

Phạm Duy Hưng – 20225850

Assignment 1:

start:

```
li $s1,9 #i=9
```

```
li $s2,2 #j=2
```

```
slt $t0,$s2,$s1 #j<i
```

```
bne $t0,$zero,else # branch to else if j<i
```

```
addi $t1,$t1,1 # then part: x=x+1
```

```
addi $t3,$zero,1 # z=1
```

```
j endif # skip "else" part
```

else: addi \$t2,\$t2,-1 # begin else part: y=y-1

```
add $t3,$t3,$t3 # z=2*z
```

endif:

The screenshot displays a MIPS simulator interface. The left pane shows the assembly code with the following instructions:

Bkpt	Address	Code	Basic	Source
	0x00400000	0x24110009	addiu \$t7,\$0,0x0000...	3: li \$s1,9
	0x00400004	0x24120002	addiu \$t8,\$0,0x0000...	4: li \$s2,2
	0x00400008	0x0251402a	slt \$t0,\$t8,\$t7	5: slt \$t0,\$s2,\$s1 # j<i
	0x0040000c	0x15000003	bne \$t0,\$0,0x00000003	6: bne \$t0,\$zero,else # branc...
	0x00400010	0x21290001	addi \$t1,\$t1,1	7: addi \$t1,\$t1,1 # then part...
	0x00400014	0x200b0001	addi \$t3,\$zero,1	8: addi \$t3,\$zero,1 # z=1
	0x00400018	0x08100009	j 0x00400024	9: j endif # skip "else" part
	0x0040001c	0x214affff	addi \$t2,\$t2,-1	10: else: addi \$t2,\$t2,-1 # begin el...
	0x00400020	0x016b5820	add \$t3,\$t3,\$t3	11: add \$t3,\$t3,\$t3 # z=2*z

The right pane shows the registers. Register \$t3 is highlighted with a green background and contains the value 0x00000000. Register \$s4 is labeled "saved temporary (preserved across call)".

Name	Number	Value
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000001
\$t1	9	0x00000000
\$t2	10	0xffffffff
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000000
\$s2	18	0x00000002
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$s9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7ffffeffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400024

Assignment 2:

.data

A: .word 9,8,7,6,5,4,3,2,1

.text

li \$s1, -1 #i=-1

la \$s2, A #s2 stores the address of array

li \$s3, 9 #number elements of A

li \$s4, 1 #step

li \$s5, 0 #sum

loop:

add \$s1,\$s1,\$s4 #i=i+step

add \$t1,\$s1,\$s1 #t1=2*s1

add \$t1,\$t1,\$t1 #t1=4*s1

add \$t1,\$t1,\$s2 #t1 store the address of A[i]

lw \$t0,0(\$t1) #load value of A[i] in \$t0

add \$s5,\$s5,\$t0 #sum=sum+A[i]

bne \$s1,\$s3,loop #if i != n, go to loop

The screenshot displays a MIPS simulator interface. On the left, the 'Text Segment' window shows assembly code with addresses and comments. The 'Data Segment' window shows memory addresses and values. On the right, the 'Registers' window lists registers \$at through \$ra and pc, with their current values. The \$s5 register is highlighted in green, showing a value of 0x00000024.

Name	Number	Value
\$at	1	0x10010000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x10010024
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000005
\$s2	18	0x10010000
\$s3	19	0x00000009
\$s4	20	0x00000001
\$s5	21	0x00000024
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$s9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$p0	28	0x10008000
\$p1	29	0x7ffffc00
\$ra	30	0x00000000
\$pc	31	0x00400034

Assignment 3:

```
.data
test: .word 2

.text
li $s2, 1 #a=1
li $s3, 2 #b=2

la $s0, test #load the address of test variable
lw $s1, 0($s0) #load the value of test to register $t1
li $t0, 0 #load value for test case
li $t1, 1
li $t2, 2
beq $s1, $t0, case_0
beq $s1, $t1, case_1
beq $s1, $t2, case_2
j default
case_0: addi $s2, $s2, 1 #a=a+1
j continue
case_1: sub $s2, $s2, $t1 #a=a-1
j continue
case_2: add $s3, $s3, $s3 #b=2*b
j continue
default:
continue:
```

Execute

Text Segment

Bkpt	Address	Code	Basic	Source
	0x0040001c	0x240a0002	addiu \$t0,\$t0,0x0000...	l1: li \$t2, 2
	0x00400020	0x12280003	beq \$t7,\$t8,0x00000003	l2: beq \$s1, \$t0, case_0
	0x00400024	0x12290004	beq \$t7,\$t9,0x00000004	l3: beq \$s1, \$t1, case_1
	0x00400028	0x122a0005	beq \$t7,\$t10,0x00000005	l4: beq \$s1, \$t2, case_2
	0x0040002c	0x08100012	j 0x00400048	l5: j default
	0x00400030	0x22520001	addi \$t8,\$t8,0x0000...	l6: case_0: addi \$s2, \$s2, 1 #a=a+1
	0x00400034	0x08100012	j 0x00400048	l7: j continue
	0x00400038	0x02499022	sub \$t8,\$t8,\$t9	l8: case_1: sub \$s2, \$s2, \$t1 #a=a-1
	0x0040003c	0x08100012	j 0x00400048	l9: j continue
	0x00400040	0x02739820	add \$t9,\$t9,\$t9	l0: case_2: add \$s3, \$s3, \$s3 #b=2*b
	0x00400044	0x08100012	j 0x00400048	l1: j continue

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x00000002	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x10010020	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x10010040	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x10010060	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x10010080	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x100100a0	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x100100c0	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x100100e0	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...
0x10010100	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...	0x00000000	0x0000...

0x10010000 (.data) Hexadecimal Addresses Hexadecimal Values ASCII

Mars Messages Run IO

Registers			Coproc 1	Coproc 0
Name	Number	Value		
\$at	1	0x10010000		
\$v0	2	0x00000000		
\$v1	3	0x00000000		
\$a0	4	0x00000000		
\$a1	5	0x00000000		
\$a2	6	0x00000000		
\$a3	7	0x00000000		
\$t0	8	0x00000000		
\$t1	9	0x00000001		
\$t2	10	0x00000002		
\$t3	11	0x00000000		
\$t4	12	0x00000000		
\$t5	13	0x00000000		
\$t6	14	0x00000000		
\$t7	15	0x00000000		
\$s0	16	0x10010000		
\$s1	17	0x00000002		
\$s2	18	0x00000001		
\$s3	19	0x00000004		
\$s4	20	0x00000000		
\$s5	21	0x00000000		
\$s6	22	0x00000000		
\$s7	23	0x00000000		
\$s8	24	0x00000000		
\$s9	25	0x00000000		
\$k0	26	0x00000000		
\$k1	27	0x00000000		
\$gp	28	0x10008000		
\$sp	29	0x7fffffc		
\$fp	30	0x00000000		
\$ra	31	0x00000000		
pc		0x00400048		

Assignment 4:

a.

start:

```
li $s1,9 #i=9
```

```
li $s2,2 #j=2
```

```
slt $t0,$s2,$s1 # j<i
```

```
bne $t0,$zero,else # branch to else if j<i
```

```
addi $t1,$t1,1 # then part: x=x+1
```

```
addi $t3,$zero,1 # z=1
```

```
j endif # skip "else" part
```

else: addi \$t2,\$t2,-1 # begin else part: y=y-1

```
add $t3,$t3,$t3 # z=2*z
```

endif:

The screenshot displays a MIPS simulator interface. The left pane shows the assembly code with the following instructions:

```
3: li $s1,9 #i=9
4: li $s2,2 #j=2
5: slt $t0,$s2,$s1 # j<i
6: bne $t0,$zero,else # branch to else if j<i
7: addi $t1,$t1,1 # then part: x=x+1
8: addi $t3,$zero,1 # z=1
9: j endif # skip "else" part
10: else: addi $t2,$t2,-1 # begin else part: y=y-1
11: add $t3,$t3,$t3 # z=2*z
```

The right pane shows the registers. The \$t3 register is highlighted in green, indicating its current value is 0x00000000.

