
```

37.1.2 Local forward

```
# Run it on host2
ssh -L 6006:localhost:22 foo@host1
# We can access the 6006 on host2, which is forwarded to host1's localhost:22
```

```
# host2's ~/.ssh/config Host host1
```

```
    Hostname ip_of_host1 User user_at_host1 LocalForward 6006 localhost:22 # 6006 is the port on this
    computer, i.e., host2 # Access 6006 on host2 will be forward to # localhost:22 on host1
```


38.1 Basics

The documentation is at <https://onnx.ai/onnx/intro/index.html>.

38.2 Python notes

See <https://onnx.ai/onnx/api/index.html#l-python-onnx-api>

```
pip install onnx
```

The data structure is defined in a protocol buffer file

<https://github.com/onnx/onnx/blob/main/onnx/onnx.proto>

See <https://protobuf.dev/reference/python/python-generated/> for Python APIs for protocol buffers.

38.2.1 TensorProto

It is defined in <https://github.com/onnx/onnx/blob/main/onnx/onnx.proto#L483>

<https://github.com/onnx/onnx/blob/main/onnx/mapping.py> defines a mapping to map datatypes from onnx to numpy.

Listing 1: `./code/tensor-proto.py`

```
1  #!/usr/bin/env python3
2  import onnx
3  import numpy as np
4
5
6  def test_dtype():
7      # the following line is not portable as it uses the internal implementation detail
8      # assert onnx.mapping.TENSOR_TYPE_TO_NP_TYPE[onnx.TensorProto.FLOAT] == np.float32
9      assert onnx.helper.tensor_dtype_to_np_dtype(onnx.TensorProto.FLOAT) == np.float32
10     assert onnx.helper.tensor_dtype_to_np_dtype(onnx.TensorProto.INT8) == np.int8
11
12     # We can convert a dtype to a string
13     assert (
14         onnx.helper.tensor_dtype_to_string(onnx.TensorProto.FLOAT)
15         == "TensorProto.FLOAT"
```

(continues on next page)

(continued from previous page)

```
16     )
17     assert onnx.TensorProto.DataType.Name(onnx.TensorProto.FLOAT) == "FLOAT"
18
19
20 def test_make_tensor():
21     v = np.array([[1, 2], [3, 4]])
22     t = onnx.helper.make_tensor(
23         name="my",
24         data_type=onnx.helper.np_dtype_to_tensor_dtype(v.dtype),
25         dims=v.shape,
26         vals=v,
27     )
28     print(t)
29
30     """
31     dims: 2
32     dims: 2
33     data_type: 7
34     int64_data: 1
35     int64_data: 2
36     int64_data: 3
37     int64_data: 4
38     name: "my"
39     """
40
41
42 def test_build_tensor_manually():
43     n = np.arange(5, dtype=np.float32)
44     t = onnx.TensorProto()
45     t.name = "my-tensor"
46     t.data_type = onnx.TensorProto.FLOAT
47     t.dims.extend(n.shape)
48     t.float_data.extend(n)
49     print(t)
50     """
51     dims: 5
52     data_type: 1
53     float_data: 0.0
54     float_data: 1.0
55     float_data: 2.0
56     float_data: 3.0
57     float_data: 4.0
58     name: "my-tensor"
59     """
60
61
62 def main():
63     test_dtype()
64     test_make_tensor()
65     test_build_tensor_manually()
66
67
```

(continues on next page)

(continued from previous page)

```

68 if __name__ == "__main__":
69     main()

```

38.2.2 TensorShapeProto

It is defined in <https://github.com/onnx/onnx/blob/main/onnx/onnx.proto#L661>.

Listing 2: ./code/tensor-shape-proto.py

```

1  #!/usr/bin/env python3
2  import onnx
3  import numpy as np
4
5
6  def main():
7      shape_proto = onnx.TensorShapeProto()
8
9      # shape_proto.dim is a list of messages, so we need to use add()
10     dim = shape_proto.dim.add()
11     dim.dim_value = 10
12
13     dim2 = shape_proto.dim.add()
14     dim2.dim_param = "N"
15     print(shape_proto)
16     """
17     dim {
18         dim_value: 10
19     }
20     dim {
21         dim_param: "N"
22     }
23     """
24
25     # Only one of dim_value and dim_param can be set
26     for d in shape_proto.dim:
27         which = d.WhichOneof("value")
28         if which == "dim_value":
29             print(d.dim_value) # 10
30         elif which == "dim_param":
31             print(d.dim_param) # N
32         else:
33             assert which is None
34
35
36 if __name__ == "__main__":
37     main()

```

One thing to note is that it contains a oneof field. Also note it can contain symbolic names for shapes.

```

message TensorShapeProto {
    message Dimension {

```

(continues on next page)

(continued from previous page)

```

oneof value {
    int64 dim_value = 1;
    string dim_param = 2;    // namespace Shape
};
// Standard denotation can optionally be used to denote tensor
// dimensions with standard semantic descriptions to ensure
// that operations are applied to the correct axis of a tensor.
// Refer to https://github.com/onnx/onnx/blob/main/docs/DimensionDenotation.md
↪ #denotation-definition
// for pre-defined dimension denotations.
optional string denotation = 3;
};
repeated Dimension dim = 1;
}

```

38.3 ValueInfoProto

38.4 onnxruntime

38.4.1 Install

Official installation doc: <https://onnxruntime.ai/docs/install/>.

One thing to note is that the so-called pre-built mobile packages contain only a subset of the operators, while the pre-built full packages contain all the operators.

See <https://onnxruntime.ai/docs/build/custom.html> for a custom build.

See <https://onnxruntime.ai/docs/build/inferencing.html> for building for inferencing.

38.4.2 Basics

Listing 3: ./code/Makefile

```

1 CXXFLAGS := -std=c++14
2 CXXFLAGS += -I /Users/fangjun/open-source/sherpa-onnx/onnxruntime-osx-x86_64-1.12.1/
↪ include
3 LDFLAGS := -L /Users/fangjun/open-source/sherpa-onnx/onnxruntime-osx-x86_64-1.12.1/lib
4 LDFLAGS += -l onnxruntime
5 LDFLAGS += -Wl,-rpath,/Users/fangjun/open-source/sherpa-onnx/onnxruntime-osx-x86_64-1.12.
↪ 1/lib
6
7 main: main.cc c-api-test.cc cpp-api-test.cc
8      $(CXX) $(CXXFLAGS) -o $@ $^ $(LDFLAGS)

```

Listing 4: ./code/main.cc

