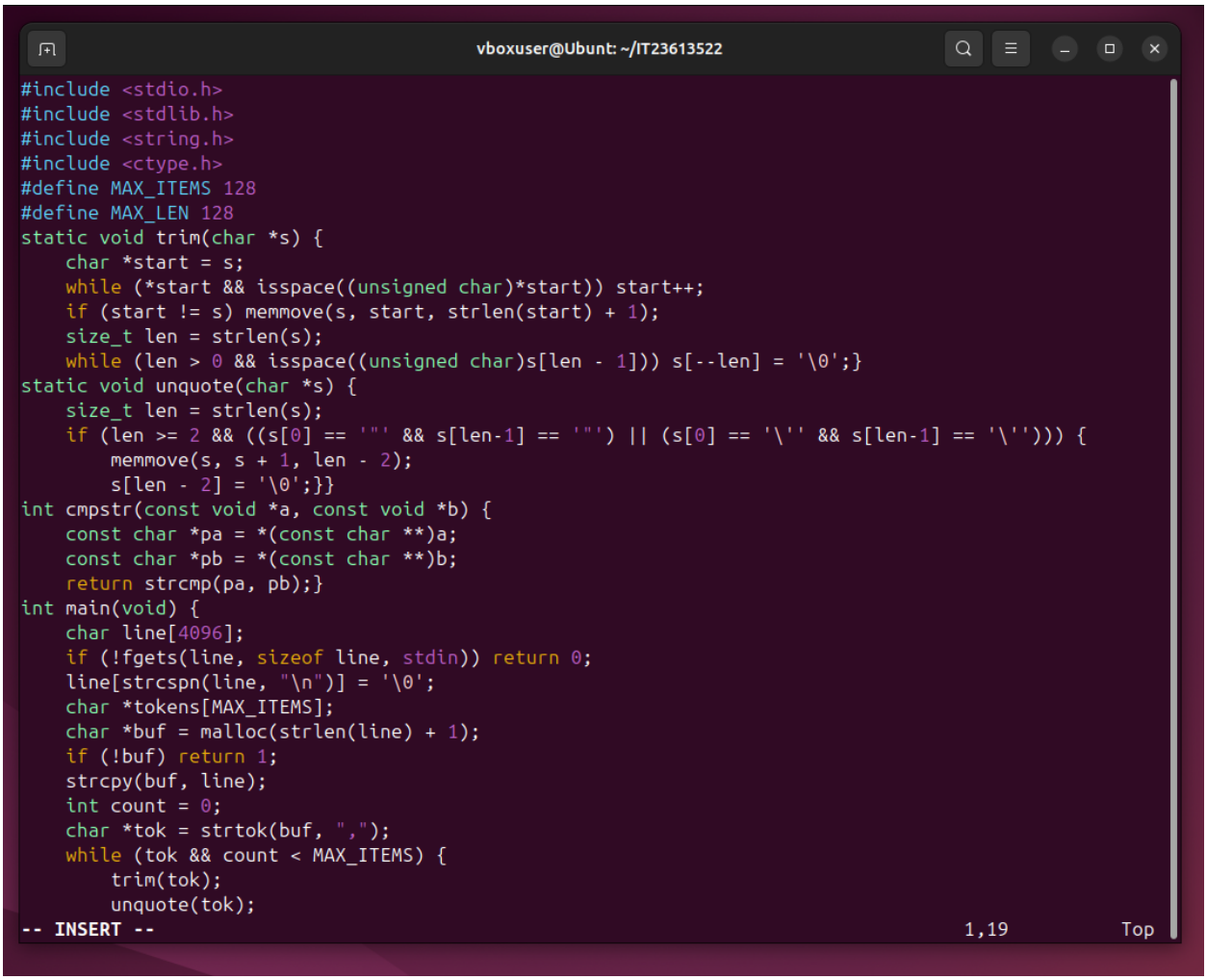
Programming Methodology - SE1012

Lab 10

IT23613522

Question 1: String Comparison



vboxuser@Ubunt: ~/IT23613522

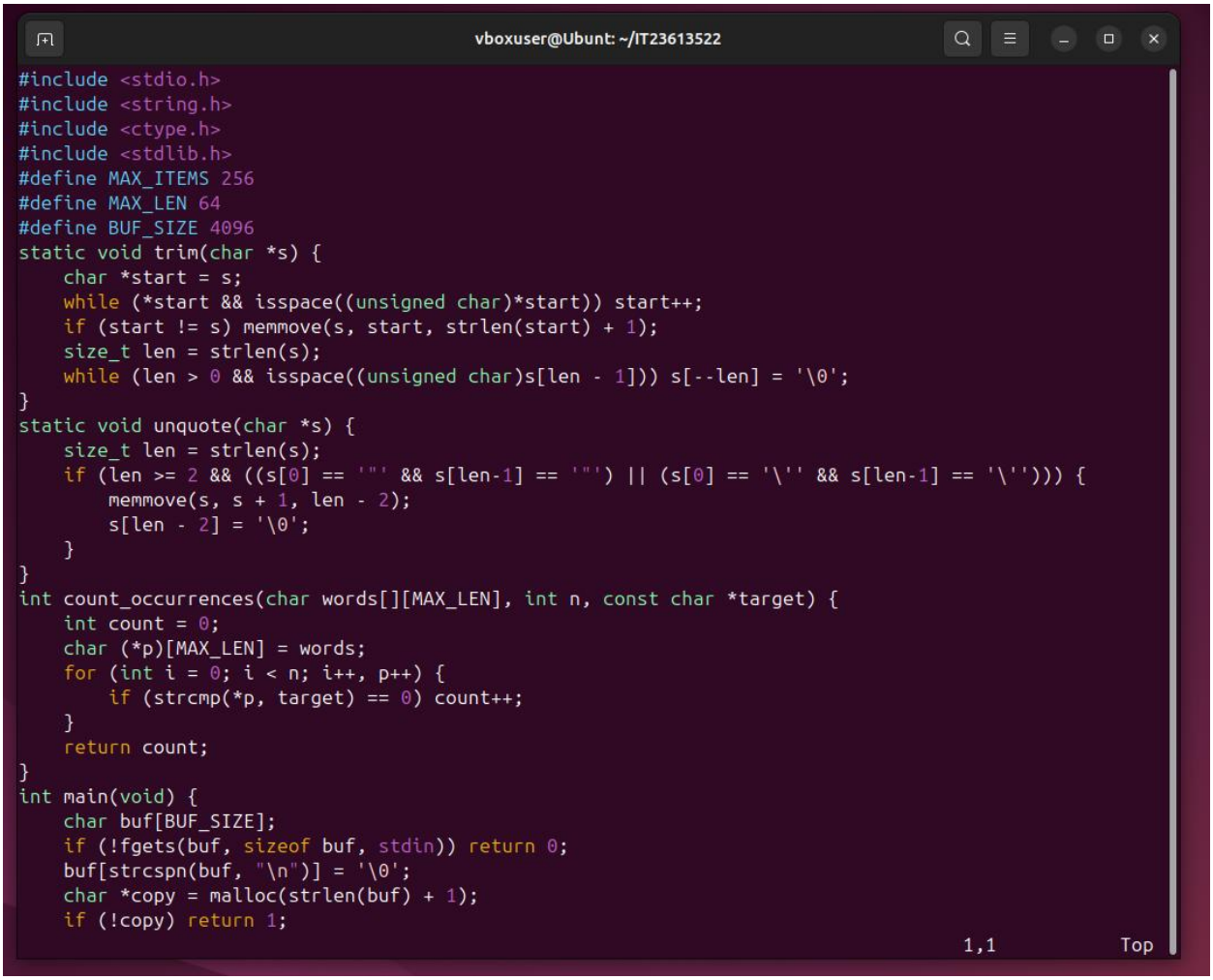
```c
int cmpstr(const void *a, const void *b) {
    const char *pa = *(const char **)a;
    const char *pb = *(const char **)b;
    return strcmp(pa, pb);}
int main(void) {
    char line[4096];
    if (!fgets(line, sizeof line, stdin)) return 0;
    line[strcspn(line, "\n")] = '\0';
    char *tokens[MAX_ITEMS];
    char *buf = malloc(strlen(line) + 1);
    if (!buf) return 1;
    strcpy(buf, line);
    int count = 0;
    char *tok = strtok(buf, ",");
    while (tok && count < MAX_ITEMS) {
        trim(tok);
        unquote(tok);
        size_t len = strlen(tok);
        char *stored = malloc(len + 1);
        if (!stored) break;
        strcpy(stored, tok);
        tokens[count++] = stored;
        tok = strtok(NULL, ",");}
    if (count == 0) {
        free(buf);
        return 0;}
    qsort(tokens, count, sizeof(char *), cmpstr);
    for (int i = 0; i < count; i++) {
        if (i) printf(", ");
        printf("%s", tokens[i]);
        free(tokens[i]);}
    printf("\n");
    free(buf);
    return 0;}
```

