

```
1: #include <stdio.h>
2:
3: int main(void)
4: {
5:     char seq1[10], seq2[10];
6:
7:     printf("Enter sequence 1: ");
8:     for (int i = 0; i < 10; i++)
9:         scanf(" %c", &seq1[i]);
10:    printf("Enter sequence 2: ");
11:    for (int i = 0; i < 10; i++)
12:        scanf(" %c", &seq2[i]);
13:
14:    for (int i = 0; i < 10; i++)
15:    {
16:        if ((seq1[i] != 'A') && (seq1[i] != 'T') && (seq1[i] != 'G') && (seq1[i]
!= 'C') && (seq1[i] != 'a') && (seq1[i] != 't') && (seq1[i] != 'g') && (seq1[i] != 'c'))
17:        {
18:            printf("Wrong Input ! \n");
19:            return 0;
20:        }
21:        if ((seq2[i] != 'A') && (seq2[i] != 'T') && (seq2[i] != 'G') && (seq2[i]
!= 'C') && (seq2[i] != 'a') && (seq2[i] != 't') && (seq2[i] != 'g') && (seq2[i] != 'c'))
22:        {
23:            printf("Wrong Input ! \n");
24:            return 0;
25:        }
26:
27:        if (seq1[i] >= 97)
28:            seq1[i] -= 32;
29:        if (seq2[i] >= 97)
30:            seq2[i] -= 32;
31:    }
32:
33:    int len = 0, lenstore = 0, first;
34:    for (int j = 0; j < 10; j++)
35:    {
36:        if (seq1[j] == seq2[j])
37:        {
38:            len++;
39:
40:            if (len > lenstore) {
41:                first = j - len + 1;
42:                lenstore = len;
43:            }
44:
45:        }
46:        else
47:            len = 0;
48:    }
49:
50:    if ((lenstore == 0) && (len == 0))
51:        first = -1;
52:
53:    printf("sequence 1: ");
54:    for (int i = 0; i < 10; i++)
55:        printf("%c", seq1[i]);
56:    printf("\n");
57:
58:    printf("sequence 2: ");
59:    for (int i = 0; i < 10; i++)
60:        printf("%c", seq2[i]);
61:    printf("\n");
62:
63:    printf("Longest match length: %d\n", lenstore);
64:    printf("Starts at %d\n", first);
65:
```

```
66: return 0;
67: }
```

```
1: #include <stdio.h>
2:
3: int main(void)
4: {
5:     char seq[10];
6:     int patlen;
7:
8:     printf("Enter sequence: ");
9:     for (int i = 0; i < 10; i++)
10:         scanf(" %c", &seq[i]);
11:
12:     for (int i = 0; i < 10; i++)
13:         if ((seq[i] != 'A') && (seq[i] != 'T') && (seq[i] != 'G') && (seq[i] !=
'C'))
14:         {
15:             printf("Wrong Input ! \n");
16:             return 0;
17:         }
18:
19:     printf("Enter pattern length: ");
20:     scanf("%d", &patlen);
21:     if ((patlen < 0) || (patlen > 10))
22:     {
23:         printf("Wrong Input ! \n");
24:         return 0;
25:     }
26:
27:     char pat[patlen];
28:     printf("Enter pattern: ");
29:     for (int i = 0; i < patlen; i++)
30:         scanf(" %c", &pat[i]);
31:
32:     for (int i = 0; i < patlen; i++)
33:         if ((pat[i] != 'A') && (pat[i] != 'T') && (pat[i] != 'G') && (pat[i] !=
'C'))
34:         {
35:             printf("Wrong Input ! \n");
36:             return 0;
37:         }
38:
39:
40:     int count=0, valid=0;
41:     int index[10]={};
42:     for (int j = 0; j < 10; j++)
43:     {
44:         if (seq[j] == pat[0]) {
45:
46:             valid = 0;
47:
48:             for (int i=1; i<patlen; i++) {
49:
50:                 if ((j+i) > 9)
51:                     break;
52:
53:                 if (seq[j+i] == pat[i])
54:                     valid++;
55:                 else {
56:                     valid = 0;
57:                     break;
58:                 }
59:             }
60:
61:             if (valid == (patlen-1)) {
62:                 count++;
63:                 index[j] = 1;
64:             }
65:
```

```
66:         }
67:     }
68:
69:     printf("Count: %d\n", count);
70:     printf("Positions: ");
71:     if (count == 0)
72:         printf("%d\n", -1);
73:     else {
74:         for (int i=0; i<10; i++)
75:             if (index[i])
76:                 printf("%d ", i);
77:         printf("\n");
78:     }
79:
80:
81:     return 0;
82: }
```

```
1: #include <stdio.h>
2:
3: int main() {
4:
5:     int arr[5][5] = {};
6:     int N, max=-1, max_r, max_c;
7:
8:     printf("Enter N: ");
9:     scanf("%d", &N);
10:    if ((N<1) || (N>5)) {
11:        printf("Wrong Input ! \n");
12:        return 0;
13:    }
14:
15:    printf("Enter %d x %d elements: ", N, N);
16:    for (int i=0; i<N; i++) { // row
17:
18:        for (int j=0; j<N; j++) { // col
19:            scanf("%d", &arr[i][j]);
20:            if (arr[i][j] > max) {
21:                max = arr[i][j];
22:                max_r = i;
23:                max_c = j;
24:            } else if (arr[i][j] == max) {
25:                if ((max_r - max_c) > (i-j)) {
26:                    max_r = i;
27:                    max_c = j;
28:                } else if ((max_r - max_c) == (i-j)) {
29:                    if (max_r > i) {
30:                        max_r = i;
31:                        max_c = j;
32:                    }
33:                }
34:            }
35:        }
36:    }
37:
38:    printf("matrix A: \n");
39:    for (int i=0; i<N; i++) { // row
40:        for (int j=0; j<N; j++) // col
41:            printf("%d", arr[i][j]);
42:        printf("\n");
43:    }
44:
45:    printf("max = %d (row=%d, col=%d)\n", max, max_r, max_c);
46:
47:
48:    int tmp;
49:
50:    for (int i=0; i<(N/2); i++) {
51:
52:        for (int j=i; j<(N-1-i); j++) {
53:            tmp = (*(arr+i) + j);
54:
55:            (*(arr+i) + j) = (*(arr+(N-1-j)) + i);
56:
57:            (*(arr + (N-1-j)) + i) = (*(arr + (N-1-i)) + (N-1-j));
58:
59:            (*(arr+(N-1-i)) + (N-1-j)) = (*(arr+j) + (N-1-i));
60:
61:            (*(arr+j) + (N-1-i)) = tmp;
62:        }
63:    }
64:
65:    printf("After rotation A: \n");
66:    for (int i=0; i<N; i++) { // row
67:        for (int j=0; j<N; j++) // col
```

```
68:            printf(" %d", (*(arr+i) + j));
69:        printf("\n");
70:    }
71:
72:    return 0;
73: }
```

```
1: #include <stdio.h>
2:
3: int leap_year(int year);
4: int month_calc(int year, int month);
5: int month_first_calc(int year, int month);
6: void print_days(int week);
7:
8: int main() {
9:
10:     int year, month, week;
11:
12:     printf("Enter year, month, first day of week (1=Mon, 2=Tue, 3=Wed, ...,
7=Sun): ");
13:     scanf("%d %d %d", &year, &month, &week);
14:
15:     if ((year < 1) || (year > 9999)) {
16:         printf("Wrong Input ! \n");
17:         return 0;
18:     }
19:
20:     if ((month < 1) || (month > 12)) {
21:         printf("Wrong Input ! \n");
22:         return 0;
23:     }
24:
25:     if ((week < 1) || (week > 7)) {
26:         printf("Wrong Input ! \n");
27:         return 0;
28:     }
29:
30:     char month_names[12][3] = {
31:         {'J', 'a', 'n'}, {'F', 'e', 'b'}, {'M', 'a', 'r'}, {'A', 'p', 'r'},
32:         {'M', 'a', 'y'}, {'J', 'u', 'n'}, {'J', 'u', 'l'}, {'A', 'u', 'g'},
33:         {'S', 'e', 'p'}, {'O', 'c', 't'}, {'N', 'o', 'v'}, {'D', 'e', 'c'}