```

1 #include <iostream>
2

```

(continues on next page)

(continued from previous page)

```

3  #include "onnxruntime_cxx_api.h" // NOLINT
4
5  void TestCApi();
6  void TestCppApi();
7
8  int main() {
9      TestCApi();
10     TestCppApi();
11
12     std::cout << "ORT_API_VERSION: " << ORT_API_VERSION << "\n";
13     return 0;
14 }

```

Listing 5: ./code/c-api-test.cc

```

1  #include "onnxruntime_c_api.h" // NOLINT
2  #include <stdio.h>
3
4  static void TestOrtApiBase() {
5      const OrtApiBase *api_base = OrtGetApiBase();
6      fprintf(stderr, "GetVersionString(): %s\n", api_base->GetVersionString());
7  }
8
9  void TestCApi() {
10     TestOrtApiBase();
11     //
12 }

```

Listing 6: ./code/cpp-api-test.cc

```

1  #include "onnxruntime_cxx_api.h" // NOLINT
2  #include <assert.h>
3  #include <iostream>
4  #include <sstream>
5
6  static void TestOrtGetApi() {
7      const OrtApi &api = Ort::GetApi(); // it returns a const reference
8  }
9
10 static void PrintAvailableProviders() {
11     std::vector<std::string> providers = Ort::GetAvailableProviders();
12     std::ostringstream os;
13     os << "Available providers: ";
14     std::string sep = "";
15     for (const auto &p : providers) {
16         os << sep << p;
17         sep = ", ";
18     }
19     std::cout << os.str() << "\n";
20 }
21
22 static void TestCreateTensorFromBuffer() {

```

(continues on next page)

(continued from previous page)

```

23  std::vector<int32_t> v = {1, 2, 3, 4, 5, 6};
24  std::array<int64_t, 2> shape = {2, 3};
25  auto memory_info =
26      Ort::MemoryInfo::CreateCpu(OrtDeviceAllocator, OrtMemTypeDefault);
27
28  Ort::Value x = Ort::Value::CreateTensor<int32_t>(
29      memory_info, v.data(), v.size(), shape.data(), shape.size());
30
31  // memory is shared between x and v
32  int32_t *p = x.GetTensorMutableData<int32_t>();
33  p[0] = 10;
34  assert(v[0] == 10);
35
36  v[1] = 20;
37  assert(p[1] == 20);
38 }
39
40 static void TestCreateTensor() {
41     Ort::AllocatorWithDefaultOptions allocator;
42
43     std::array<int64_t, 2> shape = {2, 3};
44     auto memory_info =
45         Ort::MemoryInfo::CreateCpu(OrtDeviceAllocator, OrtMemTypeDefault);
46
47     Ort::Value x =
48         Ort::Value::CreateTensor<int32_t>(allocator, shape.data(), shape.size());
49     assert(x.IsTensor());
50     assert(x.HasValue());
51     Ort::TypeInfo type_info = x.GetTypeInfo();
52     auto tensor_type_and_shape_info = type_info.GetTensorTypeAndShapeInfo();
53     assert(tensor_type_and_shape_info.GetElementCount() == 2 * 3);
54     assert(tensor_type_and_shape_info.GetDimensionsCount() == 2);
55     std::vector<int64_t> x_shape = tensor_type_and_shape_info.GetShape();
56     assert(x_shape.size() == shape.size());
57     assert(x_shape[0] == shape[0]);
58     assert(x_shape[1] == shape[1]);
59
60     assert(tensor_type_and_shape_info.GetElementType() ==
61         ONNX_TENSOR_ELEMENT_DATA_TYPE_INT32);
62
63     Ort::ConstMemoryInfo memory_info = x.GetTensorMemoryInfo();
64     std::cout << "allocator name: " << memory_info.GetAllocatorName() << "\n";
65 }
66
67 void TestCppApi() {
68     TestOrtGetApi();
69     PrintAvailableProviders();
70     TestCreateTensorFromBuffer();
71     TestCreateTensor();
72 }

```

39.1 Basics

- Interop with Native Libraries
- Default marshalling for strings
- cbrumme's WebLog
- .NET Standard
- Preprocessor symbols
- Packaging Icon within the nupkg
- MSBuild reference for .NET SDK projects
- Visual Studio for Mac documentation
- DllImportAttribute Class
- A Closer Look at Platform Invoke
- Identifying Functions in DLLs
- DllImportAttribute.CallingConvention Field
- C# Coding Conventions
- IDisposable.Dispose Method
- Quickstart: Create and publish a package with the dotnet CLI
- Creating a NuGet package for a library with platform-specific API
- .NET RID Catalog

39.1.1 Use the commandline

```
# To activate the developer commandline
"C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\Build\
↪vcvars64.bat"
```

```
# it generates ./a.exe
csc ./a.cs
```

```
# it generates b.dll
csc /target:library b.cs

# it generates a.exe
csc /r:b.dll a.cs
```

```
dumbpin -exports sherpa-onnx-c-api.dll
```

If the name in the C++ function is Hello, we can use either:

```
[DllImport("dllname")]
void Hello();
```

or

```
[DllImport("dllname", EntryPoint="Hello")]
void HelloANewName();
```

39.1.2 Visual Studio for Mac

After installation, start it and click Visual Studio -> About Visual Studio -> Show details, it will show the locations:

- Runtime: /usr/local/share/dotnet/dotnet
- SDK: /usr/local/share/dotnet/sdk/7.0.203/Sdks

```
$ /usr/local/share/dotnet/dotnet

Usage: dotnet [options]
Usage: dotnet [path-to-application]

Options:
  -h|--help           Display help.
  --info              Display .NET information.
  --list-sdks         Display the installed SDKs.
  --list-runtimes     Display the installed runtimes.

path-to-application:
  The path to an application .dll file to execute.
```

39.1.3 global usings

Requires C# >= 10.0

Create a file, e.g., GlobalUsings.cs:

39.1.4 nullable types