-- INSERT -- 51,15 Bot

vboxuser@Ubunt: ~/IT23613522

```
vboxuser@Ubunt:~/IT23613522$ gcc s.c -o s
vboxuser@Ubunt:~/IT23613522$ ./s
banana,apple,cherry
apple, banana, cherry
vboxuser@Ubunt:~/IT23613522$
```

Question 2: Arrays of Pointers



vboxuser@Ubunt: ~/IT23613522

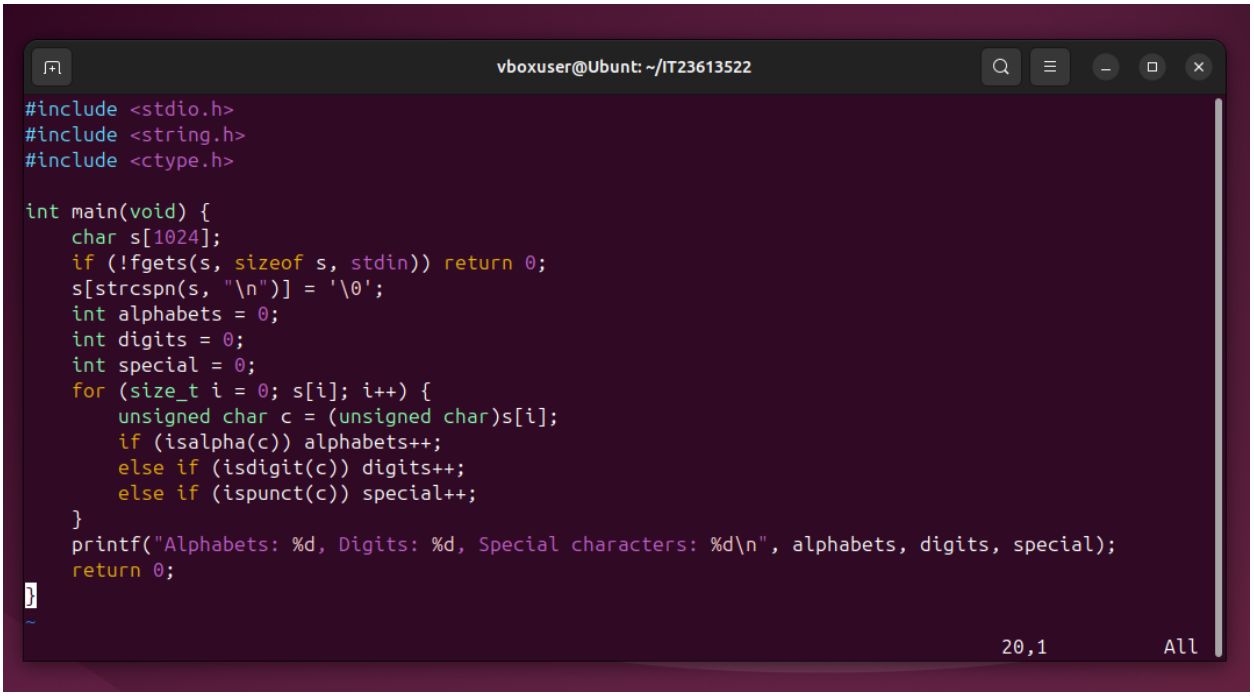
```c
int count_occurrences(char words[][MAX_LEN], int n, const char *target) {
    int count = 0;
    char (*p)[MAX_LEN] = words;
    for (int i = 0; i < n; i++, p++) {
        if (strcmp(*p, target) == 0) count++;
    }
    return count;
}
int main(void) {
    char buf[BUF_SIZE];
    if (!fgets(buf, sizeof buf, stdin)) return 0;
    buf[strcspn(buf, "\n")] = '\0';
    char *copy = malloc(strlen(buf) + 1);
    if (!copy) return 1;
    strcpy(copy, buf);
    char words[MAX_ITEMS][MAX_LEN];
    int count = 0;
    char *tok = strtok(copy, ",");
    while (tok && count < MAX_ITEMS) {
        trim(tok);
        unquote(tok);
        strncpy(words[count], tok, MAX_LEN - 1);
        words[count][MAX_LEN - 1] = '\0';
        count++;
        tok = strtok(NULL, ",");
    }
    free(copy);
    if (!fgets(buf, sizeof buf, stdin)) return 0;
    buf[strcspn(buf, "\n")] = '\0';
    trim(buf);
    unquote(buf);
    int occurrences = count_occurrences(words, count, buf);
    printf("The word '%s' appears %d %s.\n", buf, occurrences, occurrences == 1 ? "time" : "times");
    return 0;
}
```

