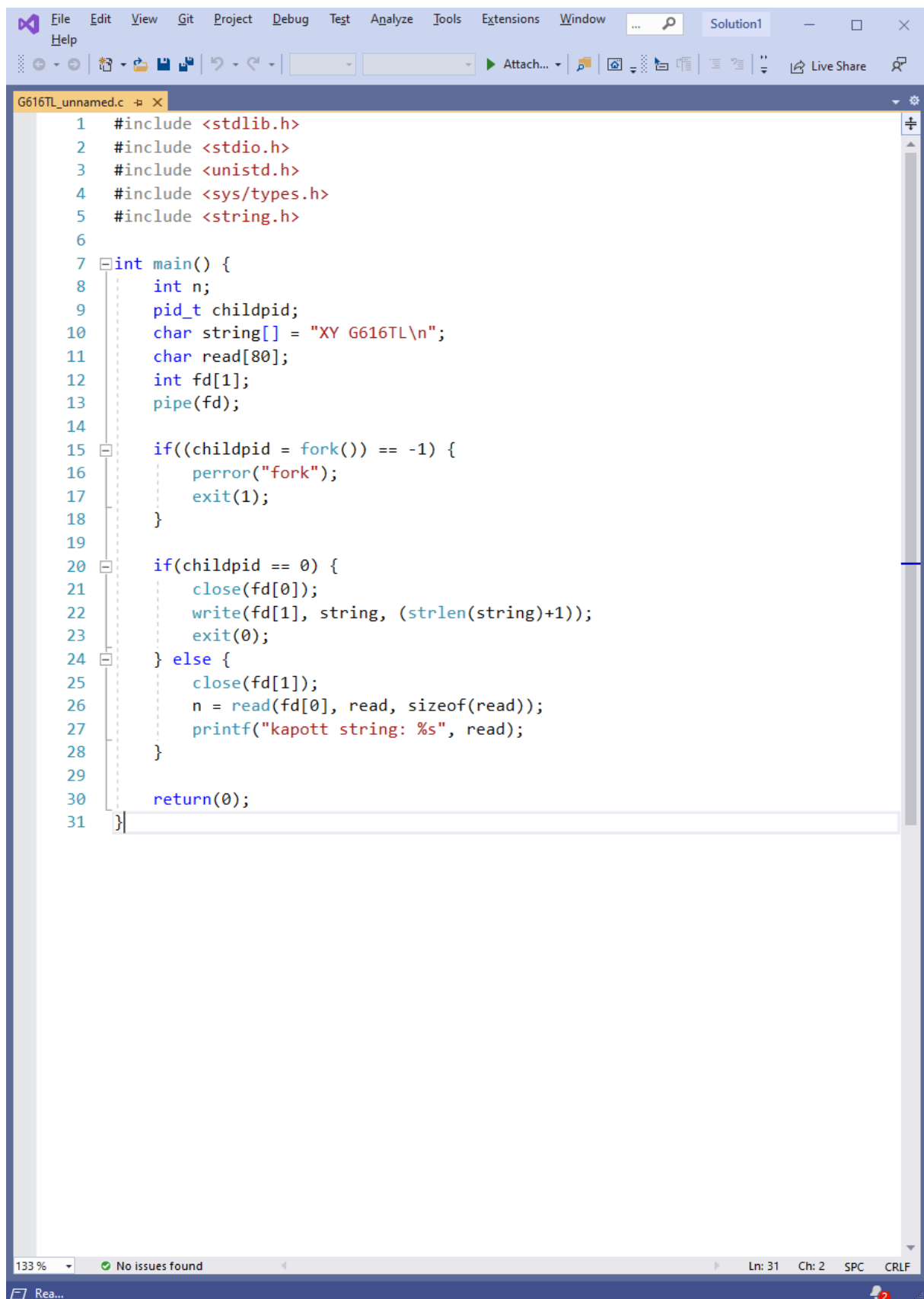


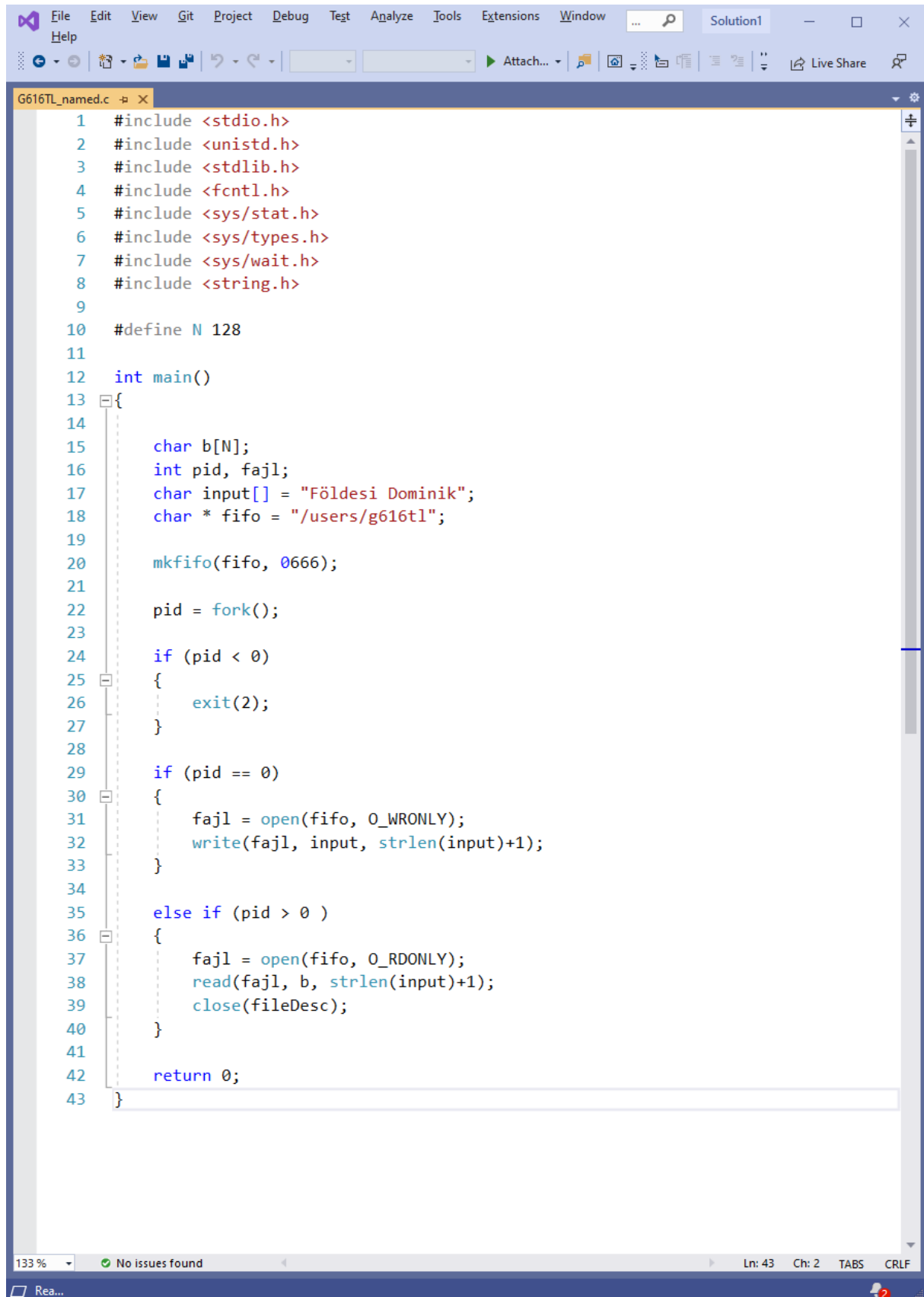
1.



```
1  #include <stdlib.h>
2  #include <stdio.h>
3  #include <unistd.h>
4  #include <sys/types.h>
5  #include <string.h>
6
7  int main() {
8      int n;
9      pid_t childpid;
10     char string[] = "XY G616TL\n";
11     char read[80];
12     int fd[1];
13     pipe(fd);
14
15     if((childpid = fork()) == -1) {
16         perror("fork");
17         exit(1);
18     }
19
20     if(childpid == 0) {
21         close(fd[0]);
22         write(fd[1], string, (strlen(string)+1));
23         exit(0);
24     } else {
25         close(fd[1]);
26         n = read(fd[0], read, sizeof(read));
27         printf("kapott string: %s", read);
28     }
29
30     return(0);
31 }
```