```
Product? p = products[0];
string val;
if (p != null) {
    val = p.Name;
} else {
    val = "No value";
}
return View(new string[] { val });

string? val = products[0]?.Name;
if (val != null) {
    return View(new string[] { val });
}
return View(new string[] { "No Value" });

return View(new string[] { products[0]?.Name ?? "No Value" });

return View(new string[] { products[0]!.Name });
```

39.1.5 Dictionary

```
Dictionary<string, Product> products = new Dictionary<string, Product> {
    { "Kayak", new Product { Name = "Kayak", Price = 275M } },
    { "Lifejacket", new Product { Name = "Lifejacket", Price = 48.95M } }
};

Dictionary<string, Product> products = new Dictionary<string, Product> {
    ["Kayak"] = new Product { Name = "Kayak", Price = 275M },
    ["Lifejacket"] = new Product { Name = "Lifejacket", Price = 48.95M }
};

Dictionary<string, Product> products = new() {
    ["Kayak"] = new Product { Name = "Kayak", Price = 275M },
    ["Lifejacket"] = new Product { Name = "Lifejacket", Price = 48.95M }
};
```

39.1.6 foreach

```
public class ShoppingCart : IEnumerable<Product?> {
    public IEnumerable<Product?>? Products { get; set; }

    public IEnumerator<Product?> GetEnumerator() => Products?.GetEnumerator() ??
        Enumerable.Empty<Product?>().GetEnumerator();

    IEnumerator IEnumerable.GetEnumerator() => GetEnumerator();
}
```

39.1.7 my first page

```
@model MyMessage
@{
    Layout = null;
}
<!DOCTYPE html>

<html>
<head>
    <meta name="viewport" content="width=device-width"/>
    <title>Hello my page</title>
    <style>
    </style>
</head>
<body>
    <h1> please fill the form</h1>
    <form asp-action="Index" method="post">
        <div>
            <label asp-for="Id">Id</label>
            <input asp-for="Id"/>
        </div>
        <div>
            <label asp-for="Name">Name</label>
            <input asp-for="Name"/>
        </div>
        <button type="submit">submit</button>
        <a asp-action="Index">clear</a>
    </form>
</body>
</html>
```

39.1.8 EF

- Microsoft.EntityFrameworkCore.SQLite
- Microsoft.VisualStudio.Web.CodeGeneration.Design
- Microsoft.EntityFrameworkCore.SqlServer
- Microsoft.EntityFrameworkCore.Design
- Microsoft.EntityFrameworkCore.Tools

```
dotnet tool uninstall --global dotnet-aspnet-codegenerator
dotnet tool install --global dotnet-aspnet-codegenerator
dotnet tool uninstall --global dotnet-ef
dotnet tool install --global dotnet-ef

export PATH=$HOME/.dotnet/tools:$PATH
```

```
dotnet aspnet-codegenerator controller -name MoviesController -m Movie -dc MvcMovie.Data.
↪MvcMovieContext --relativeFolderPath Controllers --useDefaultLayout --
↪referenceScriptLibraries -sqlite
```



```

Building project ...
Finding the generator 'controller'...
Running the generator 'controller'...
--useSqlite|-sqlite option is obsolete now. Use --databaseProvider|-dbProvider instead.
↳ in the future.

Minimal hosting scenario!
Generating a new DbContext class 'MvcMovie.Data.MvcMovieContext'
Attempting to compile the application in memory with the added DbContext.
Attempting to figure out the EntityFramework metadata for the model and DbContext: 'Movie
↳ '

Using database provider 'Microsoft.EntityFrameworkCore.Sqlite'!

Added DbContext : '/Data/MvcMovieContext.cs'
Added Controller : '/Controllers/MoviesController.cs'.
Added View : /Views/Movies/Create.cshtml
Added View : /Views/Movies/Edit.cshtml
Added View : /Views/Movies/Details.cshtml
Added View : /Views/Movies/Delete.cshtml
Added View : /Views/Movies/Index.cshtml

```

```

diff --git a/MvcMovie/Program.cs b/MvcMovie/Program.cs
index 9fbb57d..b96f671 100644
--- a/MvcMovie/Program.cs
+++ b/MvcMovie/Program.cs
@@ -1,4 +1,9 @@
-var builder = WebApplication.CreateBuilder(args);
+using Microsoft.EntityFrameworkCore;
+using Microsoft.Extensions.DependencyInjection;
+using MvcMovie.Data;
+var builder = WebApplication.CreateBuilder(args);
+builder.Services.AddDbContext<MvcMovieContext>(options =>
+    options.UseSqlite(builder.Configuration.GetConnectionString("MvcMovieContext") ??
↳ throw new InvalidOperationException("Connection string 'MvcMovieContext' not found.
↳ "));

// Add services to the container.
builder.Services.AddControllersWithViews();

```

```

diff --git a/MvcMovie/appsettings.json b/MvcMovie/appsettings.json
index af0538f..d8b9276 100644
--- a/MvcMovie/appsettings.json
+++ b/MvcMovie/appsettings.json
@@ -1,10 +1,12 @@
-{"Logging": {
+{"Logging": {
  "LogLevel": {
    "Default": "Information",
    "Microsoft.AspNetCore": "Warning"
  }
}

```

(continues on next page)

(continued from previous page)

```

    },
-   "AllowedHosts": "*"
-}
-
+   "AllowedHosts": "*",
+   "ConnectionStrings": {
+       "MvcMovieContext": "Data Source=MvcMovieContext-95c663f1-d863-4557-a405-
→0d9cf818bb16.db"
+   }
+}

```

```
dotnet ef migrations add InitialCreate
```

39.2 nuget

vosk is using non-SDK style, while the latest .NET doc uses SDK style.