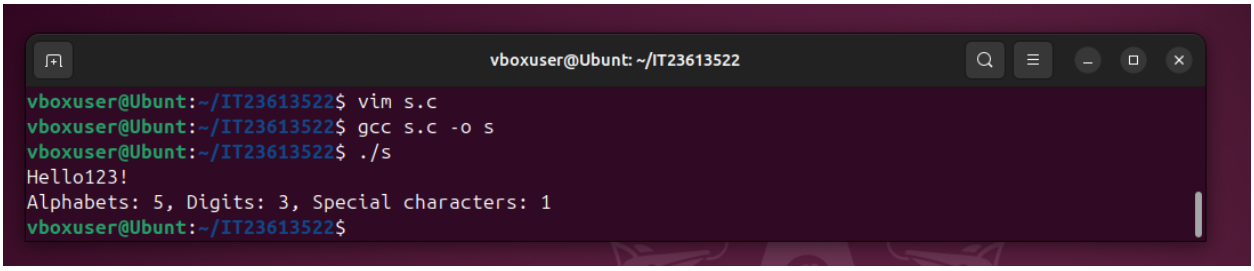
56,1 Bot

vboxuser@Ubunt: ~/IT23613522

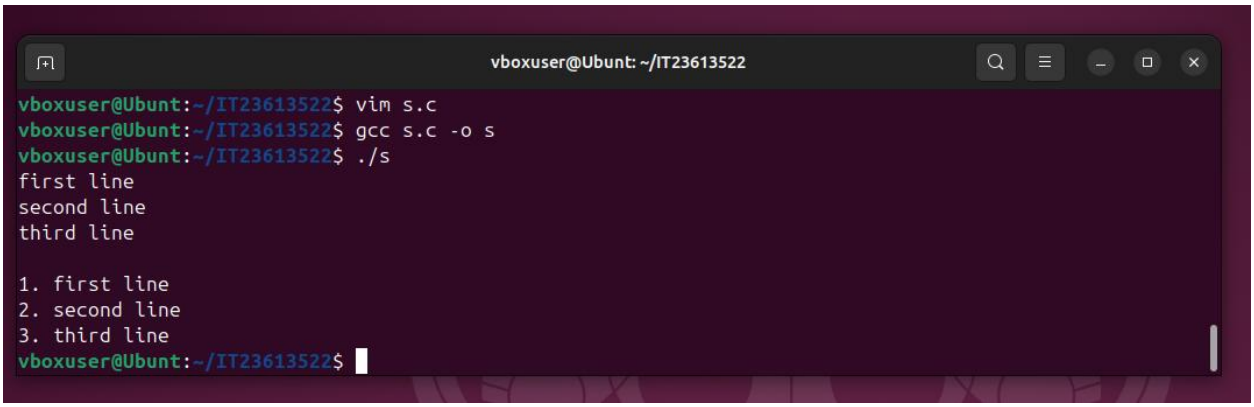
```
vboxuser@Ubunt:~/IT23613522$ vim s.c
vboxuser@Ubunt:~/IT23613522$ gcc s.c -o s
vboxuser@Ubunt:~/IT23613522$ ./s
apple, apple, banana, apple
apple
The word 'apple' appears 3 times.
vboxuser@Ubunt:~/IT23613522$
```

