测试结果如下，符合预期结果。

please input source program file name :

14061183\_test.txt

Open successfully!

1 constsy const

2 intsy int

3 ident MAX

4 becomes =

5 intcon 30

6 comma ,

7 ident MIN

8 becomes =

9 intcon 1

10 semicolon ;

11 constsy const

12 charsy char

13 ident \_CH1

14 becomes =

15 charcon +

16 semicolon ;

17 intsy int

18 ident gbinta

19 lbrack [

20 intcon 100

21 rbrack ]

22 comma ,

23 ident gbi

24 semicolon ;

25 charsy char

26 ident gbchara

27 lbrack [

28 intcon 5

29 rbrack ]

30 comma ,

31 ident gbc

32 semicolon ;

33 voidsy void

34 ident fun

35 lparent (

36 intsy int

37 ident n

38 comma ,

39 charsy char

40 ident c

41 rparent )

42 lbrace {

43 intsy int

44 ident i

45 semicolon ;

46 ident gbi

47 becomes =

48 ident n

49 semicolon ;

50 ident i

51 becomes =

52 ident n

53 semicolon ;

54 dosy do

55 lbrace {

56 ident i

57 becomes =

58 ident i

59 plus +

60 intcon 1

61 semicolon ;

62 ident gbi

63 becomes =

64 ident gbi

65 plus +

66 ident i

67 semicolon ;

68 rbrace }

69 whilesy while

70 lparent (

71 ident i

72 lss <

73 ident MAX

74 rparent )

75 printfsy printf

76 lparent (

77 stringcon fun: gbi:

78 comma ,

79 ident gbi

80 rparent )

81 semicolon ;

82 printfsy printf

83 lparent (

84 stringcon fun: c:

85 comma ,

86 ident c

87 rparent )

88 semicolon ;

89 rbrace }

90 intsy int

91 ident fun1

92 lparent (

93 intsy int

94 ident n

95 rparent )

96 lbrace {

97 intsy int

98 ident sum

99 semicolon ;

100 ifsy if

101 lparent (

102 ident n

103 lss <

104 ident MAX

105 rparent )

106 lbrace {

107 returnsy return

108 lparent (

109 ident fun1

110 lparent (

111 ident n

112 plus +

113 intcon 1

114 rparent )

115 plus +

116 ident n

117 rparent )

118 semicolon ;

119 rbrace }

120 returnsy return

121 ident n

122 semicolon ;

123 rbrace }

124 voidsy void

125 ident fun2

126 lparent (

127 rparent )

128 lbrace {

129 constsy const

130 intsy int

131 ident n

132 becomes =

133 intcon 5

134 semicolon ;

135 ident gbi

136 becomes =

137 ident n

138 times \*

139 ident n

140 div /

141 ident n

142 semicolon ;

143 forsy for

144 lparent (

145 ident i

146 becomes =

147 ident gbi

148 minus -

149 ident gbi

150 semicolon ;

151 ident i

152 lss <

153 ident n

154 semicolon ;

155 ident i

156 becomes =

157 ident i

158 plus +

159 intcon 1

160 rparent )

161 lbrace {

162 ident gbinta

163 lbrack [

164 ident i

165 rbrack ]

166 becomes =

167 ident i

168 semicolon ;

169 printfsy printf

170 lparent (

171 stringcon fun2: gbinta:

172 comma ,

173 ident gbinta

174 lbrack [

175 ident i

176 rbrack ]

177 rparent )

178 semicolon ;

179 rbrace }

180 printfsy printf

181 lparent (

182 stringcon fun2: c: no such variable

183 rparent )

184 semicolon ;

185 rbrace }

186 charsy char

187 ident fun3

188 lparent (

189 rparent )

190 lbrace {

191 printfsy printf

192 lparent (

193 stringcon fun3: return a

194 rparent )

195 semicolon ;

196 returnsy return

197 lparent (

198 charcon a

199 rparent )

200 semicolon ;

201 rbrace }

202 voidsy void

203 mainsy main

204 lparent (

205 rparent )

206 lbrace {

207 constsy const

208 ident M\_MAX

209 becomes =

210 intcon 20

211 comma ,

212 ident M\_MIN

213 becomes =

214 intcon 10

215 semicolon ;

216 intsy int

217 ident n1

218 comma ,

219 ident n2

220 comma ,

221 ident n3

222 comma ,

223 ident n4

224 semicolon ;

225 charsy char

226 ident c1

227 comma ,

228 ident c2

229 comma ,

230 ident c3

231 comma ,

232 ident c4

233 comma ,

234 ident c5

235 comma ,

236 ident C1

237 semicolon ;