Name	Number	Value
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000001
\$t1	9	0x00000000
\$t2	10	0xffffffff
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000009
\$s2	18	0x00000002
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$s9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10008000
\$sp	29	0x7fffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400024

b.

start:

```
li $s1,9 #i=9
```

```
li $s2,2 #j=2
```

```
bge $t0,$zero,else # branch to else if i>=j
```

```
addi $t1,$t1,1 # then part: x=x+1
```

```
addi $t3,$zero,1 # z=1
```

```
j endif # skip "else" part
```

else: addi \$t2,\$t2,-1 # begin else part: y=y-1

```
add $t3,$t3,$t3 # z=2*z
```

endif:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

1026

1027

1028

1029

1030

1031

1032

1033

1034

1035

1036

1037

1038

1039

1040

1041

1042

1043

1044

1045

1046

1047

1048

1049

1050

1051

1052

1053

1054

1055

1056

1057

1058

1059

1060

1061

1062

1063

1064

1065

1066

1067

1068

1069

1070

1071

1072

1073

1074

1075

1076

1077

1078

1079

1080

1081

1082

1083

1084

1085

1086

1087

1088

1089

1090

1091

1092

1093

1094

1095

1096

1097

1098

1099

1100

1101

1102

1103

1104

1105

1106

1107

1108

1109

1110

1111

1112

1113

1114

1115

1116

1117

1118

1119

1120

1121

1122

1123

1124

1125

1126

1127

1128

1129

1130

1131

1132

1133

1134

1135

1136

1137

1138

1139

1140

1141

1142

1143

1144

1145

1146

1147

1148

1149

1150

1151

1152

1153

1154

1155

1156

1157

1158

1159

1160

1161

1162

1163

1164

1165

1166

1167

1168

1169

1170

1171

1172

1173

1174

1175

1176

1177

1178

1179

1180

1181

1182

1183

1184

1185

1186

1187

1188

1189

1190

1191

1192

1193

1194

1195

1196

1197

1198

1199

1200

1201

1202

1203

1204

1205

1206

1207

1208

1209

1210

1211

1212

1213

1214

1215

1216

1217

1218

1219

1220

1221

1222

1223

1224

1225

1226

1227

1228

1229

1230

1231

1232

1233

1234

1235

1236

1237

1238

1239

1240

1241

1242

1243

1244

1245

1246

1247

1248

1249

1250

1251

1252

1253

1254

1255

1256

1257

1258

1259

1260

1261

1262

1263

1264

1265

1266

1267

1268

1269

1270

1271

1272

1273

1274

1275

1276

1277

1278

1279

1280

1281

1282

1283

1284

1285

1286

1287

1288

1289

1290

1291

1292

1293

1294

1295

1296

1297

1298

1299

1300

1301

1302

1303

1304

1305

1306

1307

1308

1309

1310

1311

1312

1313

1314

1315

1316

1317

1318

1319

1320

1321

1322

1323

1324

1325

1326

1327

1328

1329

1330

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

1350

1351

1352

1353

1354

1355

1356

1357

1358

1359

1360

1361

1362

1363

1364

1365

1366

1367

1368

1369

1370

1371

1372

1373

1374

1375

1376

1377

1378

1379

1380

1381

1382

1383

1384

1385

1386

1387

1388

1389

1390

1391

1392

1393

1394

1395

1396

1397

1398

1399

1400

1401

1402

1403

1404

1405

1406

1407

1408

1409

1410

1411

1412

1413

1414

1415

1416

1417

1418

1419

1420

1421

1422

1423

1424

1425

1426

1427

1428

1429

1430

1431

1432

1433

1434

1435

1436

1437

1438

1439

1440

1441

1442

1443

1444

1445

1446

1447

1448

1449

1450

1451

1452

1453

1454

1455

1456

1457

1458

1459

1460

1461

1462

1463

1464

1465

1466</

c.

start:

```
li $s1,9 #i=9
```

```
li $s2,2 #j=2
```

```
add $s3,$s1,$s2 # x = i+j
```

```
ble $s3,$zero,else # branch to else if x<=0
```

```
addi $t1,$t1,1 # then part: x=x+1
```

```
addi $t3,$zero,1 # z=1
```

```
j endif # skip "else" part
```

else: addi \$t2,\$t2,-1 # begin else part: y=y-1

```
add $t3,$t3,$t3 # z=2*z
```

endif:

Edit Execute					Registers Coproc 1 Coproc 0		
Text Segment					Name	Number	Value
0x00400000	0x24110009	addiu \$t7,\$0,0x0000...	2:	li \$a1,9 #i=9	\$zero	0	0x00000000
0x00400004	0x24120002	addiu \$t8,\$0,0x0000...	3:	li \$a2,2 #j=2	\$a1	1	0x00000001
0x00400008	0x02329820	addi \$t9,\$t7,\$t8	4:	add \$s3,\$a1,\$a2 # x = i+j	\$a2	2	0x00000002
0x0040000c	0x0013082a	slti \$t0,\$t9	5:	ble \$s3,\$zero,else # branch to else if x<=0	\$a3	3	0x00000003
0x00400010	0x10200003	beq \$t0,\$t0,0x00000003			\$a4	4	0x00000004
0x00400014	0x21290001	addi \$t1,\$t1,1	6:	addi \$t1,\$t1,1 # then part: x=x+1	\$a5	5	0x00000005
0x00400018	0x20080001	addi \$t2,\$t2,-1	7:	addi \$t2,\$t2,-1 # begin else part: y=y-1	\$a6	6	0x00000006
0x0040001c	0x010000a7	0x00400025	8:	j endif # skip "else" part	\$a7	7	0x00000007
0x00400020	0x21400001	addi \$t3,\$t3,1	9:	addi \$t3,\$t3,1 # then part: z=z+1	\$a8	8	0x00000008
0x00400024	0x010000a7	0x00400025	10:	add \$t3,\$t3,\$t3 # z=2*z	\$a9	9	0x00000009
					\$a10	10	0x0000000a
					\$a11	11	0x0000000b
					\$a12	12	0x0000000c
					\$a13	13	0x0000000d
					\$a14	14	0x0000000e
					\$a15	15	0x0000000f
					\$a16	16	0x00000010
					\$a17	17	0x00000011
					\$a18	18	0x00000012
					\$a19	19	0x00000013
					\$a20	20	0x00000014
					\$a21	21	0x00000015
					\$a22	22	0x00000016
					\$a23	23	0x00000017
					\$a24	24	0x00000018
					\$a25	25	0x00000019
					\$a26	26	0x0000001a
					\$a27	27	0x0000001b
					\$a28	28	0x0000001c
					\$a29	29	0x0000001d
					\$a30	30	0x0000001e
					\$a31	31	0x0000001f
					\$a32	32	0x00000020
					\$a33	33	0x00000021
					\$a34	34	0x00000022
					\$a35	35	0x00000023
					\$a36	36	0x00000024
					\$a37	37	0x00000025
					\$a38	38	0x00000026
					\$a39	39	0x00000027
					\$a40	40	0x00000028
					\$a41	41	0x00000029
					\$a42	42	0x0000002a
					\$a43	43	0x0000002b
					\$a44	44	0x0000002c
					\$a45	45	0x0000002d
					\$a46	46	0x0000002e
					\$a47	47	0x0000002f
					\$a48	48	0x00000030
					\$a49	49	0x00000031
					\$a50	50	0x00000032
					\$a51	51	0x00000033
					\$a52	52	0x00000034
					\$a53	53	0x00000035
					\$a54	54	0x00000036
					\$a55	55	0x00000037
					\$a56	56	0x00000038
					\$a57	57	0x00000039
					\$a58	58	0x0000003a
					\$a59	59	0x0000003b
					\$a60	60	0x0000003c
					\$a61	61	0x0000003d
					\$a62	62	0x0000003e
					\$a63	63	0x0000003f
					\$a64	64	0x00000040
					\$a65	65	0x00000041
					\$a66	66	0x00000042
					\$a67	67	0x00000043
					\$a68	68	0x00000044
					\$a69	69	0x00000045
					\$a70	70	0x00000046
					\$a71	71	0x00000047
					\$a72	72	0x00000048
					\$a73	73	0x00000049
					\$a74	74	0x0000004a
					\$a75	75	0x0000004b
					\$a76	76	0x0000004c
					\$a77	77	0x0000004d
					\$a78	78	0x0000004e
					\$a79	79	0x0000004f
					\$a80	80	0x00000050
					\$a81	81	0x00000051
					\$a82	82	0x00000052
					\$a83	83	0x00000053
					\$a84	84	0x00000054
					\$a85	85	0x00000055
					\$a86	86	0x00000056
					\$a87	87	0x00000057
					\$a88	88	0x00000058
					\$a89	89	0x00000059
					\$a90	90	0x0000005a
					\$a91	91	0x0000005b
					\$a92	92	0x0000005c
					\$a93	93	0x0000005d
					\$a94	94	0x0000005e
					\$a95	95	0x0000005f
					\$a96	96	0x00000060
					\$a97	97	0x00000061
					\$a98	98	0x00000062
					\$a99	99	0x00000063
					\$a100	100	0x00000064
					\$a101	101	0x00000065
					\$a102	102	0x00000066
					\$a103	103	0x00000067
					\$a104	104	0x00000068
					\$a105	105	0x00000069
					\$a106	106	0x0000006a
					\$a107	107	0x0000006b
					\$a108	108	0x0000006c
					\$a109	109	0x0000006d
					\$a110	110	0x0000006e
					\$a111	111	0x0000006f
					\$a112	112	0x00000070
					\$a113	113	0x00000071
					\$a114	114	0x00000072
					\$a115	115	0x00000073
					\$a116	116	0x00000074
					\$a117	117	0x00000075
					\$a118	118	0x00000076
					\$a119	119	0x00000077
					\$a120	120	0x00000078
					\$a121	121	0x00000079
					\$a122	122	0x0000007a
					\$a123	123	0x0000007b
					\$a124	124	0x0000007c
					\$a125	125	0x0000007d
					\$a126	126	0x0000007e
					\$a127	127	0x0000007f
					\$a128	128	0x00000080
					\$a129	129	0x00000081
					\$a130	130	0x00000082
					\$a131	131	0x00000083
					\$a132	132	0x00000084
					\$a133	133	0x00000085
					\$a134	134	0x00000086
					\$a135	135	0x00000087
					\$a136	136	0x00000088
					\$a137	137	0x00000089
					\$a138	138	0x0000008a
					\$a139	139	0x0000008b
					\$a140	140	0x0000008c
					\$a141	141	0x0000008d

endif:

[illegible]

Assignment 5:

a.

