**(gdb) disass /rs**

**Dump of assembler code for function main:**

**poin3.c:**

**3 {**

**0x5655619d <+0>: 8d 4c 24 04 lea 0x4(%esp),%ecx**

**0x565561a1 <+4>: 83 e4 f0 and $0xfffffff0,%esp**

**0x565561a4 <+7>: ff 71 fc push -0x4(%ecx)**

**0x565561a7 <+10>: 55 push %ebp**

**0x565561a8 <+11>: 89 e5 mov %esp,%ebp**

**0x565561aa <+13>: 53 push %ebx**

**0x565561ab <+14>: 51 push %ecx**

**0x565561ac <+15>: 83 ec 10 sub $0x10,%esp**

**0x565561af <+18>: e8 ec fe ff ff call 0x565560a0 <\_\_x86.get\_pc\_thunk.bx>**

**0x565561b4 <+23>:81 c3 24 2e 00 00 add $0x2e24,%ebx**

**4 static short num[ ][4]={ {2, 9, -1, 5},{3, 8, 2, -6}};**

**5 static short \*pn[ ]={num[0], num[1]};**

**6 static short s[2]={0, 0};**

**7 int i, j;**

**8 for (i=0; i<2; i++) {**

**0x565561ba <+29>:c7 45 f0 00 00 00 00 movl $0x0,-0x10(%ebp) #取i值**

**0x565561c1 <+36>:eb 75 jmp 0x56556238 <main+155>**

**9 for (j=0; j<4; j++)**

**0x565561c3 <+38>:c7 45 f4 00 00 00 00 movl $0x0,-0xc(%ebp)#取j值**

**0x565561ca <+45>: eb 3c jmp 0x56556208 <main+107>**

**10 s[i]+=\*pn[i]++;**

**0x565561cc <+47>: 8b 45 f0 mov -0x10(%ebp),%eax #取i值**

**0x565561cf <+50>: 8b 84 83 40 00 00 00 mov 0x40(%ebx,%eax,4),%eax 取pn[i]即num[0]的地址**

**0x565561d6 <+57>:8d 48 02 lea 0x2(%eax),%ecx #实现\*pn[i]++即指针指向num[i]的第j+1个数**

**0x565561d9 <+60>: 8b 55 f0 mov -0x10(%ebp),%edx #取i值**

**0x565561dc <+63>: 89 8c 93 40 00 00 00 mov %ecx,0x40(%ebx,%edx,4)#送回pn**

**0x565561e3 <+70>: 0f b7 10 movzwl (%eax),%edx #取num[i][j]值0扩展至edx**

**0x565561e6 <+73>: 8b 45 f0 mov -0x10(%ebp),%eax**

**0x565561e9 <+76>:0f b7 84 43 4c 00 00 00 movzwl 0x4c(%ebx,%eax,2),%eax#取s[i]的值0扩展**

**0x565561f1 <+84>: 89 c1 mov %eax,%ecx #s[i]的值送入ecx**

**0x565561f3 <+86>: 89 d0 mov %edx,%eax #num[i][j]的值送入eax**

**0x565561f5 <+88>: 01 c8 add %ecx,%eax #s[i]+=num[i][j]**

**0x565561f7 <+90>: 89 c2 mov %eax,%edx**

**0x565561f9 <+92>: 8b 45 f0 mov -0x10(%ebp),%eax**

**0x565561fc <+95>: 66 89 94 43 4c 00 00 00 mov %dx,0x4c(%ebx,%eax,2)**

**9 for (j=0; j<4; j++)**

**0x56556204 <+103>: 83 45 f4 01 addl $0x1,-0xc(%ebp)**

**0x56556208 <+107>: 83 7d f4 03 cmpl $0x3,-0xc(%ebp)**

**0x5655620c <+111>: 7e be jle 0x565561cc <main+47>**

**11 printf("sum of line %d：%d\n", i+1, s[i]);**

**0x5655620e <+113>: 8b 45 f0 mov -0x10(%ebp),%eax**

**0x56556211 <+116>: 0f b7 84 43 4c 00 00 00 movzwl 0x4c(%ebx,%eax,2),%eax**

**0x56556219 <+124>: 98 cwtl**

**0x5655621a <+125>: 8b 55 f0 mov -0x10(%ebp),%edx**

**0x5655621d <+128>: 83 c2 01 add $0x1,%edx**

**0x56556220 <+131>: 83 ec 04 sub $0x4,%esp**

**0x56556223 <+134>: 50 push %eax**

**0x56556224 <+135>: 52 push %edx**

**0x56556225 <+136>: 8d 83 30 e0 ff ff lea -0x1fd0(%ebx),%eax**

**0x5655622b <+142>: 50 push %eax**

**0x5655622c <+143>: e8 1f fe ff ff call 0x56556050 <printf@plt>**

**0x56556231 <+148>: 83 c4 10 add $0x10,%esp**

**8 for (i=0; i<2; i++) {**

**0x56556234 <+151>: 83 45 f0 01 addl $0x1,-0x10(%ebp)**

**0x56556238 <+155>: 83 7d f0 01 cmpl $0x1,-0x10(%ebp)**

**0x5655623c <+159>: 7e 85 jle 0x565561c3 <main+38>**

**0x5655623e <+161>: b8 00 00 00 00 mov $0x0,%eax**

**12 }**

**13 }**

**0x56556243 <+166>: 8d 65 f8 lea -0x8(%ebp),%esp**

**0x56556246 <+169>: 59 pop %ecx**

**0x56556247 <+170>: 5b pop %ebx**

**0x56556248 <+171>: 5d pop %ebp**

**0x56556249 <+172>: 8d 61 fc lea -0x4(%ecx),%esp**

**0x5655624c <+175>: c3 ret**