```
msbuild ./t.csproj -t:targetName -p:Configuration=Release -verbosity:detail
```

```
dotnet msbuild -property:TargetFramework=netcoreapp2.0 -preprocess:output.xml
```

39.2.1 References

- [Walkthrough: Use MSBuild](#)
- [Build items](#)
- [MSBuild items](#)
- [ItemGroup element \(MSBuild\)](#)
- [Use wildcards to specify items](#)
- [How to: Exclude files from the build](#)
- [MSBuild well-known item metadata](#)
- [How to: Select the files to build](#)
- [.NET project SDKs](#)

Select file

```

// All cs files in the current directory
<CSFile Include="*.cs"/>

// All files in the D drive
<VBFile Include="D:/**/*.*.vb"/>

// Include all cs files except the file DoNotBuild.cs
<ItemGroup>
  <CSFile Include="*.cs" Exclude="DoNotBuild.cs"/>
</ItemGroup>

```

(continues on next page)

(continued from previous page)

```

<Compile Include="*.cs" />
<Compile Include="*.res" Exclude="Form1.cs"> // note: Form1.cs is included in the first.
↳statement

<CSFile Include="Form1.cs"/>
<CSFile Include ="File1.cs;File2.cs"/>
<CSFile Include="*.cs"/>
<JPGFile Include="Images\**\*.jpg"/>

// All *.cs except Form2.cs and Form3.cs
<CSFile Include="*.cs" Exclude="Form2.cs;Form3.cs"/>

// all *.jpg files except the version2 directory
<JPGFile
  Include="Images\**\*.jpg"
  Exclude = "Images\**\Version2\*.jpg"/>

// Include only in Release build
<Compile
  Include="Formula.vb"
  Condition=" '$(Configuration)' == 'Release' " />

```

- To reference an item metadata: %(<ItemMetadataName>) or %(<ItemType.ItemMetadataName>)
- To reference a property, use \$(property)
- To reference an item, use @(item)

```

<Project xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <ItemGroup>
    <Stuff Include="One.cs" >
      <Display>false</Display>
    </Stuff>
    <Stuff Include="Two.cs">
      <Display>true</Display>
    </Stuff>
  </ItemGroup>
  <Target Name="Batching">
    <Message Text="@(<Stuff>)" Condition=" '%(Display)' == 'true' "/>
  </Target>
</Project>

// Stuff is an item, so we use @(Stuff)
// Display is an item metadata, so we use %(Display)
// @(<Stuff>) is "One.cs;Two.cs"

```

Some well-known item metadata %(FullPath), %(Filename)

39.3 dotnet

```
dotnet --list-sdks

# Create a directory and create a file global.json in it
# global.json contains the default sdk version to use
dotnet new globaljson --sdk-version 6.0.100 --output FirstProject

# Create an ASP.Net Core MVC web project, and put it inside the directory
# FirstProject.
dotnet new mvc --no-https --output FirstProject --framework net6.0

# Create a solution named FirstProject and put it inside ./FirstProject
dotnet new sln -o FirstProject

# Add the project "FirstProject" to the solution "FirstProject"
dotnet sln FirstProject add FirstProject

# list supported projects
dotnet new --list
```

39.4 Blazor

Create a page:

```
dotnet new razorcomponent -n Todo -o Pages
```

First TODO page:

```
public class TodoItem {
    public string? Title {get; set;}
    public bool IsDone {get; set;} = false;
}
```

```
@page "/todo"

<h3>Todo (@todos.Count(todo => !todo.IsDone))</h3>

<ul>
    @foreach (var todo in todos)
    {
        <li>
            <input type="checkbox" @bind="todo.IsDone" />
            <input @bind="todo.Title" />
        </li>
    }
</ul>

<input placeholder="something to do" @bind="newTodo" @bind:event="oninput" />
<button @onclick="AddItem">Add</button>
```

(continues on next page)

(continued from previous page)

```

<p> todo is @newTodo </p>

@code {
    private List<TodoItem> todos = new();
    private string? newTodo;

    private void AddItem() {
        todos.Add(new TodoItem { Title = newTodo});
        newTodo = string.Empty;
    }
}

```

On parameter set:

```

@page "/counter/{startingValue}"
@code {

    private int currentCount = 0;

    [Parameter]
    private int startingValue { get; set; } = 0

    protected override void OnParametersSet()
    {
        currentCount = startingValue;
        base.OnParametersSet();
    }
}

```

39.5 Hello

```
dotnet new console -h Hello
```

39.5.1 unsafe

```

using System.Diagnostics;
using System.Runtime.InteropServices;

/*
We have to add to ./Hello.csproj
  <AllowUnsafeBlocks>true</AllowUnsafeBlocks>
*/

```

(continues on next page)

(continued from previous page)

```
public class UnsafeTest {
    class MyStruct {
        public int A;
        public char B;
    }

    public static void TestPointer() {
        // for a primitive (unmanaged type)
        int a = 3;
        unsafe {
            int*p = &a;
            *p = 10;
            Debug.Assert(a == 10);
            a = 20;
            Debug.Assert(p[0] == 20); // like C/C++
        }

        // now for an array. We have to ensure the address of the array
        // is not moved by using fixed. This is also called a pin operation.
        unsafe {
            byte[] b = new byte[5];
            fixed(byte*p = &b[0]) {
                // p is a constant inside this block due to the usage of fixed.
                // That is, p is a fixed variable
                //
                // Note: We can also use (byte*p = b)
                p[0] = 10;
                Debug.Assert(b[0] == 10);

                b[1] = 20;
                Debug.Assert(*(p+1) == 20);
                Debug.Assert(p[1] == 20);
                Console.WriteLine($"address of p: 0x{(long)p:X}");
                Console.WriteLine($"address of p+1: 0x{(long)(p+1):X}");
            }
        }

        // test struct
        MyStruct s = new MyStruct {A=10, B='B'};
        unsafe {
            MyStruct* p = &s;
            Debug.Assert(p->A == 10); // similar to C/C++
            Debug.Assert(p->B == 'B');
            p->B = 'C';
            Debug.Assert(s.B == 'C');
        }

        // from stackalloc
        unsafe {
            int*p = stackalloc int[3];
            // the content is uninitialized
            p[0] = 10;
        }
    }
}
```

(continues on next page)

(continued from previous page)

```

    ++p;
    p[0] = 20;
    ++p;
    Debug.Assert(p[-2] == 10);
    Debug.Assert(p[-1] == 20);
}

IntPtr pp = Marshal.AllocHGlobal(sizeof(int));
unsafe {
    int* p = (int*)pp.ToPointer();
    p[0] = 10;
}
int i = Marshal.ReadInt32(pp);
Debug.Assert(i == 10);

