

```

1 #include <stdio.h>
2 int main() {
3     int V;
4     scanf("%d", &V);
5     int denominations[] = {1000, 500, 100, 50, 20, 10, 5, 2, 1};
6     int n = sizeof(denominations) / sizeof(denominations[0]);
7     int count = 0;
8     for (int i = 0; i < n; i++) {
9         if (V >= denominations[i]) {
10             count += V / denominations[i];
11             V = V % denominations[i];
12         }
13     }
14     printf("%d\n", count);
15     return 0;
16 }

```

|   | Input | Expected | Got |   |
|---|-------|----------|-----|---|
| ✓ | 49    | 5        | 5   | ✓ |

Passed all tests! ✓

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int cmp(const void *a, const void *b) {
4     return (*(int *)a - *(int *)b);
5 }
6 int main() {
7     int n, m;
8     scanf("%d", &n);
9     int g[n];
10    for (int i = 0; i < n; i++) {
11        scanf("%d", &g[i]);
12    }
13    scanf("%d", &m);
14    int s[m];
15    for (int i = 0; i < m; i++) {
16        scanf("%d", &s[i]);
17    }
18    qsort(g, n, sizeof(int), cmp);
19    qsort(s, m, sizeof(int), cmp);
20
21    int i = 0, j = 0, count = 0;
22    while (i < n && j < m) {
23        if (s[j] >= g[i]) {
24            count++;
25            i++;
26            j++;
27        } else {
28            j++;
29        }
30    }
31
32    printf("%d\n", count);
33    return 0;
34 }
```

```

1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<math.h>
4 int compare(const void *a, const void *b){
5     int x = *(int*)a;
6     int y = *(int*)b;
7     return y - x;
8 }
9 int main(){
10     int n;
11     scanf("%d", &n);
12     int calories[n];
13     for(int i=0;i<n;i++){
14         scanf("%d", &calories[i]);
15     }
16     qsort(calories,n,sizeof(int),compare);
17     long long distance=0;
18     for(int i=0;i<n;i++){
19         distance+=pow(n,i)*calories[i];
20     }
21     printf("%lld\n",distance);
22     return 0;
23 }

```

|   | Test        | Input        | Expected | Got |   |
|---|-------------|--------------|----------|-----|---|
| ✓ | Test Case 1 | 3<br>1 3 2   | 18       | 18  | ✓ |
| ✓ | Test Case 2 | 4<br>7 4 9 6 | 389      | 389 | ✓ |

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 int cmp(const void *a, const void *b) {
4     return (*(int *)a - *(int *)b);
5 }
6 int main() {
7     int n;
8     scanf("%d", &n);
9
10    int arr[n];
11    for (int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14    qsort(arr, n, sizeof(int), cmp);
15
16    long long sum = 0;
17    for (int i = 0; i < n; i++) {
18        sum += (long long)arr[i] * i;
19    }
20
21    printf("%lld\n", sum);
22
23    return 0;
24 }

```

|   | Input | Expected | Got |   |
|---|-------|----------|-----|---|
| ✓ | 5     | 40       | 40  | ✓ |
|   | 2     |          |     |   |
|   | 5     |          |     |   |
|   | 3     |          |     |   |
|   | 4     |          |     |   |

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int cmpAsc(const void *a, const void *b) {
4     return (*(int *)a - *(int *)b);
5 }
6 int cmpDesc(const void *a, const void *b) {
7     return (*(int *)b - *(int *)a);
8 }
9
10 int main() {
11     int n;
12     scanf("%d", &n);
13     int A[n], B[n];
14     for (int i = 0; i < n; i++) {
15         scanf("%d", &A[i]);
16     }
17     for (int i = 0; i < n; i++) {
18         scanf("%d", &B[i]);
19     }
20     qsort(A, n, sizeof(int), cmpAsc);
21     qsort(B, n, sizeof(int), cmpDesc);
22
23     long long sum = 0;
24     for (int i = 0; i < n; i++) {
25         sum += (long long)A[i] * B[i];
26     }
27
28     printf("%lld\n", sum);
29     return 0;
30 }
```