34:     };
35:
36:     printf("\n\t %d ", year);
37:     printf("%c%c%c\n", month_names[month-1][0], month_names[month-1][1],
month_names[month-1][2]);
38:
39:     print_days(week);
40:     printf("\n");
41:
42:     int days_in_month = month_calc(year, month);
43:     int start_day_of_month = month_first_calc(year, month);
44:
45:     int padding = (start_day_of_month - week + 7) % 7;
46:
47:     int calendar[6][7] = {};
48:
49:     int day_count = 1;
50:     for (int i = 0; i < 6; i++) // if%
51:         for (int j = 0; j < 7; j++) { // i\235%
52:             if (i == 0 && j < padding)
53:                 calendar[i][j] = 0;
54:             else if (day_count <= days_in_month) {
55:                 calendar[i][j] = day_count;
56:                 day_count++;
57:             } else
58:                 calendar[i][j] = 0;
59:         }
60:
61:     for (int i = 0; i < 6; i++) {
62:         int row_has_data = 0;
63:
64:         for (int j = 0; j < 7; j++) {
65:             if (calendar[i][j] == 0)
```

```
66:                 printf("    ");
67:             else {
68:                 printf("%3d ", calendar[i][j]);
69:                 row_has_data = 1;
70:             }
71:         }
72:
73:         if (row_has_data == 0)
74:             break;
75:         printf("\n");
76:     }
77:
78:     return 0;
79: }
80:
81: int leap_year(int year) {
82:     if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
83:         return 1;
84:     } else {
85:         return 0;
86:     }
87: }
88:
89:
90: void print_days(int week) {
91:     char day_names[7][3] = {
92:         {'M', 'o', 'n'}, {'T', 'u', 'e'}, {'W', 'e', 'd'},
93:         {'T', 'h', 'u'}, {'F', 'r', 'i'}, {'S', 'a', 't'}, {'S', 'u', 'n'}
94:     };
95:
96:     int start_index = week - 1;
97:
98:     for (int i = start_index; i < 7; i++) {
99:         printf("%c%c%c ", day_names[i][0], day_names[i][1], day_names[i][2]);
100:     }
101:
102:     for (int i = 0; i < start_index; i++) {
103:         printf("%c%c%c ", day_names[i][0], day_names[i][1], day_names[i][2]);
104:     }
105: }
106:
107:
108: int month_calc(int year, int month) {
109:     switch(month) {
110:         case 1:
111:         case 3:
112:         case 5:
113:         case 7:
114:         case 8:
115:         case 10:
116:         case 12:
117:             return 31;
118:             break;
119:
120:         case 4:
121:         case 6:
122:         case 9:
123:         case 11:
124:             return 30;
125:             break;
126:
127:         case 2:
128:             if(leap_year(year) == 1) {
129:                 return 29;
130:                 break;
131:             } else {
132:                 return 28;
```

```
133:                                     break;
134:                                     }
135:
136:                                     default:
137:                                     printf("Wrong Month ! \n");
138:                                     }
139: }
140:
141:
142: int month_first_calc(int year, int month)
143: {
144:     int total = 0;
145:
146:     for(int i=1; i<year; i++)
147:     {
148:         if (leap_year(i) == 1) {
149:             total += 366;
150:         } else {
151:             total += 365;
152:         }
153:     }
154:
155:     for(int i=1; i<month; i++)
156:         total += month_calc(year, i);
157:
158:     return (total % 7) + 1;
159: }
```