Marshal.FreeHGlobal(pp);

}
}

```

References:

- [Unmanaged types \(C# reference\)](#)
- [fixed statement - pin a variable for pointer operations](#)
- [stackalloc](#)
- [Pointer related operators - take the address of variables, dereference storage locations, and access memory locations](#)
- [Unsafe code, pointer types, and function pointers](#)
- [23 Unsafe code](#)
- [Dos and Don'ts of stackalloc](#)

39.5.2 Span

```

using System.Diagnostics;
using System.Runtime.InteropServices;

class SpanTest {
    public static void Test() {
        TestByteArray();
        TestStackAlloc();
        TestGlobalAlloc();
    }
    public static void TestByteArray() {
        byte[] b = new byte[10];
        Span<byte> s = new Span<byte>(b);
        // s points to the memory holds by b
    }
}

```

(continues on next page)

(continued from previous page)

```
s[0] = 10;
Debug.Assert(b[0] == 10);

// s2 = b[3:5]
Span<byte> s2 = s.Slice(start:3, length: 2);
s2[0] = 12;
s2[1] = 13;
Debug.Assert(b[3] == 12);
Debug.Assert(b[4] == 13);
}

public static void TestStackAlloc() {
    Span<int> s = stackalloc int[10];
    s.Clear(); // set all contents to default(int);
    Debug.Assert(s.Length == 10);
}

public static void TestGlobalAlloc() {
    IntPtr p = Marshal.AllocHGlobal(sizeof(int) * 2);
    Span<int> s;
    unsafe {
        // p.ToPointer() returns void*, so it has to be used in void*
        s = new Span<int>(p.ToPointer(), 2);
        s[0] = 10;
        s[1] = 11;
    }

    int a = Marshal.ReadInt32(p);
    int b = Marshal.ReadInt32(p, sizeof(int));
    Debug.Assert(a == s[0]);
    Debug.Assert(b == s[1]);

    Marshal.FreeHGlobal(p);
}
}
```

References:

-

39.5.3 File

39.5.4 TODO

- stackalloc
- unsafe code
- unit test
- Marshal, read/write memory
- MemoryStream, FileStream
- FileReader, BinaryWriter

- directory exists, file exists

40.1 Install

```
python3 -m venv flask
source flask/bin/activate
pip3 install flask
```

40.2 Hello

```
#!/usr/bin/env python3

# flask --app hello run

from flask import Flask
from markupsafe import escape

print("__name__", __name__)

app = Flask(__name__)

@app.route("/")
def hello_world():
    app.logger.debug("A value for debugging")
    app.logger.warning("A warning occurred (%d apples)", 42)
    app.logger.error("An error occurred")

    return "<p>Hello world!</p>"

@app.route("/<name>")
def hello(name):
    return f"Hello, {escape(name)}!"

@app.route("/user/<username>")
def show_user_profile(username):
    # show the user profile for that user
```

(continues on next page)

(continued from previous page)

```

return f"User {escape(username)}"

# If a user uses /post/ab, it will show Not Found
@app.route("/post/<int:post_id>")
def show_post(post_id):
    print("type(post_id)", type(post_id)) # type(pos_id) <class 'int'>
    # show the post with the given id, the id is an integer
    return f"Post {post_id}"

@app.route("/path/<path:subpath>")
def show_subpath(subpath):
    print("type(subpath)", type(subpath)) # type(subpath) <class 'str'>
    # show the subpath after /path/
    return f"Subpath {escape(subpath)}"

```

```
flask --app hello run
```

The default filename is `app.py`. Since ours is `hello.py` we have to use `--app hello`.

To listen on all public IP addresses, use:

```
flask --app hello run --host=0.0.0.0
```

To enable debug mode, use `flask --app hello run --debug`

40.3 MySQL

Use the community edition.

40.3.1 Installation on macOS

See <https://dev.mysql.com/doc/mysql-macos-excerpt/5.7/en/macos-installation.html>

```

tar xvf mysql-8.0.33-macos13-x86_64.tar.gz

2023-05-28T15:14:16.627733Z 0 [System] [MY-013169] [Server] /Users/fangjun/software/
→mysql/mysql-8.0.33-macos13-x86_64/bin/mysqld (mysqld 8.0.33) initializing of server in_
→progress as process 11553
2023-05-28T15:14:16.629538Z 0 [Warning] [MY-010159] [Server] Setting lower_case_table_
→names=2 because file system for /Users/fangjun/software/mysql/mysql-8.0.33-macos13-x86_
→64/data/ is case insensitive
2023-05-28T15:14:16.633467Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has_
→started.
2023-05-28T15:14:16.800011Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has_
→ended.
2023-05-28T15:14:17.600930Z 6 [Note] [MY-010454] [Server] A temporary password is_
→generated for root@localhost: ;;Geh%7QmF%<
2023-05-28T15:14:18.316658Z 0 [System] [MY-013172] [Server] Received SHUTDOWN from user

```

(continues on next page)

(continued from previous page)

```
↪<via user signal>. Shutting down mysqld (Version: 8.0.33).

(py38) fangjuns-MacBook-Pro:mysql-8.0.33-macos13-x86_64 fangjun$ bin/mysql_ssl_rsa_setup_
↪--datadir ./data

(py38) fangjuns-MacBook-Pro:mysql-8.0.33-macos13-x86_64 fangjun$ ./bin/mysql -u root -p
# Enter the password generated in the above logs
# After logging, change the password to test

mysql> ALTER USER 'root'@localhost IDENTIFIED BY 'test';
mysql> EXIT
```

40.4 Node

40.4.1 Installation on macos

<https://nodejs.org/en/download>

```
wget https://nodejs.org/dist/v18.16.0/node-v18.16.0-darwin-x64.tar.gz
tar xvf node-v18.16.0-darwin-x64.tar.gz
export PATH=/Users/fangjun/software/node-v18.16.0-darwin-x64/bin:$PATH
```


41.1 Hello ARMv8

Starting from armv8, it has hardware floating point unit as well as NEON.

41.2 Registers

- Integer registers

<https://learn.microsoft.com/en-us/cpp/build/arm64-windows-abi-conventions?view=msvc-170#integer-registers>

41.3 Intrinsics

42.1 Add new disk