.data

A: .word 1, 2, 5, 8, 9, 10, 3, 4, 6

.text

li \$s1, -1 #i = -1

la \$s2, A #s2 stores the address of array

li \$s3, 9 #number of elements of A

li \$s4, 1 #step

li \$s5, 0 #sum

loop:

add \$s1,\$s1,\$s4 #i=i+step

add \$t1,\$s1,\$s1 #t1=2*s1

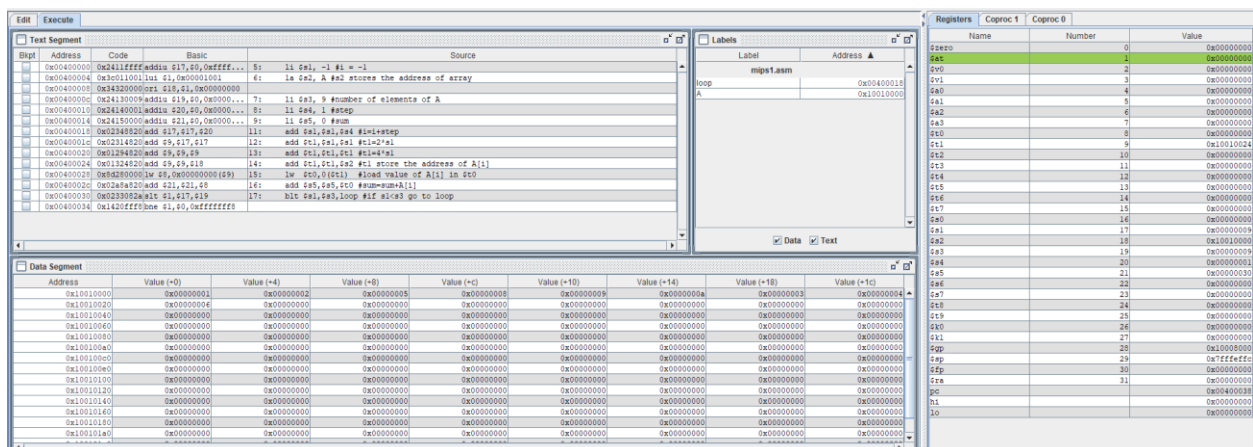
add \$t1,\$t1,\$t1 #t1=4*s1

add \$t1,\$t1,\$s2 #t1 store the address of A[i]

lw \$t0,0(\$t1) #load value of A[i] in \$t0

add \$s5,\$s5,\$t0 #sum=sum+A[i]

blt \$s1,\$s3,loop #if s1<s3 go to **loop**



b.

.data

A: .word 1, 2, 5, 8, 9, 10, 3, 4, 6

.text

li \$s1, -1 #i = -1

la \$s2, A #s2 stores the address of array

li \$s3, 9 #number of elements of A

li \$s4, 1 #step

li \$s5, 0 #sum

loop:

add \$s1,\$s1,\$s4 #i=i+step

add \$t1,\$s1,\$s1 #t1=2*s1

add \$t1,\$t1,\$t1 #t1=4*s1

add \$t1,\$t1,\$s2 #t1 store the address of A[i]

lw \$t0,0(\$t1) #load value of A[i] in \$t0

add \$s5,\$s5,\$t0 #sum=sum+A[i]