133 % No issues found Ln: 31 Ch: 2 SPC CRLF

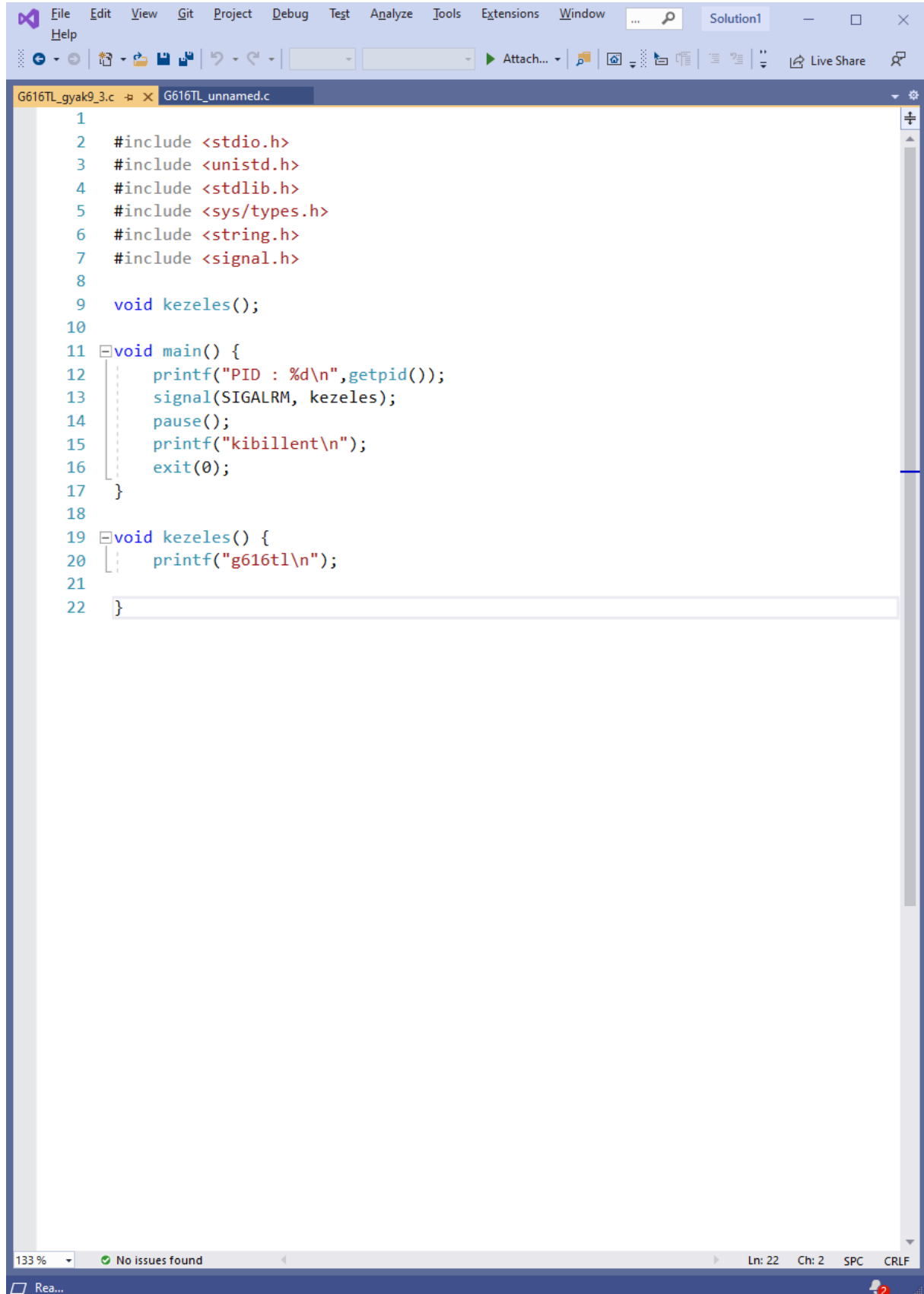
2.



```
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <stdlib.h>
4  #include <fcntl.h>
5  #include <sys/stat.h>
6  #include <sys/types.h>
7  #include <sys/wait.h>
8  #include <string.h>
9
10 #define N 128
11
12 int main()
13 {
14
15     char b[N];
16     int pid, fajt;
17     char input[] = "Földesi Dominik";
18     char * fifo = "/users/g616tl";
19
20     mkfifo(fifo, 0666);
21
22     pid = fork();
23
24     if (pid < 0)
25     {
26         exit(2);
27     }
28
29     if (pid == 0)
30     {
31         fajt = open(fifo, O_WRONLY);
32         write(fajt, input, strlen(input)+1);
33     }
34
35     else if (pid > 0 )
36     {
37         fajt = open(fifo, O_RDONLY);
38         read(fajt, b, strlen(input)+1);
39         close(fajt);
40     }
41
42     return 0;
43 }
```

133 % No issues found Ln: 43 Ch: 2 TABS CRLF Rea...