238 intsy int

239 ident inta

240 lbrack [

241 intcon 100

242 rbrack ]

243 semicolon ;

244 charsy char

245 ident chara

246 lbrack [

247 intcon 5

248 rbrack ]

249 semicolon ;

250 scanfsy scanf

251 lparent (

252 ident n1

253 comma ,

254 ident c1

255 rparent )

256 semicolon ;

257 scanfsy scanf

258 lparent (

259 ident n2

260 comma ,

261 ident c2

262 rparent )

263 semicolon ;

264 scanfsy scanf

265 lparent (

266 ident n3

267 comma ,

268 ident c3

269 comma ,

270 ident n4

271 comma ,

272 ident c4

273 rparent )

274 semicolon ;

275 scanfsy scanf

276 lparent (

277 ident c5

278 rparent )

279 semicolon ;

280 ifsy if

281 lparent (

282 ident n1

283 lss <

284 ident MAX

285 rparent )

286 lbrace {

287 ident fun

288 lparent (

289 ident n1

290 comma ,

291 ident c1

292 rparent )

293 semicolon ;

294 rbrace }

295 elsesy else

296 lbrace {

297 printfsy printf

298 lparent (

299 stringcon n1 >= MAX

300 rparent )

301 semicolon ;

302 rbrace }

303 ifsy if

304 lparent (

305 ident n2

306 neq !=

307 ident MIN

308 rparent )

309 lbrace {

310 printfsy printf

311 lparent (

312 stringcon n2 != MIN

313 rparent )

314 semicolon ;

315 rbrace }

316 elsesy else

317 lbrace {

318 ident gbi

319 becomes =

320 ident fun1

321 lparent (

322 ident n2

323 rparent )

324 semicolon ;

325 printfsy printf

326 lparent (

327 stringcon fun1: gbi:

328 comma ,

329 ident gbi

330 rparent )

331 semicolon ;

332 rbrace }

333 ifsy if

334 lparent (

335 ident n3

336 gtr >

337 ident M\_MAX

338 rparent )

339 lbrace {

340 printfsy printf

341 lparent (

342 stringcon n3 <= M\_MAX

343 rparent )

344 semicolon ;

345 rbrace }

346 elsesy else

347 lbrace {

348 ident fun2

349 lparent (

350 ident n3

351 comma ,

352 ident c3

353 rparent )

354 semicolon ;

355 ident fun1

356 lparent (

357 ident n3

358 rparent )

359 semicolon ;

360 rbrace }

361 ifsy if

362 lparent (

363 ident n4

364 geq >=

365 ident M\_MIN

366 rparent )

367 lbrace {

368 ident C1

369 becomes =

370 ident fun3

371 lparent (

372 rparent )

373 semicolon ;

374 printfsy printf

375 lparent (

376 stringcon fun3: C1:

377 comma ,

378 ident C1

379 rparent )

380 semicolon ;

381 rbrace }

382 ifsy if

383 lparent (

384 ident n4

385 leq <=

386 ident M\_MIN

387 rparent )

388 lbrace {

389 printfsy printf

390 lparent (

391 stringcon n4 <= M\_MIN

392 rparent )

393 semicolon ;

394 rbrace }

395 ifsy if

396 lparent (

397 ident c1

398 neq !=

399 ident C1

400 rparent )

401 lbrace {

402 printfsy printf

403 lparent (

404 stringcon c1 and C1 are not equal.

405 rparent )

406 semicolon ;

407 rbrace }

408 ifsy if

409 lparent (

410 intcon 2

411 rparent )

412 lbrace {

413 printfsy printf

414 lparent (

415 stringcon 2 is true.

416 rparent )

417 semicolon ;

418 rbrace }

419 ifsy if

420 lparent (

421 intcon 0

422 rparent )

423 lbrace {

424 printfsy printf

425 lparent (

426 stringcon 0 is true.

427 rparent )

428 semicolon ;

429 rbrace }

430 elsesy else

431 lbrace {

432 printfsy printf

433 lparent (

434 stringcon 0 is false.

435 rparent )

436 semicolon ;

437 rbrace }

438 ifsy if

439 lparent (

440 ident c5

441 minus -

442 ident c1

443 rparent )

444 lbrace {

445 printfsy printf

446 lparent (

447 stringcon c5 - c1 is true.

448 rparent )

449 semicolon ;

450 rbrace }

451 elsesy else

452 lbrace {

453 printfsy printf

454 lparent (

455 stringcon c5 - c1 is false.

456 rparent )

457 semicolon ;

458 rbrace }

Process returned 0 (0x0) execution time : 6.537 s

Press any key to continue.