ble \$s1,\$s3,loop #if s1<=s3 go to loop

Edit		Execute		Registers		Coproc 1	Coproc 0
Text Segment				Name	Number	Value	
Bkpt	Address	Code	Basic	Source			
	0x00400000	0x241fffff	addiu \$t1,\$t0,0xffff...	5: li \$s1, -1 #i = -1	\$zero	0	0x00000000
	0x00400004	0x3c011001	lui \$t1,0x00000101	6: la \$s2, A #s2 stores the address of array	\$a0	1	0x00000001
	0x00400008	0x34320000	ori \$t1,\$t1,0x00000000		\$v0	2	0x00000000
	0x0040000c	0x24130009	addiu \$t1,\$t0,0x0000...	7: li \$s3, 9 #number of elements of A	\$v1	3	0x00000000
	0x00400010	0x24140001	addiu \$t1,\$t0,0x0000...	8: li \$s4, 1 #step	\$a0	4	0x00000000
	0x00400014	0x24150000	addiu \$t1,\$t0,0x0000...	9: li \$s5, 0 #sum	\$a1	5	0x00000000
	0x00400018	0x02348820	add \$t1,\$t1,\$t2	11: add \$t1,\$t1,\$t2 #i=i+step	\$a0	6	0x00000000
	0x0040001c	0x02314820	add \$t1,\$t1,\$t2	12: add \$t1,\$t1,\$t2 #i=i+step	\$t0	7	0x00000000
	0x00400020	0x02348820	add \$t1,\$t1,\$t2	13: add \$t1,\$t1,\$t2 #i=i+step	\$t1	8	0x00000000
	0x00400024	0x02314820	add \$t1,\$t1,\$t2	14: add \$t1,\$t1,\$t2 #i=i+step	\$t2	9	0x00000000
	0x00400028	0x02348820	add \$t1,\$t1,\$t2	15: add \$t1,\$t1,\$t2 #i=i+step	\$t3	10	0x00000000
	0x0040002c	0x02314820	add \$t1,\$t1,\$t2	16: add \$t1,\$t1,\$t2 #i=i+step	\$t4	11	0x00000000
	0x00400030	0x02348820	add \$t1,\$t1,\$t2	17: add \$t1,\$t1,\$t2 #i=i+step	\$t5	12	0x00000000
	0x00400034	0x02314820	add \$t1,\$t1,\$t2	18: add \$t1,\$t1,\$t2 #i=i+step	\$t6	13	0x00000000
	0x00400038	0x02348820	add \$t1,\$t1,\$t2	19: add \$t1,\$t1,\$t2 #i=i+step	\$t7	14	0x00000000
	0x0040003c	0x02314820	add \$t1,\$t1,\$t2	20: add \$t1,\$t1,\$t2 #i=i+step	\$t8	15	0x00000000
	0x00400040	0x02348820	add \$t1,\$t1,\$t2	21: add \$t1,\$t1,\$t2 #i=i+step	\$t9	16	0x00000000
	0x00400044	0x02314820	add \$t1,\$t1,\$t2	22: add \$t1,\$t1,\$t2 #i=i+step	\$a2	17	0x0000000a
	0x00400048	0x02348820	add \$t1,\$t1,\$t2	23: add \$t1,\$t1,\$t2 #i=i+step	\$a3	18	0x00000009
	0x0040004c	0x02314820	add \$t1,\$t1,\$t2	24: add \$t1,\$t1,\$t2 #i=i+step	\$a4	19	0x00000001
	0x00400050	0x02348820	add \$t1,\$t1,\$t2	25: add \$t1,\$t1,\$t2 #i=i+step	\$a5	20	0x00000000
	0x00400054	0x02314820	add \$t1,\$t1,\$t2	26: add \$t1,\$t1,\$t2 #i=i+step	\$a6	21	0x00000000
	0x00400058	0x02348820	add \$t1,\$t1,\$t2	27: add \$t1,\$t1,\$t2 #i=i+step	\$a7	22	0x00000000
	0x0040005c	0x02314820	add \$t1,\$t1,\$t2	28: add \$t1,\$t1,\$t2 #i=i+step	\$t7	23	0x00000000
	0x00400060	0x02348820	add \$t1,\$t1,\$t2	29: add \$t1,\$t1,\$t2 #i=i+step	\$t8	24	0x00000000
	0x00400064	0x02314820	add \$t1,\$t1,\$t2	30: add \$t1,\$t1,\$t2 #i=i+step	\$t9	25	0x00000000
	0x00400068	0x02348820	add \$t1,\$t1,\$t2	31: add \$t1,\$t1,\$t2 #i=i+step	\$k0	26	0x00000000
	0x0040006c	0x02314820	add \$t1,\$t1,\$t2	32: add \$t1,\$t1,\$t2 #i=i+step	\$k1	27	0x00000000
	0x00400070	0x02348820	add \$t1,\$t1,\$t2	33: add \$t1,\$t1,\$t2 #i=i+step	\$k2	28	0x00000000
	0x00400074	0x02314820	add \$t1,\$t1,\$t2	34: add \$t1,\$t1,\$t2 #i=i+step	\$k3	29	0x00000000
	0x00400078	0x02348820	add \$t1,\$t1,\$t2	35: add \$t1,\$t1,\$t2 #i=i+step	\$k4	30	0x00000000
	0x0040007c	0x02314820	add \$t1,\$t1,\$t2	36: add \$t1,\$t1,\$t2 #i=i+step	\$k5	31	0x00000000
	0x00400080	0x02348820	add \$t1,\$t1,\$t2	37: add \$t1,\$t1,\$t2 #i=i+step	\$k6	32	0x00000000
	0x00400084	0x02314820	add \$t1,\$t1,\$t2	38: add \$t1,\$t1,\$t2 #i=i+step	\$k7	33	0x00000000
	0x00400088	0x02348820	add \$t1,\$t1,\$t2	39: add \$t1,\$t1,\$t2 #i=i+step	\$k8	34	0x00000000
	0x0040008c	0x02314820	add \$t1,\$t1,\$t2	40: add \$t1,\$t1,\$t2 #i=i+step	\$k9	35	0x00000000
	0x00400090	0x02348820	add \$t1,\$t1,\$t2	41: add \$t1,\$t1,\$t2 #i=i+step	\$k10	36	0x00000000
	0x00400094	0x02314820	add \$t1,\$t1,\$t2	42: add \$t1,\$t1,\$t2 #i=i+step	\$k11	37	0x00000000
	0x00400098	0x02348820	add \$t1,\$t1,\$t2	43: add \$t1,\$t1,\$t2 #i=i+step	\$k12	38	0x00000000
	0x0040009c	0x02314820	add \$t1,\$t1,\$t2	44: add \$t1,\$t1,\$t2 #i=i+step	\$k13	39	0x00000000
	0x004000a0	0x02348820	add \$t1,\$t1,\$t2	45: add \$t1,\$t1,\$t2 #i=i+step	\$k14	40	0x00000000
	0x004000a4	0x02314820	add \$t1,\$t1,\$t2	46: add \$t1,\$t1,\$t2 #i=i+step	\$k15	41	0x00000000
	0x004000a8	0x02348820	add \$t1,\$t1,\$t2	47: add \$t1,\$t1,\$t2 #i=i+step	\$k16	42	0x00000000
	0x004000ac	0x02314820	add \$t1,\$t1,\$t2	48: add \$t1,\$t1,\$t2 #i=i+step	\$k17	43	0x00000000
	0x004000b0	0x02348820	add \$t1,\$t1,\$t2	49: add \$t1,\$t1,\$t2 #i=i+step	\$k18	44	0x00000000
	0x004000b4	0x02314820	add \$t1,\$t1,\$t2	50: add \$t1,\$t1,\$t2 #i=i+step	\$k19	45	0x00000000
	0x004000b8	0x02348820	add \$t1,\$t1,\$t2	51: add \$t1,\$t1,\$t2 #i=i+step	\$k20	46	0x00000000
	0x004000bc	0x02314820	add \$t1,\$t1,\$t2	52: add \$t1,\$t1,\$t2 #i=i+step	\$k21	47	0x00000000
	0x004000c0	0x02348820	add \$t1,\$t1,\$t2	53: add \$t1,\$t1,\$t2 #i=i+step	\$k22	48	0x00000000
	0x004000c4	0x02314820	add \$t1,\$t1,\$t2	54: add \$t1,\$t1,\$t2 #i=i+step	\$k23	49	0x00000000
	0x004000c8	0x02348820	add \$t1,\$t1,\$t2	55: add \$t1,\$t1,\$t2 #i=i+step	\$k24	50	0x00000000
	0x004000cc	0x02314820	add \$t1,\$t1,\$t2	56: add \$t1,\$t1,\$t2 #i=i+step	\$k25	51	0x00000000
	0x004000d0	0x02348820	add \$t1,\$t1,\$t2	57: add \$t1,\$t1,\$t2 #i=i+step	\$k26	52	0x00000000
	0x004000d4	0x02314820	add \$t1,\$t1,\$t2	58: add \$t1,\$t1,\$t2 #i=i+step	\$k27	53	0x00000000
	0x004000d8	0x02348820	add \$t1,\$t1,\$t2	59: add \$t1,\$t1,\$t2 #i=i+step	\$k28	54	0x00000000
	0x004000dc	0x02314820	add \$t1,\$t1,\$t2	60: add \$t1,\$t1,\$t2 #i=i+step	\$k29	55	0x00000000
	0x004000e0	0x02348820	add \$t1,\$t1,\$t2	61: add \$t1,\$t1,\$t2 #i=i+step	\$k30	56	0x00000000
	0x004000e4	0x02314820	add \$t1,\$t1,\$t2	62: add \$t1,\$t1,\$t2 #i=i+step	\$k31	57	0x00000000
	0x004000e8	0x02348820	add \$t1,\$t1,\$t2	63: add \$t1,\$t1,\$t2 #i=i+step	\$k32	58	0x00000000
	0x004000ec	0x02314820	add \$t1,\$t1,\$t2	64: add \$t1,\$t1,\$t2 #i=i+step	\$k33	59	0x00000000
	0x004000f0	0x02348820	add \$t1,\$t1,\$t2	65: add \$t1,\$t1,\$t2 #i=i+step	\$k34	60	0x00000000
	0x004000f4	0x02314820	add \$t1,\$t1,\$t2	66: add \$t1,\$t1,\$t2 #i=i+step	\$k35	61	0x00000000
	0x004000f8	0x02348820	add \$t1,\$t1,\$t2	67: add \$t1,\$t1,\$t2 #i=i+step	\$k36	62	0x00000000
	0x004000fc	0x02314820	add \$t1,\$t1,\$t2	68: add \$t1,\$t1,\$t2 #i=i+step	\$k37	63	0x00000000
	0x00400100	0x02348820	add \$t1,\$t1,\$t2	69: add \$t1,\$t1,\$t2 #i=i+step	\$k38	64	0x00000000
	0x00400104	0x02314820	add \$t1,\$t1,\$t2	70: add \$t1,\$t1,\$t2 #i=i+step	\$k39	65	0x00000000
	0x00400108	0x02348820	add \$t1,\$t1,\$t2	71: add \$t1,\$t1,\$t2 #i=i+step	\$k40	66	0x00000000
	0x0040010c	0x02314820	add \$t1,\$t1,\$t2	72: add \$t1,\$t1,\$t2 #i=i+step	\$k41	67	0x00000000
	0x00400110	0x02348820	add \$t1,\$t1,\$t2	73: add \$t1,\$t1,\$t2 #i=i+step	\$k42	68	0x00000000
	0x00400114	0x02314820	add \$t1,\$t1,\$t2	74: add \$t1,\$t1,\$t2 #i=i+step	\$k43	69	0x00000000
	0x00400118	0x02348820	add \$t1,\$t1,\$t2	75: add \$t1,\$t1,\$t2 #i=i+step	\$k44	70	0x00000000
	0x0040011c	0x02314820	add \$t1,\$t1,\$t2	76: add \$t1,\$t1,\$t2 #i=i+step	\$k45	71	0x00000000
	0x00400120	0x02348820	add \$t1,\$t1,\$t2	77: add \$t1,\$t1,\$t2 #i=i+step	\$k46	72	0x00000000
	0x00400124	0x02314820	add \$t1,\$t1,\$t2	78: add \$t1,\$t1,\$t2 #i=i+step	\$k47	73	0x00000000
	0x00400128	0x02348820	add \$t1,\$t1,\$t2	79: add \$t1,\$t1,\$t2 #i=i+step	\$k48	74	0x00000000
	0x0040012c	0x02314820	add \$t1,\$t1,\$t2	80: add \$t1,\$t1,\$t2 #i=i+step	\$k49	75	0x00000000
	0x00400130	0x02348820	add \$t1,\$t1,\$t2	81: add \$t1,\$t1,\$t2 #i=i+step	\$k50	76	0x00000000
	0x00400134	0x02314820	add \$t1,\$t1,\$t2	82: add \$t1,\$t1,\$t2 #i=i+step	\$k51	77	0x00000000
	0x00400138	0x02348820	add \$t1,\$t1,\$t2	83: add \$t1,\$t1,\$t2 #i=i+step	\$k52	78	0x00000000
	0x0040013c	0x02314820	add \$t1,\$t1,\$t2	84: add \$t1,\$t1,\$t2 #i=i+step	\$k53	79	0x00000000
	0x00400140	0x02348820	add \$t1,\$t1,\$t2	85: add \$t1,\$t1,\$t2 #i=i+step	\$k54	80	0x00000000
	0x00400144	0x02314820	add \$t1,\$t1,\$t2	86: add \$t1,\$t1,\$t2 #i=i+step	\$k55	81	0x00000000
	0x00400148	0x02348820	add \$t1,\$t1,\$t2	87: add \$t1,\$t1,\$t2 #i=i+step	\$k56	82	0x00000000
	0x0040014c	0x02314820	add \$t1,\$t1,\$t2	88: add \$t1,\$t1,\$t2 #i=i+step	\$k57	83	0x00000000
	0x00400150	0x02348820	add \$t1,\$t1,\$t2	89: add \$t1,\$t1,\$t2 #i=i+step	\$k58	84	0x00000000
	0x00400154	0x02314820	add \$t1,\$t1,\$t2	90: add \$t1,\$t1,\$t2 #i=i+step	\$k59	85	0x00000000
	0x00400158	0x02348820	add \$t1,\$t1,\$t2	91: add \$t1,\$t1,\$t2 #i=i+step	\$k60	86	0x00000000
	0x0040015c	0x02314820	add \$t1,\$t1,\$t2	92: add \$t1,\$t1,\$t2 #i=i+step	\$k61	87	0x00000000
	0x00400160	0x02348820	add \$t1,\$t1,\$t2	93: add \$t1,\$t1,\$t2 #i=i+step	\$k62	88	0x00000000
	0x00400164	0x02314820	add \$t1,\$t1,\$t2	94: add \$t1,\$t1,\$t2 #i=i+step	\$k63	89	0x00000000
	0x00400168	0x02348820	add \$t1,\$t1,\$t2	95: add \$t1,\$t1,\$t2 #i=i+step	\$k64	90	0x00000000
	0x0040016c	0x02314820	add \$t1,\$t1,\$t2	96: add \$t1,\$t1,\$t2 #i=i+step	\$k65	91	0x00000000
	0x00400170	0x02348820	add \$t1,\$t1,\$t2	97: add \$t1,\$t1,\$t2 #i=i+step	\$k66	92	0x00000000
	0x00400174	0x02314820	add \$t1,\$t1,\$t2	98: add \$t1,\$t1,\$t2 #i=i+step	\$k67	93	0x00000000
	0x00400178	0x02348820	add \$t1,\$t1,\$t2	99: add \$t1,\$t1,\$t2 #i=i+step	\$k68	94	0x00000000
	0x0040017c	0x02314820	add \$t1,\$t1,\$t2	100: add \$t1,\$t1,\$t2 #i=i+step	\$k69	95	0x00000000
	0x00400180	0x02348820	add \$t1,\$t1,\$t2	101: add \$t1,\$t1,\$t2 #i=i+step	\$k70	96	0x00000000
	0x00400184	0x02314820	add \$t1,\$t1,\$t2	102: add \$t1,\$t1,\$t2 #i=i+step	\$k71	97	0x00000000
	0x00400188	0x02348820	add \$t1,\$t1,\$t2	103: add \$t1,\$t1,\$t2 #i=i+step	\$k72	98	0x00000000
	0x0040018c	0x02314820	add \$t1,\$t1,\$t2	104: add \$t1,\$t1,\$t2 #i=i+step	\$k73	99	0x00000000
	0x00400190	0x02348820	add \$t1,\$t1,\$t2	105: add \$t1,\$t1,\$t2 #i=i+step	\$k74	100	0x00000000
	0x00400194	0x02314820	add \$t1,\$t1,\$t2	106: add \$t1,\$t1,\$t2 #i=i+step	\$k75	101	0x00000000
	0x00400198	0x02348820	add \$t1,\$t1,\$t2	107: add \$t1,\$t1,\$t2 #i=i+step	\$k76	102	0x00000000
	0x0040019c	0x02314820	add \$t1,\$t1,\$t2	108: add \$t1,\$t1,\$t2 #i=i+step	\$k77	103	0x00000000
	0x004001a0	0x02348820	add \$t1,\$t1,\$t2	109: add \$t1,\$t1,\$t2 #i=i+step	\$k78	104	0x00000000
	0x004001a4	0x02314820	add \$t1,\$t1,\$t2	110: add \$t1,\$t1,\$t2 #i=i+step	\$k79	105	0x00000000
	0x004001a8	0x02348820	add \$t1,\$t1,\$t2	111: add \$t1,\$t1,\$t2 #i=i+step	\$k80	106	0x00000000
	0x004001ac	0x02314820	add \$t1,\$t1,\$t2	112: add \$t1,\$t1,\$t2 #i=i+step	\$k81	107	0x00000000
	0x004001b0	0x02348820	add \$t1,\$t1,\$t2	113: add \$t1,\$t1,\$t2 #i=i+step	\$k82	108	0x00000000
	0x004001b4	0x02314820	add \$t1,\$t1,\$t2	114: add \$t1,\$t1,\$t2 #i=i+step	\$k83	109	0x00000000
	0x004001b8	0x02348820	add \$t1,\$t1,\$t2	115: add \$t1,\$t1,\$t2 #i=i+step	\$k84	110	0x00000000
	0x004001bc	0x02314820	add \$t1,\$t1,\$t2	116: add \$t1,\$t1,\$t2 #i=i+step	\$k85	111	0x00000000
	0x004001c0	0x02348820	add \$t1,\$t1,\$t2	117: add \$t1,\$t1,\$t2 #i=i+step	\$k86	112	0x00000000
	0x004001c4	0x02314820	add \$t1,\$t1,\$t2	118: add \$t1,\$t1,\$t2 #i=i+step	\$k87	113	0x00000000
	0x004001c8	0x02348820	add \$t1,\$t1,\$t2	119: add \$t1,\$t1,\$t2 #i=i+step	\$k88	114	0x00000000
	0x004001cc	0x02314820	add \$t1,\$t1,\$t2	120: add \$t1,\$t1,\$t2 #i=i+step	\$k89	115	0x0000