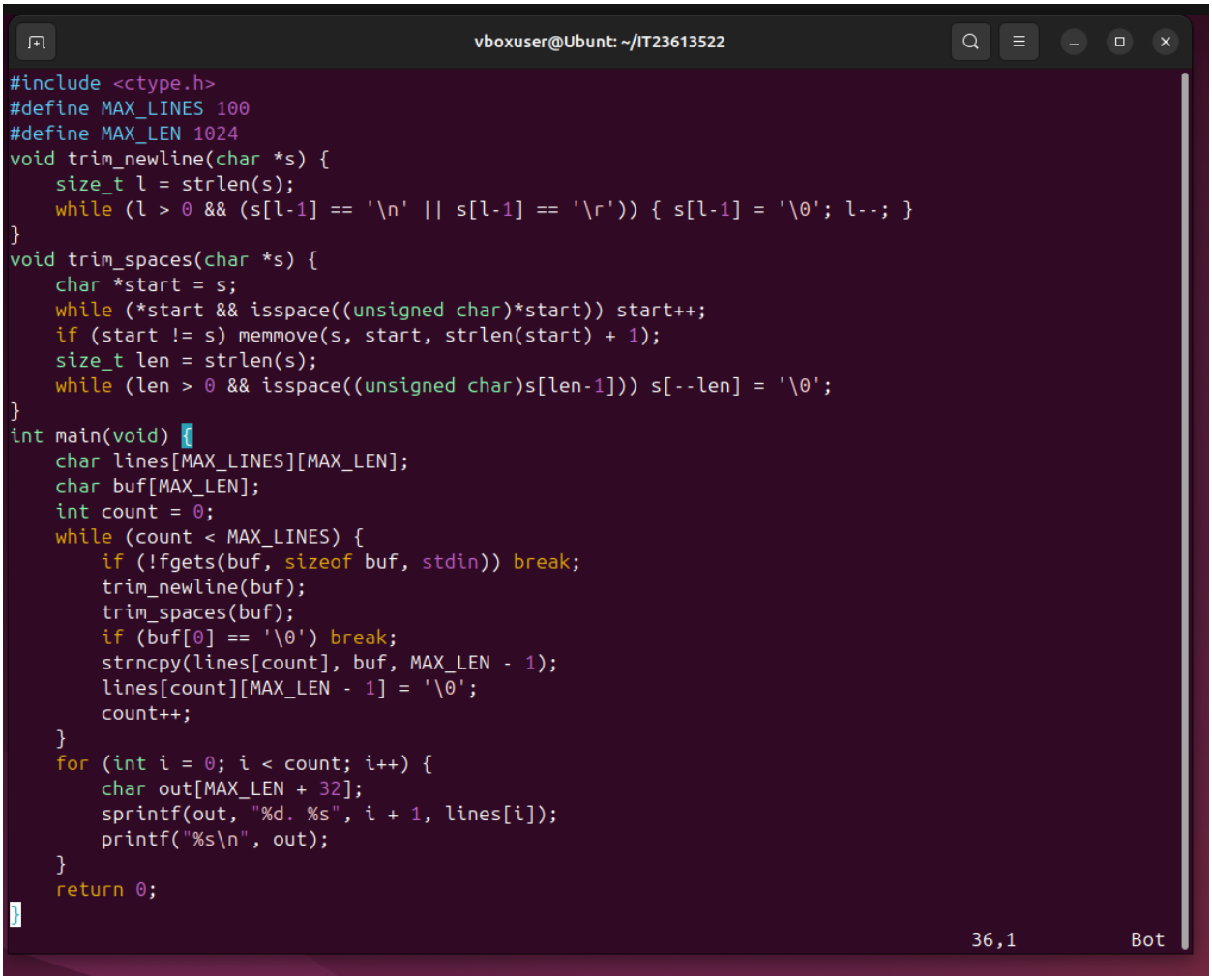
Question 3: Character Operations



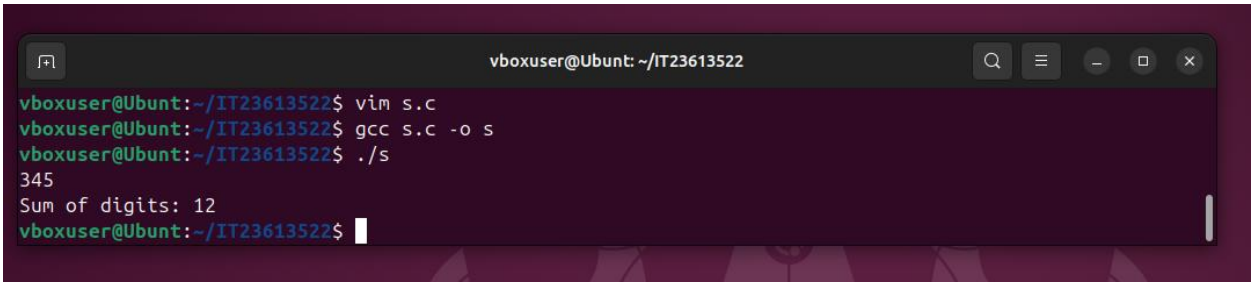


Question 4: Using Strings in Programs





Question 5: The Nature of Recursion



Question 6: Sum ofArray Elements Using Recursion (With Tracing)