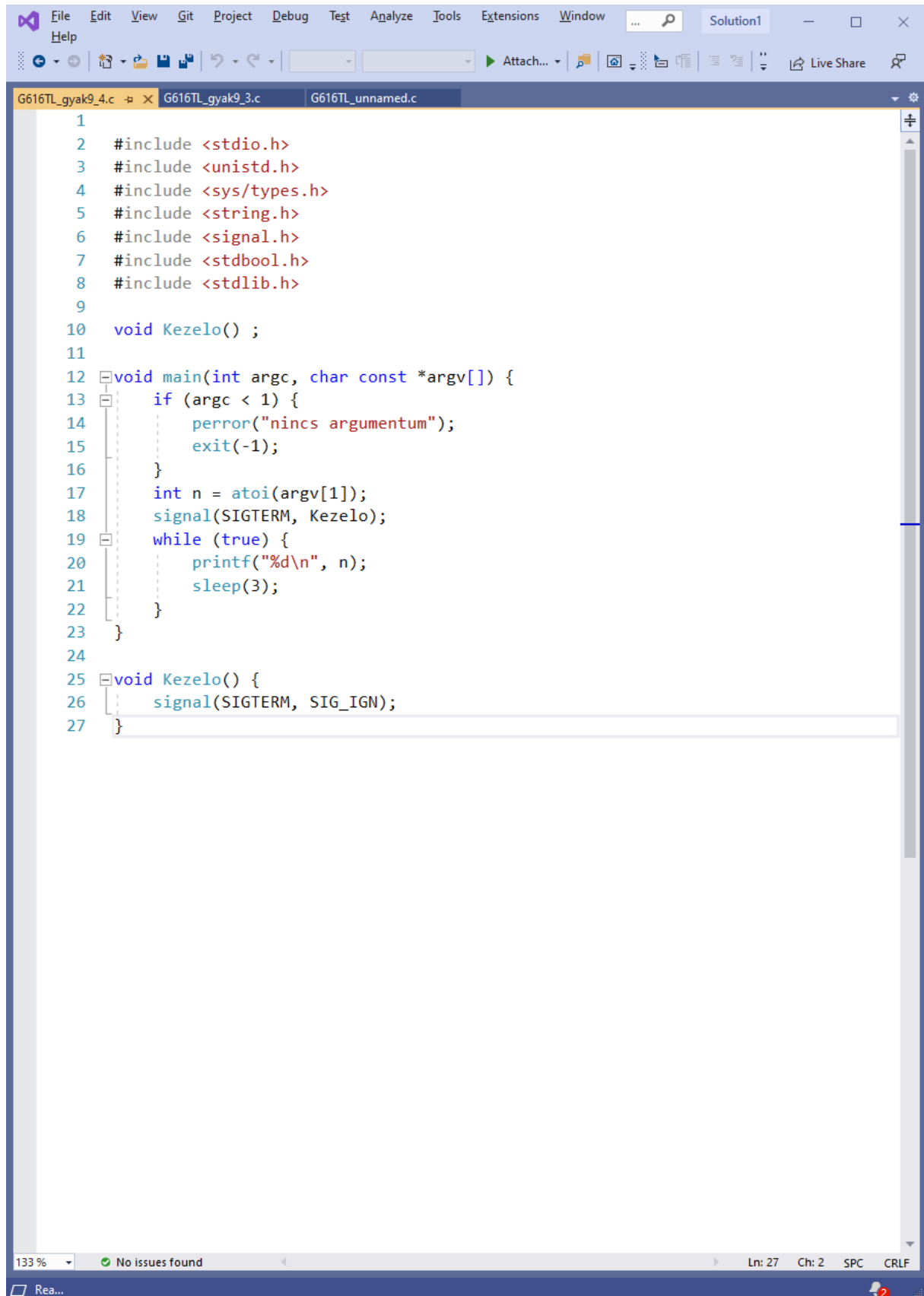
3



```
1
2 #include <stdio.h>
3 #include <unistd.h>
4 #include <stdlib.h>
5 #include <sys/types.h>
6 #include <string.h>
7 #include <signal.h>
8
9 void kezeles();
10
11 void main() {
12     printf("PID : %d\n", getpid());
13     signal(SIGALRM, kezeles);
14     pause();
15     printf("kibillent\n");
16     exit(0);
17 }
18
19 void kezeles() {
20     printf("g616t1\n");
21 }
22 }
```

133 % No issues found Ln: 22 Ch: 2 SPC CRLF

4.



```
1
2 #include <stdio.h>
3 #include <unistd.h>
4 #include <sys/types.h>
5 #include <string.h>
6 #include <signal.h>
7 #include <stdbool.h>
8 #include <stdlib.h>
9
10 void Kezelo() ;
11
12 void main(int argc, char const *argv[]) {
13     if (argc < 1) {
14         perror("nincs argumentum");
15         exit(-1);
16     }
17     int n = atoi(argv[1]);
18     signal(SIGTERM, Kezelo);
19     while (true) {
20         printf("%d\n", n);
21         sleep(3);
22     }
23 }
24
25 void Kezelo() {
26     signal(SIGTERM, SIG_IGN);
27 }
```

133 % No issues found Ln: 27 Ch: 2 SPC CRLF