c.

.data

A: .word 1, 2, 5, 8, 9, 10, 3, 4, 6

.text

li \$s1, -1 #i = -1

la \$s2, A #s2 stores the address of array

li \$s3, 9 #number of elements of A

li \$s4, 1 #step

li \$s5, 0 #sum

loop:

add \$s1,\$s1,\$s4 #i=i+step

add \$t1,\$s1,\$s1 #t1=2*s1

add \$t1,\$t1,\$t1 #t1=4*s1

add \$t1,\$t1,\$s2 #t1 store the address of A[i]

lw \$t0,0(\$t1) #load value of A[i] in \$t0

add \$s5,\$s5,\$t0 #sum=sum+A[i]

bge \$s5,\$zero,loop #if s5>=0 go to loop

The screenshot displays a MIPS simulator interface. The top toolbar includes icons for file operations, execution, and debugging. Below the toolbar, the 'Text Segment' tab is active, showing assembly code with columns for Bit, Address, Code, Basic, and Source. The code implements a loop to calculate the sum of elements in array A. The 'Registers' tab on the right shows the state of registers \$zero through \$31, with their respective values in hexadecimal. The 'Data Segment' tab at the bottom shows memory addresses and their corresponding values, which are all zero in this view.

Bit	Address	Code	Basic	Source
	0x00400000	0x2411ffff	addiu \$t1,\$s0,0xffff...	li \$s1, -1 #i = -1
	0x00400004	0x3c110013	la \$s2, A	la \$s2, A #s2 stores the address of array
	0x00400008	0x34320009	li \$s3, 9	li \$s3, 9 #number of elements of A
	0x0040000c	0x24130009	addiu \$t1,\$s1,0	li \$s4, 1 #step
	0x00400010	0x24140000	addiu \$s5,\$s0,0	li \$s5, 0 #sum
	0x00400014	0x24150000	addiu \$t1,\$s2,0	
	0x00400018	0x03480020	add \$t1,\$t1,\$s4	add \$t1,\$t1,\$s4 #i=i+step
	0x0040001c	0x03480020	add \$t1,\$t1,\$t1	add \$t1,\$t1,\$t1 #t1=4*s1
	0x00400020	0x03480020	add \$t1,\$t1,\$s2	add \$t1,\$t1,\$s2 #t1 store the address of A[i]
	0x00400024	0x03480020	lw \$t0,0(\$t1)	lw \$t0,0(\$t1) #load value of A[i] in \$t0
	0x00400028	0x03480020	add \$s5,\$s5,\$t0	add \$s5,\$s5,\$t0 #sum=sum+A[i]
	0x0040002c	0x03480020	bge \$s5,\$zero,loop	bge \$s5,\$zero,loop #if s5>=0 go to loop
	0x00400030	0x03480020		
	0x00400034	0x03480020		