```
sudo fdisk -l  
sudo fdisk /dev/sdb  
sudo mkfs.ext4 /dev/sdb1  
sudo mkdir /mnt/sdb  
sudo mount /dev/sdb1 /mnt/sdb
```


43.1 Install

See <https://go.dev/doc/install>

43.1.1 macos

```
cd ~/software
wget https://go.dev/dl/go1.20.5.darwin-amd64.tar.gz
tar xvf go1.20.5.darwin-amd64.tar.gz
export PATH=~/software/go/bin:$PATH

go install golang.org/x/tools/cmd/godoc
```

43.2 Cross compile

```
go env
```

```
GO111MODULE=""
GOARCH="amd64"
GOBIN=""
GOCACHE="/Users/fangjun/Library/Caches/go-build"
GOENV="/Users/fangjun/Library/Application Support/go/env"
GOEXE=""
GOEXPERIMENT=""
GOFLAGS=""
GOHOSTARCH="amd64"
GOHOSTOS="darwin"
GOINSECURE=""
GOMODCACHE="/Users/fangjun/go/pkg/mod"
GONOPROXY=""
GONOSUMDB=""
GOOS="darwin"
GOPATH="/Users/fangjun/go"
GOPRIVATE=""
GOPROXY="https://proxy.golang.org,direct"
```

(continues on next page)

(continued from previous page)

```
GOROOT="/Users/fangjun/software/go"
GOSUMDB="sum.golang.org"
GOTMPDIR=""
GOTOOLDIR="/Users/fangjun/software/go/pkg/tool/darwin_amd64"
GOVCS=""
GOVERSION="go1.20.5"
GCCGO="gccgo"
GOAMD64="v1"
AR="ar"
CC="clang"
CXX="clang++"
CGO_ENABLED="1"
GOMOD="/dev/null"
GOWORK=""
CGO_CFLAGS="-O2 -g"
CGO_CPPFLAGS=""
CGO_CXXFLAGS="-O2 -g"
CGO_FFLAGS="-O2 -g"
CGO_LDFLAGS="-O2 -g"
PKG_CONFIG="pkg-config"
GOGCCFLAGS="-fPIC -arch x86_64 -m64 -pthread -fno-caret-diagnostics -Qunused-arguments -
↳ fmessage-length=0 -fdebug-prefix-map=/var/folders/fw/cg40_07j7cb18_p_ymf0zhkr0000gn/T/
↳ go-build1403459817=/tmp/go-build -gno-record-gcc-switches -fno-common"
```

43.2.1 Build for windows

```
GOOS=windows go build
```

43.2.2 Build for Linux

```
GOOS=linux go build
```

43.2.3 Build for macOS

```
GOOS=darwin go build
```

43.2.4 Build for Raspberry Pi (arm64)

```
GOOS=linux GOARCH=arm64 go build
```

43.2.5 Build for Raspberry Pi (arm32)

```
GOOS=linux GOARCH=arm go build
```

43.3 Basics

43.3.1 Environment variables

`go env VARIABLE_NAME` to view the value of the go environment variable with name `VARIABLE_NAME`.

```
go env GOPATH
go env GOARCH # example value: amd64, 386, arm
go env GOOS # example value: linux, darwin, windows
```

`go help environment`

43.3.2 Hello world

See <https://go.dev/doc/tutorial/getting-started>

```
go mod init example/hello
```

The above command will create a file `go.mod` in the current directory.

```
# content of go.mod
module example/hello

go 1.20
```

Note that `go mod init` does not create any directories. It only creates a file `go.mod` with the above content.

43.3.3 verbose build

```
go build -x -v
```

43.3.4 exported

For non-builtin functions and variables, if the name begin with an uppercase, then it is exported. Otherwise, it is not exported.

Note that for builtin functions, the above rule does not apply. For instance, both `print` and `println` are exported.

43.4 hello

Listing 1: ./code/hello2/ex.go

```
package main

import "fmt"

func main() {
    fmt.Println("hello world")
}
```

To run the program, use:

```
go run ex.go
```

For recent version of go, go build will throw the following error:

```
$ go build
go: cannot find main module, but found .git/config in /Users/fangjun/open-source/notes
    to create a module there, run:
    cd ../../../../.. && go mod init
```

But we can use go build ex.go, which works fine and will generate an executable ex.

To format the file in-place, use go fmt ex.go.

To see the help for a command, for instance the command for go fmt, use go help fmt.

To view the doc, run:

```
gofmt
# The above command is equivalent to
gofmt -http localhost:6060
# Then start your browser and visit http://localhost:6060
```

To install a tour of go, run

```
# see https://go.dev/tour/list
go install golang.org/x/website/tour@latest
```

```
import "fmt"
import "math/rand"
```

is equivalent to

```
import (
    "fmt"
    "math/rand"
)
```

Format specifier:

- %v, for any value
- %T, print the type

- %x, same as C/C++
- %S, for string and byte slice

```
# to view the help information of the function fmt.Printf
go doc fmt.Printf
```


44.1 Basics

<https://github.com/microsoft/onnxruntime/issues/15235#issuecomment-1487609280>

<https://medium.com/microsoftazure/build-and-deploy-fast-and-portable-speech-recognition-applications-with-onnx-runtime-and-whisper-1487609280>

44.1.1 transforms

```
from transformers import AutoProcessor, pipeline
model_path = "optimum/whisper-tiny.en"
processor = WhisperProcessor.from_pretrained(model_path)
```

https://huggingface.co/optimum/whisper-tiny.en/blob/main/preprocessor_config.json defines the preprocessor.

44.1.2 optimum

45.1 Scripts

45.1.1 Activate Visual studio

```
"C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\Build\  
↪vcvars64.bat"
```

45.1.2 View DLL dependencies

```
dumpbin /dependents .\onnxruntime_providers_cuda.dll
```

45.1.3 Remove folder

```
del /s /q .\*
```

45.2 Visual studio

Activate it in the command line:

```
"C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\Build\  
↪vcvars64.bat"
```

45.2.1 Property files

- Managing dependencies in Visual Studio C++ projects with property files
<https://blog.conan.io/2021/02/10/Dependencies-Visual-Studio-C++-property-files.html>
- .vcxproj and .props file structure
<https://learn.microsoft.com/en-us/cpp/build/reference/vcxproj-file-structure?view=msvc-160>

View symbols:

```
dumpbin -linkermember .\xxx.lib
```

46.1 Install

```
pip install aqtinstall
aqt list-qt mac desktop --arch 6.2.0
```

```
# It prints:
clang_64 wasm_32
```

```
aqt list-qt mac desktop --modules 6.2.0 clang_64
```

```
# It prints:
debug_info qt3d qt5compat qtcharts qtconnectivity qtdatavis3d qtimageformats qtlottie_
↳ qtmultimedia qtnetworkauth qtpositioning qtquick3d qtquicktimeline qtremoteobjects_
↳ qtscxml qtsensors qtserialbus qtserialport qtshadertools qtvirtualkeyboard_
↳ qtwebchannel qtwebengine qtwebsockets qtwebview
```

```
aqt install-qt mac desktop 6.2.0 clang_64 -m all
aqt install-qt mac desktop 6.2.0 -m multimedia
```

It creates `./6.2.0` and installs everything inside.

```
aqt install-qt mac desktop 6.2.0 --archives qtbase
```

46.2 Hello

WEBASSEMBLY

47.1 hello

<https://webassembly.github.io/wabt/demo/wat2wasm/index.html>

47.2 emsdk

47.2.1 Install

See https://emscripten.org/docs/getting_started/downloads.html

```
git clone https://github.com/emscripten-core/emsdk.git
cd emsdk
git pull
./emsdk install latest
./emsdk activate latest
source ./emsdk_env.sh
emcc -v
# See https://emscripten.org/docs/getting_started/downloads.html
```

47.2.2 hello.cc

```
emcc ./hello.cc
# It generates a.out.js, a.out.wasm

(py38) kuangfangjun:test-wasm$ ls
a.out.js  a.out.wasm  hello.cc

(py38) kuangfangjun:test-wasm$ file a.out.wasm
a.out.wasm: WebAssembly (wasm) binary module version 0x1 (MVP)
```

(continues on next page)

(continued from previous page)

```
(py38) kuangfangjun:test-wasm$ ls -lh a.out.wasm
-rwxr-xr-x 1 kuangfangjun root 156K Jul 27 15:10 a.out.wasm
```

```
emcc ./hello.cc -o hello.js
# It generates hello.js, hello.wasm
```

Create a file `test.html` containing:

```
<!doctype html>

<html>
  <head>
    <meta charset="utf-8">
    <title>Emscripten:Hello world</title>
  </head>

  <body>
    <script src="hello.js"></script>
  </body>
</html>
```

Start a python http web server `python3 -m http.server 6006`, the log is printed to the browser console.

Or use `emrun ./hello.html`.

Also, we can use node to run it: `node hello.js`.

Now delete `hello.html` and use

```
emcc hello.cc -o hello.html
# It will generate: hello.html, hello.js, hello.wasm
```

47.2.3 emscripten.h

https://emscripten.org/docs/api_reference/emscripten.h.html

48.1 Basics

The GitHub repo is <https://github.com/deezer/spleeter>

Configuration is at <https://github.com/deezer/spleeter/tree/master/spleeter/resources>.

DJANGO

49.1 Basics

```
python3 -m pip install Django
python3 -m django --version

django-admin startproject mysite

cd mysite
python3 manage.py runserver

python3 manage.py runserver 8080
python3 manage.py runserver 0.0.0.0:8000

python3 manage.py diffsettings

python3 manage.py startapp polls
```

https://github.com/django/django/blob/main/django/conf/global_settings.py

50.1 Basics

- <https://github.com/moonhighway/learning-react>

React developer tools: a browser extension

- <https://github.com/facebook/react>
- <https://chrome.google.com/webstore/detail/react-developer-tools/fmkadmapgofadopljbjfkapdkoienihi>

See *node*

51.1 vits

https://github.com/jaywalnut310/vits/blob/main/filelists/ljs_audio_text_test_filelist.txt.cleaned is processed by phonemizer

<https://github.com/FENRIR/MB-iSTFT-VITS2> has more detailed documentation.

<https://github.com/rhasspy/piper> is a C++ implementation of vits.

51.2 unicode

<https://github.com/avian2/unicode>

```
pip install unicode
```

```
(py38) fangjuns-MacBook-Pro:~ fangjun$ echo heiße |unicode  
heisse  
(py38) fangjuns-MacBook-Pro:~ fangjun$ echo 'hello hanzi nihao!' | unicode  
hello Ni Hao !
```

51.3 phonemizer

Its documentation is at <https://bootphon.github.io/phonemizer/>.

See <https://bootphon.github.io/phonemizer/install.html> for installation.

On macOS, it is:

```
brew install espeak  
pip install phonemizer
```

```
DUMMY1/LJ049-0022.wav|The Secret Service believed  
  
# cleaned
```

```
(py38) fangjuns-MacBook-Pro:~ fangjun$ echo "The Secret Service believed" | phonemize
```

51.4 books

Visit

- <https://libgen.is/>
- <http://gen.lib.rus.ec/>
- NeuralText-to-SpeechSynthesis, 2023

51.5 papers

- A Survey on Neural Speech Synthesis
<https://browse.arxiv.org/pdf/2106.15561.pdf>
- WaveNet: A generative model for raw audio
<https://arxiv.org/pdf/1609.03499.pdf>

51.6 pypinyin

- doc: https://pypinyin.readthedocs.io/zh_CN/master/
- pypi: <https://pypi.org/project/pypinyin/>

51.7 todo

- <https://github.com/PABannier/bark.cpp>, c++
- <https://github.com/rhasspy/piper>, c++
- <https://github.com/coqui-ai/TTS>
- <https://github.com/neonbjb/tortoise-tts>
- <https://github.com/gitmylo/bark-voice-cloning-HuBERT-quantizer>
<https://github.com/Plachtaa/VITS-fast-fine-tuning>
- https://huggingface.co/spaces/Plachta/VITS-Umamusume-voice-synthesizer/blob/main/ONNXVITS_to_onnx.py
- <https://huggingface.co/spaces/zomehwh/vits-models>
- <https://github.com/elevenlabs>
- <https://github.com/Plachtaa/VALL-E-X>
- <https://github.com/lifeiteng/vall-e>
- <https://github.com/alibaba-damo-academy/FunCodec>
- <https://github.com/alibaba-damo-academy/KAN-TTS>
- <https://github.com/CjangCjengh/MoeGoe>
- <https://github.com/weirdseed/vits-ncnn-convert-tool>
- https://colab.research.google.com/github/espnet/espnet_onnx/blob/master/demo/tts_onnx_demo.ipynb

- https://github.com/espnet/espnet_onnx
- <https://huggingface.co/spaces/Plachta/VITS-Umamusume-voice-synthesizer>
- <https://github.com/suno-ai/bark>
- <https://github.com/serp-ai/bark-with-voice-clone>
- <https://github.com/snakers4/silero-models#text-to-speech>

CHAPTER
FIFTYTWO

RUST

53.1 Basics

53.1.1 What is this GLIBCXX error?

See <https://omairmajid.com/posts/2020-07-08-what-is-glibcxx-error/>

It shows how to build and use glibc.