Registers	Coproc 1	Coproc 0	Name	Number	Value
			\$zero	0	0x00000000
			\$at	1	0x00000000
			\$v0	2	0x00000000
			\$t1	3	0x00000000
			\$a0	4	0x00000000
			\$a1	5	0x00000000
			\$a2	6	0x00000000
			\$a3	7	0x00000000
			\$t0	8	0x00000000
			\$t1	9	0x00000000
			\$t2	10	0x00000000
			\$t3	11	0x00000000
			\$t4	12	0x00000000
			\$t5	13	0x00000000
			\$t6	14	0x00000000
			\$t7	15	0x00000000
			\$t8	16	0x00000000
			\$t9	17	0x00000000
			\$t10	18	0x00000000
			\$t11	19	0x00000000
			\$t12	20	0x00000000
			\$t13	21	0x00000000
			\$t14	22	0x00000000
			\$t15	23	0x00000000
			\$t16	24	0x00000000
			\$t17	25	0x00000000
			\$t18	26	0x00000000
			\$t19	27	0x00000000
			\$t20	28	0x00000000
			\$t21	29	0x00000000
			\$t22	30	0x00000000
			\$t23	31	0x00000000
			\$t24	32	0x00000000
			\$t25	33	0x00000000
			\$t26	34	0x00000000
			\$t27	35	0x00000000
			\$t28	36	0x00000000
			\$t29	37	0x00000000
			\$t30	38	0x00000000
			\$t31	39	0x00000000
			\$t32	40	0x00000000
			\$t33	41	0x00000000
			\$t34	42	0x00000000
			\$t35	43	0x00000000
			\$t36	44	0x00000000
			\$t37	45	0x00000000
			\$t38	46	0x00000000
			\$t39	47	0x00000000
			\$t40	48	0x00000000
			\$t41	49	0x00000000
			\$t42	50	0x00000000
			\$t43	51	0x00000000
			\$t44	52	0x00000000
			\$t45	53	0x00000000
			\$t46	54	0x00000000
			\$t47	55	0x00000000
			\$t48	56	0x00000000
			\$t49	57	0x00000000
			\$t50	58	0x00000000
			\$t51	59	0x00000000
			\$t52	60	0x00000000
			\$t53	61	0x00000000
			\$t54	62	0x00000000
			\$t55	63	0x00000000
			\$t56	64	0x00000000
			\$t57	65	0x00000000
			\$t58	66	0x00000000
			\$t59	67	0x00000000
			\$t60	68	0x00000000
			\$t61	69	0x00000000
			\$t62	70	0x00000000
			\$t63	71	0x00000000
			\$t64	72	0x00000000
			\$t65	73	0x00000000
			\$t66	74	0x00000000
			\$t67	75	0x00000000
			\$t68	76	0x00000000
			\$t69	77	0x00000000
			\$t70	78	0x00000000
			\$t71	79	0x00000000
			\$t72	80	0x00000000
			\$t73	81	0x00000000
			\$t74	82	0x00000000
			\$t75	83	0x00000000
			\$t76	84	0x00000000
			\$t77	85	0x00000000
			\$t78	86	0x00000000
			\$t79	87	0x00000000
			\$t80	88	0x00000000
			\$t81	89	0x00000000
			\$t82	90	0x00000000
			\$t83	91	0x00000000
			\$t84	92	0x00000000
			\$t85	93	0x00000000
			\$t86	94	0x00000000
			\$t87	95	0x00000000
			\$t88	96	0x00000000
			\$t89	97	0x00000000
			\$t90	98	0x00000000
			\$t91	99	0x00000000
			\$t92	100	0x00000000
			\$t93	101	0x00000000
			\$t94	102	0x00000000
			\$t95	103	0x00000000
			\$t96	104	0x00000000
			\$t97	105	0x00000000
			\$t98	106	0x00000000
			\$t99	107	0x00000000
			\$t100	108	0x00000000
			\$t101	109	0x00000000
			\$t102	110	0x00000000
			\$t103	111	0x00000000
			\$t104	112	0x00000000
			\$t105	113	0x00000000
			\$t106	114	0x00000000
			\$t107	115	0x00000000
			\$t108	116	0x00000000
			\$t109	117	0x00000000
			\$t110	118	0x00000000
			\$t111	119	0x00000000
			\$t112	120	0x00000000
			\$t113	121	0x00000000
			\$t114	122	0x00000000
			\$t115	123	0x00000000
			\$t116	124	0x00000000
			\$t117	125	0x00000000
			\$t118	126	0x00000000
			\$t119	127	0x00000000
			\$t120	128	0x00000000
			\$t121	129	0x00000000
			\$t122	130	0x00000000
			\$t123	131	0x00000000
			\$t124	132	0x00000000
			\$t125	133	0x00000000
			\$t126	134	0x00000000
			\$t127	135	0x00000000
			\$t128	136	0x00000000
			\$t129	137	0x00000000
			\$t130	138	0x00000000
			\$t131	139	0x00000000
			\$t132	140	0x00000000
			\$t133	141	0x00000000
			\$t134	142	0x00000000
			\$t135	143	0x00000000
			\$t136	144	0x00000000
			\$t137	145	0x00000000
			\$t138	146	0x00000000
			\$t139	147	0x00000000
			\$t140	148	0x00000000
			\$t141	149	0x00000000
			\$t142	150	0x00000000
			\$t143	151	0x00000000
			\$t144	152	0x00000000
			\$t145	153	0x00000000
			\$t146	154	0x00000000
			\$t147	155	0x00000000
			\$t148	156	0x00000000
			\$t149	157	0x00000000
			\$t150	158	0x00000000
			\$t151	159	0x00000000
			\$t152	160	0x00000000
			\$t153	161	0x00000000
			\$t154	162	0x00000000
			\$t155	163	0x00000000
			\$t156	164	0x00000000
			\$t157	165	0x00000000
			\$t158	166	0x00000000
			\$t159	167	0x00000000
			\$t160	168	0x00000000
			\$t161	169	0x00000000
			\$t162	170	0x00000000
			\$t163	171	0x00000000
			\$t164	172	0x00000000
			\$t165	173	0x00000000
			\$t166	174	0x00000000
			\$t167	175	0x00000000
			\$t168	176	0x00000000
			\$t169	177	0x00000000
			\$t170	178	0x00000000
			\$t171	179	0x00000000
			\$t172	180	0x00000000
			\$t173	181	0x00000000
			\$t174	182	0x00000000
			\$t175	183	0x00000000
			\$t176	184	0x00000000
			\$t177	185	0x00000000
			\$t178	186	0x00000000
			\$t179	187	0x00000000
			\$t180	188	0x00000000
			\$t181	189	0x00000000
			\$t182	190	0x00000000
			\$t183	191	0x00000000
			\$t184	192	0x00000000
			\$t185	193	0x00000000
			\$t186	194	0x00000000
			\$t187	195	0x00000000
			\$t188	196	0x00000000
			\$t189	197	0x00000000
			\$t190	198	0x00000000
			\$t191	199	0x00000000
			\$t192	200	0x00000000
			\$t193	201	0x00000000
			\$t194	202	0x00000000
			\$t195	203	0x00000000
			\$t196	204	0x00000000
			\$t197	205	0x00000000
			\$t198	206	0x00000000
			\$t199	207	0x00000000
			\$t200	208	0x00000000
			\$t201	209	0x00000000
			\$t202	210	0x00000000
			\$t203	211	0x00000000
			\$t204	212	0x00000000
			\$t205	213	0x00000000
			\$t206	214	0x00000000
			\$t207	215	0x00000000
			\$t208	216	0x00000000
			\$t209	217	0x00000000
			\$t210	218	0x00000000
			\$t211	219	0x00000000
			\$t212	220	0x00000000
			\$t213	221	

d.

.data

A: .word 1, 2, 5, 8, 9, 10, 3, 4, 6

.text

li \$s1, -1 #i = -1

la \$s2, A #s2 stores the address of array

li \$s3, 9 #number of elements of A

li \$s4, 1 #step

li \$s5, 0 #sum

loop:

add \$s1,\$s1,\$s4 #i=i+step

add \$t1,\$s1,\$s1 #t1=2*s1

add \$t1,\$t1,\$t1 #t1=4*s1

add \$t1,\$t1,\$s2 #t1 store the address of A[i]

lw \$t0,0(\$t1) #load value of A[i] in \$t0

add \$s5,\$s5,\$t0 #sum=sum+A[i]

bnez \$t0,loop #if s5>=0 go to loop

The screenshot displays a MIPS simulator interface. The main window is divided into two panes: 'Text Segment' and 'Data Segment'.

Text Segment: This pane shows the assembly code being executed. The code is as follows:

```
li $s1, -1 #i = -1
la $s2, A #s2 stores the address of array
li $s3, 9 #number of elements of A
li $s4, 1 #step
li $s5, 0 #sum

loop:
add $s1,$s1,$s4 #i=i+step
add $t1,$s1,$s1 #t1=2*s1
add $t1,$t1,$t1 #t1=4*s1
add $t1,$t1,$s2 #t1 store the address of A[i]
lw $t0,0($t1) #load value of A[i] in $t0
add $s5,$s5,$t0 #sum=sum+A[i]
bnez $t0,loop #if s5>=0 go to loop
```

Data Segment: This pane shows the memory layout of the data segment. It contains a table with columns for Address, Value (+0), Value (+4), Value (+8), Value (+C), Value (+10), Value (+14), and Value (+18). The data is as follows:

Address	Value (+0)	Value (+4)	Value (+8)	Value (+C)	Value (+10)	Value (+14)	Value (+18)
0x10010000	0x00000001	0x00000002	0x00000005	0x00000008	0x00000009	0x0000000a	0x00000003
0x10010020	0x00000004	0x00000006	0x00000007	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

Registers: The right pane shows the state of the MIPS registers. The registers are listed in a table with columns for Name, Number, and Value. The registers are as follows:

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$t8	25	0x00000000
\$t9	26	0x00000000
\$k0	27	0x00000000
\$k1	28	0x00000000
\$k2	29	0x00000000
\$k3	30	0x00000000
\$k4	31	0x00000000
\$k5	32	0x00000000
\$k6	33	0x00000000
\$k7	34	0x00000000
\$k8	35	0x00000000
\$k9	36	0x00000000
\$k10	37	0x00000000
\$k11	38	0x00000000
\$k12	39	0x00000000
\$k13	40	0x00000000
\$k14	41	0x00000000
\$k15	42	0x00000000
\$k16	43	0x00000000
\$k17	44	0x00000000
\$k18	45	0x00000000
\$k19	46	0x00000000
\$k20	47	0x00000000
\$k21	48	0x00000000
\$k22	49	0x00000000
\$k23	50	0x00000000
\$k24	51	0x00000000
\$k25	52	0x00000000
\$k26	53	0x00000000
\$k27	54	0x00000000
\$k28	55	0x00000000
\$k29	56	0x00000000
\$k30	57	0x00000000
\$k31	58	0x00000000
\$k32	59	0x00000000
\$k33	60	0x00000000
\$k34	61	0x00000000
\$k35	62	0x00000000
\$k36	63	0x00000000
\$k37	64	0x00000000
\$k38	65	0x00000000
\$k39	66	0x00000000
\$k40	67	0x00000000
\$k41	68	0x00000000
\$k42	69	0x00000000
\$k43	70	0x00000000
\$k44	71	0x00000000
\$k45	72	0x00000000
\$k46	73	0x00000000
\$k47	74	0x00000000
\$k48	75	0x00000000
\$k49	76	0x00000000
\$k50	77	0x00000000
\$k51	78	0x00000000
\$k52	79	0x00000000
\$k53	80	0x00000000
\$k54	81	0x00000000
\$k55	82	0x00000000
\$k56	83	0x00000000
\$k57	84	0x00000000
\$k58	85	0x00000000
\$k59	86	0x00000000
\$k60	87	0x00000000
\$k61	88	0x00000000
\$k62	89	0x00000000
\$k63	90	0x00000000
\$k64	91	0x00000000
\$k65	92	0x00000000
\$k66	93	0x00000000
\$k67	94	0x00000000
\$k68	95	0x00000000
\$k69	96	0x00000000
\$k70	97	0x00000000
\$k71	98	0x00000000
\$k72	99	0x00000000
\$k73	100	0x00000000
\$k74	101	0x00000000
\$k75	102	0x00000000
\$k76	103	0x00000000
\$k77	104	0x00000000
\$k78	105	0x00000000
\$k79	106	0x00000000
\$k80	107	0x00000000
\$k81	108	0x00000000
\$k82	109	0x00000000
\$k83	110	0x00000000
\$k84	111	0x00000000
\$k85	112	0x00000000
\$k86	113	0x00000000
\$k87	114	0x00000000
\$k88	115	0x00000000
\$k89	116	0x00000000
\$k90	117	0x00000000
\$k91	118	0x00000000
\$k92	119	0x00000000
\$k93	120	0x00000000
\$k94	121	0x00000000
\$k95	122	0x00000000
\$k96	123	0x00000000
\$k97	124	0x00000000
\$k98	125	0x00000000
\$k99	126	0x00000000
\$k100	127	0x00000000
\$k101	128	0x00000000
\$k102	129	0x00000000
\$k103	130	0x00000000
\$k104	131	0x00000000
\$k105	132	0x00000000
\$k106	133	0x00000000
\$k107	134	0x00000000
\$k108	135	0x00000000
\$k109	136	0x00000000
\$k110	137	0x00000000
\$k111	138	0x00000000
\$k112	139	0x00000000
\$k113	140	0x00000000
\$k114	141	0x00000000
\$k115	142	0x00000000
\$k116	143	0x00000000
\$k117	144	0x00000000
\$k118	145	0x00000000
\$k119	146	0x00000000
\$k120	147	0x00000000
\$k121	148	0x00000000
\$k122	149	0x00000000
\$k123	150	0x00000000
\$k124	151	0x00000000
\$k125	152	0x00000000
\$k126	153	0x00000000
\$k127	154	0x00000000
\$k128	155	0x00000000
\$k129	156	0x00000000
\$k130	157	0x00000000
\$k131	158	0x00000000
\$k132	159	0x00000000
\$k133	160	0x00000000
\$k134	161	0x00000000
\$k135	162	0x00000000
\$k136	163	0x00000000
\$k137	164	0x00000000
\$k138	165	0x00000000
\$k139	166	0x00000000
\$k140	167	0x00000000
\$k141	168	0x00000000
\$k142	169	0x00000000
\$k143	170	0x00000000
\$k144	171	0x00000000
\$k145	172	0x00000000
\$k146	173	0x00000000
\$k147	174	0x00000000
\$k148	175	0x00000000
\$k149	176	0x00000000
\$k150	177	0x00000000
\$k151	178	0x00000000
\$k152	179	0x00000000
\$k153	180	0x00000000
\$k154	181	0x00000000
\$k155	182	0x00000000
\$k156	183	0x00000000
\$k157	184	0x00000000
\$k158	185	0x00000000
\$k159	186	0x00000000
\$k160	187	0x00000000
\$k161	188	0x00000000
\$k162	189	0x00000000
\$k163	190	0x00000000
\$k164	191	0x00000000
\$k165	192	0x00000000
\$k166	193	0x00000000
\$k167	194	0x00000000
\$k168	195	0x00000000
\$k169	196	0x00000000
\$k170	197	0x00000000
\$k171	198	0x00000000
\$k172	199	0x00000000
\$k173	200	0x00000000
\$k174	201	0x00000000
\$k175	202	0x00000000
\$k176	203	0x00000000
\$k177	204	0x00000000
\$k178	205	0x00000000
\$k179	206	0x00000000
\$k180	207	0x00000000
\$k181	208	0x00000000
\$k182	209	0x00000000
\$k183	210	0x00000000
\$k184	211	0x00000000
\$k185	212	0x00000000
\$k186	213	0x00000000
\$k187	214	0x00000000
\$k188	215	0x00000000
\$k189	216	0x00000000
\$k190	217	0x00000000
\$k191	218	0x00000000
\$k192	219	0x00000000
\$k193	220	0x00000000
\$k194	221	0x00000000
\$k195	222	0x00000000
\$k196	223	0x00000000
\$k197	224	0x00000000
\$k198	225	0x00000000
\$k199	226	0x00000000
\$k200	227	0x00000000
\$k201	228	0x00000000
\$k202	229	0x00000000
\$k203	230	0x00000000
\$k204	231	0x00000000
\$k205	232	0x00000000
\$k206	233	0x00000000
\$k207	234	0x00000000
\$k208	235	0x00000000
\$k209	236	0x00000000
\$k210	237	0x00000000
\$k211	238	0x00000000
\$k212	239	0x00000000
\$k213	240	0x00000000
\$k214	241	0x00000000
\$k215	242	0x00000000
\$k216	243	0x00000000
\$k217	244	0x00000000
\$k218	245	0x00000000
\$k219	246	0x00000000
\$k220	247	0x00000000
\$k221	248	0x00000000
\$k222	249	0x00000000
\$k223	250	0x00000000
\$k224	251	0x00000000
\$k225	252	0x00000000
\$k226	253	0x00000000
\$k227	254	0x00000000
\$k228	255	0x00000000
\$k229	256	0x00000000
\$k230	257	0x00000000
\$k231	258	0x00000000
\$k232	259	0x00000000
\$k233	260	0x00000000
\$k234	261	0x00000000
\$k235	262	0x00000000
\$k236	263	0x00000000
\$k237	264	0x00000000
\$k238	265	0x00000000
\$k239	266	0x00000000
\$k240		

Assignment 6:

```
.data
i: .word -1
A: .word -5, -10, 8, 9, 4
max: .word -1

.text
la $t0, i # load the address of i variable
lw $s0, 0($t0) # load the value of i to $s0
la $t1, A # load the address of array A
la $t6, max # load the address of max variable
lw $s6, 0($t6) # load the value of max to $s6
addi $s7, $s0, 5 # the number of elements in array = 5
loop:
addi $s0, $s0, 1 # i = i + 1
sw $s0, 0($t0)
sll $s1, $s0, 2 # s1 = i * 4
add $t2, $s1, $t1 # load the address of A[i]
lw $t2, 0($t2) # load the value of A[i] to $t2
slt $t3, $t2, $0 # t3 = 1 if (t2 < 0) else t3 = 0
beq $t3, $0, else
sub $t2, $0, $t2 # exp(A[i]) = -A[i]
j endif
else:
endif:
slt $t5, $s6, $t2 # t5 = 1 if (max < A[i]) else t5 = 0
beq $t5, $0, end # branch to end if (max >= A[i])
```

Edit

Execute

Text Segment

Brkt	Address	Code	Basic		
	0x00400000	0x00400000	0x00000000		
	0x00400001	0x00400001	0x00000000		
	0x00400002	0x00400002	0x00000000		
	0x00400003	0x00400003	0x00000000		
	0x00400004	0x00400004	0x00000000		
	0x00400005	0x00400005	0x00000000		
	0x00400006	0x00400006	0x00000000		
	0x00400007	0x00400007	0x00000000		
	0x00400008	0x00400008	0x00000000		
	0x00400009	0x00400009	0x00000000		
	0x0040000A	0x0040000A	0x00000000		
	0x0040000B	0x0040000B	0x00000000		
	0x0040000C	0x0040000C	0x00000000		
	0x0040000D	0x0040000D	0x00000000		
	0x0040000E	0x0040000E	0x00000000		
	0x0040000F	0x0040000F	0x00000000		
	0x00400010	0x00400010	0x00000000		
	0x00400011	0x00400011	0x00000000		
	0x00400012	0x00400012	0x00000000		
	0x00400013	0x00400013	0x00000000		
	0x00400014	0x00400014	0x00000000		
	0x00400015	0x00400015	0x00000000		
	0x00400016	0x00400016	0x00000000		
	0x00400017	0x00400017	0x00000000		
	0x00400018	0x00400018	0x00000000		
	0x00400019	0x00400019	0x00000000		
	0x0040001A	0x0040001A	0x00000000		
	0x0040001B	0x0040001B	0x00000000		
	0x0040001C	0x0040001C	0x00000000		
	0x0040001D	0x0040001D	0x00000000		
	0x0040001E	0x0040001E	0x00000000		
	0x0040001F	0x0040001F	0x00000000		
	0x00400020	0x00400020	0x00000000		
	0x00400021	0x00400021	0x00000000		
	0x00400022	0x00400022	0x00000000		
	0x00400023	0x00400023	0x00000000		
	0x00400024	0x00400024	0x00000000		
	0x00400025	0x00400025	0x00000000		
	0x00400026	0x00400026	0x00000000		
	0x00400027	0x00400027	0x00000000		
	0x00400028	0x00400028	0x00000000		
	0x00400029	0x00400029	0x00000000		
	0x0040002A	0x0040002A	0x00000000		
	0x0040002B	0x0040002B	0x00000000		
	0x0040002C	0x0040002C	0x00000000		
	0x0040002D	0x0040002D	0x00000000		
	0x0040002E	0x0040002E	0x00000000		
	0x0040002F	0x0040002F	0x00000000		
	0x00400030	0x00400030	0x00000000		
	0x00400031	0x00400031	0x00000000		
	0x00400032	0x00400032	0x00000000		
	0x00400033	0x00400033	0x00000000		
	0x00400034	0x00400034	0x00000000		
	0x00400035	0x00400035	0x00000000		
	0x00400036	0x00400036	0x00000000		
	0x00400037	0x00400037	0x00000000		
	0x00400038	0x00400038	0x00000000		
	0x00400039	0x00400039	0x00000000		
	0x0040003A	0x0040003A	0x00000000		
	0x0040003B	0x0040003B	0x00000000		
	0x0040003C	0x0040003C	0x00000000		
	0x0040003D	0x0040003D	0x00000000		
	0x0040003E	0x0040003E	0x00000000		
	0x0040003F	0x0040003F	0x00000000		
	0x00400040	0x00400040	0x00000000		
	0x00400041	0x00400041	0x00000000		
	0x00400042	0x00400042	0x00000000		
	0x00400043	0x00400043	0x00000000		
	0x00400044	0x00400044	0x00000000		
	0x00400045	0x00400045	0x00000000		
	0x00400046	0x00400046	0x00000000		
	0x00400047	0x00400047	0x00000000		
	0x00400048	0x00400048	0x00000000		
	0x00400049	0x00400049	0x00000000		
	0x0040004A	0x0040004A	0x00000000		
	0x0040004B	0x0040004B	0x00000000		
	0x0040004C	0x0040004C	0x00000000		
	0x0040004D	0x0040004D	0x00000000		
	0x0040004E	0x0040004E	0x00000000		
	0x0040004F	0x0040004F	0x00000000		
	0x00400050	0x00400050	0x00000000		
	0x00400051	0x00400051	0x00000000		
	0x00400052	0x00400052	0x00000000		
	0x00400053	0x00400053	0x00000000		
	0x00400054	0x00400054	0x00000000		
	0x00400055	0x00400055	0x00000000		
	0x00400056	0x00400056	0x00000000		
	0x00400057	0x00400057	0x00000000		
	0x00400058	0x00400058	0x00000000		
	0x00400059	0x00400059	0x00000000		
	0x0040005A	0x0040005A	0x00000000		
	0x0040005B	0x0040005B	0x00000000		
	0x0040005C	0x0040005C	0x00000000		
	0x0040005D	0x0040005D	0x00000000		
	0x0040005E	0x0040005E	0x00000000		
	0x0040005F	0x0040005F	0x00000000		
	0x00400060	0x00400060	0x00000000		
	0x00400061	0x00400061	0x00000000		
	0x00400062	0x00400062	0x00000000		
	0x00400063	0x00400063	0x00000000		
	0x00400064	0x00400064	0x00000000		
	0x00400065	0x00400065	0x00000000		
	0x00400066	0x00400066	0x00000000		
	0x00400067	0x00400067	0x00000000		
	0x00400068	0x00400068	0x00000000		
	0x00400069	0x00400069	0x00000000		
	0x0040006A	0x0040006A	0x00000000		
	0x0040006B	0x0040006B	0x00000000		
	0x0040006C	0x0040006C	0x00000000		
	0x0040006D	0x0040006D	0x00000000		
	0x0040006E	0x0040006E	0x00000000		
	0x0040006F	0x0040006F	0x00000000		
	0x00400070	0x00400070	0x00000000		
	0x00400071	0x00400071	0x00000000		
	0x00400072	0x00400072	0x00000000		
	0x00400073	0x00400073	0x00000000		
	0x00400074	0x00400074	0x00000000		
	0x00400075	0x00400075	0x00000000		
	0x00400076	0x00400076	0x00000000		
	0x00400077	0x00400077	0x00000000		
	0x00400078	0x00400078	0x00000000		
	0x00400079	0x00400079	0x00000000		
	0x0040007A	0x0040007A	0x00000000		
	0x0040007B	0x0040007B	0x00000000		
	0x0040007C	0x0040007C	0x00000000		
	0x0040007D	0x0040007D	0x00000000		
	0x0040007E	0x0040007E	0x00000000		
	0x0040007F	0x0040007F	0x00000000		
	0x00400080	0x00400080	0x00000000		
	0x00400081	0x00400081	0x00000000		
	0x00400082	0x00400082	0x00000000		
	0x00400083	0x00400083	0x00000000		
	0x00400084	0x00400084	0x00000000		
	0x00400085	0x00400085	0x00000000		
	0x00400086	0x00400086	0x00000000		
	0x00400087	0x00400087	0x00000000		
	0x00400088	0x00400088	0x00000000		
	0x00400089	0x00400089	0x00000000		
	0x0040008A	0x0040008A	0x00000000		
	0x0040008B	0x0040008B	0x00000000		
	0x0040008C	0x0040008C	0x00000000		
	0x0040008D	0x0040008D	0x00000000		
	0x0040008E	0x0040008E	0x00000000		
	0x0040008F	0x0040008F	0x00000000		
	0x00400090	0x00400090	0x00000000		
	0x00400091	0x00400091	0x00000000		
	0x00400092	0x00400092	0x00000000		
	0x00400093	0x00400093	0x00000000		
	0x00400094	0x00400094	0x00000000		
	0x00400095	0x00400095	0x00000000		
	0x00400096	0x00400096	0x00000000		
	0x00400097	0x00400097	0x00000000		
	0x00400098	0x00400098	0x00000000		
	0x00400099	0x00400099	0x00000000		
	0x0040009A	0x0040009A	0x00000000		
	0x0040009B	0x0040009B	0x00000000		
	0x0040009C	0x0040009C	0x00000000		
	0x0040009D	0x0040009D	0x00000000		
	0x0040009E	0x0040009E	0x00000000		
	0x0040009F	0x0040009F	0x00000000		
	0x004000A0	0x004000A0	0x00000000		
	0x004000A1	0x004000A1	0x00000000		
	0x004000A2	0x004000A2	0x00000000		
	0x004000A3	0x004000A3	0x00000000		
	0x004000A4	0x004000A4	0x00000000		
	0x004000A5	0x004000A5	0x00000000		
	0x004000A6	0x004000A6	0x00000000		
	0x004000A7	0x004000A7	0x00000000		
	0x004000A8	0x004000A8	0x00000000		
	0x004000A9	0x004000A9	0x00000000		
	0x004000AA	0x004000AA	0x00000000		
	0x004000AB	0x004000AB	0x00000000		
	0x004000AC	0x004000AC	0x00000000		
	0x004000AD	0x004000AD	0x00000000		
	0x004000AE	0x004000AE	0x00000000		
	0x004000AF	0x004000AF	0x00000000		
	0x004000B0	0x004000B0	0x00000000		
	0x004000B1	0x004000B1	0x00000000		
	0x004000B2	0x004000B2	0x00000000		
	0x004000B3	0x004000B3	0x00000000		
	0x004000B4	0x004000B4	0x00000000		
	0x004000B5	0x004000B5	0x00000000		
	0x004000B6	0x004000B6	0x00000000		
	0x004000B7	0x004000B7	0x00000000		
	0x004000B8	0x004000B8	0x00000000		
	0x004000B9	0x004000B9	0x00000000		
	0x004000BA	0x004000BA	0x00000000		
	0x004000BB	0x004000BB	0x00000000		
	0x004000BC	0x004000BC	0x00000000		
	0x004000BD	0x004000BD	0x00000000		
	0x004000BE	0x004000BE	0x00000000		
	0x004000BF	0x004000BF	0x00000000		
	0x004000C0	0x004000C0	0x00000000		
	0x004000C1	0x004000C1	0x00000000		
	0x004000C2	0x004000C2	0x00000000		
	0x004000C3	0x004000C3	0x00000000		
	0x004000C4	0x004000C4	0x00000000		
	0x004000C5	0x004000C5	0x00000000		
	0x004000C6	0x004000C6	0x00000000		
	0x004000C7	0x004000C7	0x00000000		
	0x004000C8	0x004000C8	0x00000000		
	0x004000C9	0x004000C9	0x00000000		
	0x				