```
readelf --dyn-syms onnxruntime-linux-aarch64-1.16.3/lib/libonnxruntime.so | grep GLIBCXX
```

```
$ readelf --dyn-syms onnxruntime-linux-aarch64-1.16.3/lib/libonnxruntime.so | grep GLIBCXX
  6: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _Znam@GLIBCXX_3.4 (3)
  8: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZSo3putEc@GLIBCXX_3.4 (3)
 10: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZNSt6chrono3_V212system_
  @GLIBCXX_3.4.19 (5)
 11: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZNSt8__detail15_List_
 nod@GLIBCXX_3.4.15 (6)
 15: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZNSirsERS@GLIBCXX_3.4 (3)
 16: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZSt29_Rb_tree_insert_
 and@GLIBCXX_3.4 (3)
 19: 0000000000000000      0 FUNC    GLOBAL DEFAULT  UND _ZNSdD2Ev@GLIBCXX_3.4 (3)
```


54.1 Docker

See

- <https://docs.ros.org/en/iron/Installation/Ubuntu-Install-Debian.html>
- <https://docs.ros.org/en/iron/Tutorials.html>

The QA forum is <https://robotics.stackexchange.com/>

```
docker pull osrf/ros2:nightly
docker run -it osrf/ros2:nightly
docker run --rm -it --network host --name ros2 -v /host/path:/container/path osrf/
↪ros2:nightly /bin/bash
# connect to the container
docker exec -it ros2 /bin/bash
```

After starting the docker container,

```
# rolling is the name of the distro
#
# which ros2 prints: /opt/ros/rolling/bin/ros2

source /opt/ros/rolling/setup.bash

ros2 run demo_nodes_cpp talker

# In another terminal
source /opt/ros/rolling/setup.bash
ros2 run demo_nodes_py listener
```

```
printenv | grep -i ROS
```

prints the following:

```
ROS_VERSION=2
PKG_CONFIG_PATH=/opt/ros/rolling/lib/x86_64-linux-gnu/pkgconfig:/opt/ros/rolling/lib/
↪pkgconfig
ROS_PYTHON_VERSION=3
ROSDISTRO_INDEX_URL=https://raw.githubusercontent.com/osrf/docker_images/master/ros2/
↪nightly/nightly/index-v4.yaml
```

(continues on next page)

(continued from previous page)

```
AMENT_PREFIX_PATH=/opt/ros/rolling
CMAKE_PREFIX_PATH=/opt/ros/rolling
ROS_AUTOMATIC_DISCOVERY_RANGE=SUBNET
COLCON_PREFIX_PATH=/opt/ros/rolling
PYTHONPATH=/opt/ros/rolling/lib/python3.10/site-packages
LD_LIBRARY_PATH=/opt/ros/rolling/opt/rviz_ogre_vendor/lib:/opt/ros/rolling/opt/mimick_
↪ vendor/lib:/opt/ros/rolling/lib
PATH=/opt/ros/rolling/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
ROS_DISTRO=rolling
```

See <https://mirrors.tuna.tsinghua.edu.cn/help/ubuntu/> to change the apt source. Also see <https://mirrors.tuna.tsinghua.edu.cn/help/ros2/>.

Install the package `ros-rolling-turtlesim`:

```
sudo apt update
sudo apt install ros-rolling-turtlesim

ros2 pkg -h

# Check that it is installed
ros2 pkg executables turtlesim
```

It prints:

Now we can run

```
# ros2 run <package_name> <executable_name>

ros2 run turtlesim turtlesim_node
ros2 node list
ros2 topic list
ros2 service list
ros2 action list
ros2 node info <node_name>
```

Source code for `turtlesim`: https://github.com/ros/ros_tutorials/tree/iron/turtlesim

OPENFST

55.1 Basics

The latest version as of 2023.03.21 is v1.8.3

<https://github.com/csukuangfj/openfst-1.8.3/blob/master/BUILD.bazel> lists the build rules. The base lib is:

```
cc_library(  
    name = "base",  
    srcs = [  
        prefix_dir + "lib/compat.cc",  
        prefix_dir + "lib/flags.cc",  
    ],  
    hdrs = [  
        prefix_dir + "include/fst/compat.h",  
        prefix_dir + "include/fst/flags.h",  
        prefix_dir + "include/fst/icu.h",  
        prefix_dir + "include/fst/lock.h",  
        prefix_dir + "include/fst/log.h",  
        prefix_dir + "include/fst/windows_defs.inc",  
    ],  
    defines = select({  
        ":has_absl": ["OPENFST_HAS_ABSL=1"],  
        "//conditions:default": [],  
    }),  
    includes = [prefix_dir + "include"],  
    deps = select({  
        ":has_absl": ["@com_google_absl//absl/synchronization"],  
        "//conditions:default": [],  
    }),  
)
```

55.1.1 compat.h

```
#if defined(__GNUC__) || defined(__clang__)
#define OPENFST_DEPRECATED(message) __attribute__((deprecated(message)))
#elif defined(_MSC_VER)
#define OPENFST_DEPRECATED(message) [[deprecated(message)]]
#else
#define OPENFST_DEPRECATED(message)
#endif
```